SIEMENS

SINUMERIK

SINUMERIK 840D sl
Programmation CN

Manuel de programmation

Avant-propos

Consignes de sécurité élémentaires 1

Notions de base 2

Notions complémentaires 3

Tableaux 4

Annexe A

Valable pour

Commande
SINUMERIK 840D sl / 840DE sl

Logiciel CNC version 4.94

10/2020
6FC5398-2EP40-0DA1

Промэлектроника, sales@prom-elec.com

Mentions légales

Signalétique d'avertissement

Ce manuel donne des consignes que vous devez respecter pour votre propre sécurité et pour éviter des dommages matériels. Les avertissements servant à votre sécurité personnelle sont accompagnés d'un triangle de danger, les avertissements concernant uniquement des dommages matériels sont dépourvus de ce triangle. Les avertissements sont représentés ci-après par ordre décroissant de niveau de risque.

⚠ **DANGER**
signifie que la non-application des mesures de sécurité appropriées **entraîne** la mort ou des blessures graves.

⚠ **ATTENTION**
signifie que la non-application des mesures de sécurité appropriées **peut entraîner** la mort ou des blessures graves.

⚠ **PRUDENCE**
signifie que la non-application des mesures de sécurité appropriées peut entraîner des blessures légères.

IMPORTANT
signifie que la non-application des mesures de sécurité appropriées peut entraîner un dommage matériel.

En présence de plusieurs niveaux de risque, c'est toujours l'avertissement correspondant au niveau le plus élevé qui est reproduit. Si un avertissement avec triangle de danger prévient des risques de dommages corporels, le même avertissement peut aussi contenir un avis de mise en garde contre des dommages matériels.

Personnes qualifiées

L'appareil/le système décrit dans cette documentation ne doit être manipulé que par du **personnel qualifié** pour chaque tâche spécifique. La documentation relative à cette tâche doit être observée, en particulier les consignes de sécurité et avertissements. Les personnes qualifiées sont, en raison de leur formation et de leur expérience, en mesure de reconnaître les risques liés au maniement de ce produit / système et de les éviter.

Utilisation des produits Siemens conforme à leur destination

Tenez compte des points suivants:

⚠ **ATTENTION**
Les produits Siemens ne doivent être utilisés que pour les cas d'application prévus dans le catalogue et dans la documentation technique correspondante. S'ils sont utilisés en liaison avec des produits et composants d'autres marques, ceux-ci doivent être recommandés ou agréés par Siemens. Le fonctionnement correct et sûr des produits suppose un transport, un entreposage, une mise en place, un montage, une mise en service, une utilisation et une maintenance dans les règles de l'art. Il faut respecter les conditions d'environnement admissibles ainsi que les indications dans les documentations afférentes.

Marques de fabrique

Toutes les désignations repérées par ® sont des marques déposées de Siemens AG. Les autres désignations dans ce document peuvent être des marques dont l'utilisation par des tiers à leurs propres fins peut enfreindre les droits de leurs propriétaires respectifs.

Exclusion de responsabilité

Nous avons vérifié la conformité du contenu du présent document avec le matériel et le logiciel qui y sont décrits. Ne pouvant toutefois exclure toute divergence, nous ne pouvons pas nous porter garants de la conformité intégrale. Si l'usage de ce manuel devait révéler des erreurs, nous en tiendrons compte et apporterons les corrections nécessaires dès la prochaine édition.

Siemens AG
Digital Industries
Postfach 48 48
90026 NÜRNBERG
ALLEMAGNE

Numéro de référence du document: 6FC5398-2EP40-0DA1
Ⓟ 10/2020 Sous réserve de modifications

Copyright © Siemens AG 2019 - 2020.
Tous droits réservés

Avant-propos

Documentation SINUMERIK

La documentation SINUMERIK comporte les catégories suivantes :

- Documentation générale/catalogues
- Documentation utilisateur
- Documentation constructeur/S.A.V.

Informations complémentaires

Sous ce lien (https://support.industry.siemens.com/cs/de/en/view/108464614), vous trouverez différentes informations pour :

- Commander la documentation/aperçu des documents
- Télécharger des documents à partir de liens supplémentaires
- Utiliser la documentation en ligne (trouver des manuels/informations et y effectuer des recherches)

Pour toute question concernant la documentation technique (par ex. suggestion, correction), envoyez un courriel à cette adresse (mailto:docu.motioncontrol@siemens.com).

mySupport/Documentation

Plus d'informations sur la documentation Siemens disponible et sur les possibilités de l'utiliser pour créer votre propre documentation machine, voir : lien (https://support.industry.siemens.com/My/ww/fr/documentation).

Formation

Ce lien (http://www.siemens.com/sitrain) livre des informations sur SITRAIN, le programme de formations de Siemens pour les produits, systèmes et solutions d'entraînements et d'automatisation.

FAQ

La Foire Aux Questions se trouve sur les pages Service&Support sous Support produit (https://support.industry.siemens.com/cs/de/en/ps/faq).

SINUMERIK

Des informations relatives à SINUMERIK figurent à cette adresse (http://www.siemens.com/sinumerik).

Groupe cible

Le présent manuel s'adresse aux :

- programmeurs
- ingénieurs de projet

Utilité

Le Manuel de programmation permet au groupe cible de créer, d'écrire, de tester des programmes et des interfaces logicielles et de supprimer des erreurs.

Version standard

Le présent manuel de programmation décrit les fonctionnalités de la version standard. Les options complémentaires ou modifications apportées par le constructeur de la machine-outil sont documentées par celui-ci.

La commande numérique peut posséder des fonctions qui dépassent le cadre de la présente description. Le client ne peut toutefois pas faire valoir de droit en liaison avec ces fonctions, que ce soit dans le cas de matériels neufs ou dans le cadre d'interventions du service après-vente.

De même, pour des raisons de clarté, la présente documentation ne contient pas toutes les informations de détail relatives à toutes les variantes du produit ; elle ne peut pas non plus tenir compte de tous les cas d'installation, d'exploitation et de maintenance.

Remarque concernant le règlement général sur la protection des données

Siemens respecte les principes de la protection des données, en particulier l'impératif de minimisation des données (protection de la vie privée dès la conception). Pour ce produit, cela signifie :

Le produit ne traite/n'enregistre aucune donnée à caractère personnel mais uniquement des données techniques fonctionnelles (p. ex. horodatage). Si l'utilisateur relie ces données à d'autres données (p. ex. plannings d'équipes) ou s'il enregistre des données à caractère personnel sur le même support (p. ex. disque dur) et crée par là même un lien avec des personnes, il est tenu de garantir le respect des prescriptions en matière de protection des données.

Technical Support

Si vous avez besoin de conseils techniques, vous trouverez sur Internet les coordonnées téléphoniques appropriées à l'adresse (https://support.industry.siemens.com/cs/ww/fr/sc/2090) suivante, dans la zone "Contact".

Pour poser une question technique, utilisez le formulaire en ligne dans la zone de demande d'assistance "Support Request".

Informations sur la structure et le contenu de la documentation

La documentation relative à la programmation CN est divisée en deux chapitres principaux :

1. **Notions de base**
   Le chapitre "Notions de base" s'adresse aux techniciens utilisant la machine-outil et suppose d'avoir la connaissance préalable des opérations de perçage, de fraisage et de tournage. Les instructions et commandes, conformes à la norme DIN 66025, sont illustrées par des exemples de programmation simples.
2. **Notions complémentaires**
   Le chapitre "Notions complémentaires" s'adresse aux technologues connaissant toutes les possibilités de programmation. Grâce à un langage de programmation spécifique, la commande SINUMERIK permet d'élaborer un programme pièce complexe (par exemple : surfaces gauches, coordination de canaux...) et dispense le technologue d'une programmation fastidieuse.

Sommaire

Avant-propos **3**

1 Consignes de sécurité élémentaires **23**

1.1 Consignes de sécurité générales 23

1.2 Garantie et responsabilité pour les exemples d'application 24

1.3 Note relative à la sécurité 25

2 Notions de base **27**

2.1 Notions géométriques de base 27
2.1.1 Positions des pièces 27
2.1.1.1 Système de référence des indications de position 27
2.1.1.2 Coordonnées cartésiennes 27
2.1.1.3 Coordonnées polaires 29
2.1.1.4 Cote absolue 30
2.1.1.5 Cote relative 32
2.1.2 Plans de travail 33
2.1.3 Origines et points de référence 34
2.1.4 Systèmes de coordonnées 36
2.1.4.1 Système de coordonnées machine (SCM) 36
2.1.4.2 Système de coordonnées de base (SCB) 38
2.1.4.3 Système de coordonnées de base décalé (SCD) 39
2.1.4.4 Système de coordonnées réglable (SCR) 40
2.1.4.5 Système de coordonnées pièce (SCP) 41
2.1.4.6 Quels sont les rapports entre les différents systèmes de coordonnées ? 42

2.2 Bases de la programmation de la commande numérique 43
2.2.1 Dénomination d'un programme CN 43
2.2.2 Structure et contenu d'un programme CN 44
2.2.2.1 Blocs et composantes de blocs 44
2.2.2.2 Règles valables pour les blocs 47
2.2.2.3 Affectation de valeurs 48
2.2.2.4 Commentaires 48
2.2.2.5 Masquage de blocs 49

2.3 Création d'un programme CN 51
2.3.1 Procédure générale 51
2.3.2 Caractères disponibles 52
2.3.3 En-tête du programme 53
2.3.4 Exemples de programmes 55
2.3.4.1 Exemple 1 : Premières opérations de programmation 55
2.3.4.2 Exemple 2 : Programme CN de tournage 55
2.3.4.3 Exemple 3 : Programme CN de fraisage 57

2.4 Changement d'outil 60
2.4.1 Changement d'outil lorsque la gestion des outils est active 61
2.4.1.1 Changement d'outil lorsque la gestion des outils est active avec la fonction T=emplacement 62

2.4.1.2 Changement d'outil lorsque la gestion des outils est active avec M6 ... 64
2.4.2 Changement d'outil lorsque la gestion des outils est inactive ... 65
2.4.2.1 Changement d'outil lorsque la gestion des outils est inactive avec numéro T ... 65
2.4.2.2 Changement d'outil lorsque la gestion des outils est inactive avec M6 ... 66
2.4.3 Réaction en cas de programmation erronée de T ... 67

2.5 Corrections d'outils ... 68
2.5.1 Contour et trajectoire d'outil programmés ... 68
2.5.2 Correction de longueur d'outil ... 68
2.5.3 Correction du rayon d'outil ... 69
2.5.4 Mémoire de corrections d'outil ... 70
2.5.5 Types d'outil ... 71
2.5.5.1 Numéro de type d'outil et groupes d'outils ... 71
2.5.5.2 Fraises ... 72
2.5.5.3 Forets ... 74
2.5.5.4 Outils de rectification ... 75
2.5.5.5 Outils de tournage ... 77
2.5.5.6 Outils spéciaux ... 79
2.5.6 Activation/désactivation des corrections d'outil (D, D0) ... 81
2.5.7 Inhibition des corrections d'outil (SUPD) ... 83
2.5.8 Offset de correction d'outil programmable (TOFFL, TOFF, TOFFR, TOFFLR) ... 85

2.6 Mouvement de broche ... 91
2.6.1 Vitesse de rotation de broche (S), sens de rotation (M3, M4, M5) ... 91
2.6.2 Vitesse de coupe d'outil (SVC) ... 94
2.6.3 Vitesse de coupe constante (G96/G961/G962, G97/G971/G972, G973, LIMS, SCC) ... 100
2.6.4 Activer/désactiver la vitesse circonférentielle de meule constante (GWPSON, GWPSOF) ... 106
2.6.5 Limitation programmable de la vitesse de rotation de broche (G25, G26) ... 106

2.7 Régulation d'avance ... 108
2.7.1 Avance (G93, G94, G95, F, FGROUP, FL, FGREF) ... 108
2.7.2 Déplacement des axes de positionnement (POS, POSA, POSP, FA, WAITP, WAITMC) ... 116
2.7.3 Broche en asservissement de position (SPCON, SPCOF) ... 120
2.7.4 Positionner des broches (SPOS, SPOSA, M19, M70, WAITS) ... 121
2.7.5 Avance pour axes/broches et de positionnement (FA, FPR, FPRAON, FPRAOF) ... 125
2.7.6 Correction programmable de l'avance (OVR, OVRRAP, OVRA) ... 129
2.7.7 Correction programmable de l'accélération (ACC) ... 130
2.7.8 Avance avec correction par manivelle (FD, FDA) ... 131
2.7.9 Optimisation de l'avance sur contours courbes (CFTCP, CFC, CFIN) ... 135
2.7.10 Plusieurs des valeurs d'avance dans un bloc (F, ST, SR, FMA, STA, SRA) ... 137
2.7.11 Avance à effet non modal (FB) ... 140
2.7.12 Avance par dent (G95 FZ) ... 141

2.8 Paramètres géométriques ... 147
2.8.1 Décalage d'origine réglable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153) ... 147
2.8.2 Décalage d'origine réglable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153) : Informations complémentaires ... 148
2.8.3 Sélection du plan de travail (G17/G18/G19) ... 149
2.8.4 Cotes ... 152
2.8.4.1 Indication de cotes absolues (G90, AC) ... 153
2.8.4.2 Indication de cotes relatives (G91, IC) ... 155
2.8.4.3 Indication de cotes absolues et de cotes relatives pour le tournage et le fraisage (G90/G91) ... 159
2.8.4.4 Introduction en cotes absolues pour axes rotatifs (DC, ACP, ACN) ... 160
2.8.4.5 Système de mesure en pouces / métrique (G70/G71, G700/G710) ... 162

2.8.4.6 Programmation diamétrale/radiale spécifique au canal (DIAMON, DIAM90, DIAMOF, DIAMCYCOF) ... 166
2.8.4.7 Programmation diamétrale/radiale spécifique à l'axe (DIAMONA, DIAM90A, DIAMOFA, DIACYCOFA, DIAMCHANA, DIAMCHAN, DAC, DIC, RAC, RIC) ... 168
2.8.5 Position de la pièce lors du tournage ... 172

2.9 Instructions de déplacement ... 174
2.9.1 Informations générales pour les instructions de déplacement ... 174
2.9.2 Instructions de déplacement avec des coordonnées cartésiennes (G0, G1, G2, G3, X..., Y..., Z...) ... 176
2.9.3 Instructions de déplacement avec des coordonnées polaires ... 177
2.9.3.1 Point de référence des coordonnées polaires (G110, G111, G112) ... 177
2.9.3.2 Instructions de déplacement avec des coordonnées polaires (G0, G1, G2, G3, AP, RP) ... 179
2.9.4 Déplacements en rapide ... 183
2.9.4.1 Activation de la vitesse rapide (G0) ... 183
2.9.4.2 Activer/désactiver l'interpolation linéaire des déplacements de vitesse rapide (RTLION, RTLIOF) ... 184
2.9.4.3 Adaptation des tolérances pour les déplacements en marche rapide (STOLF, CTOLG0, OTOLG0) ... 186
2.9.5 Interpolation linéaire (G1) ... 190
2.9.6 Interpolation circulaire ... 193
2.9.6.1 Vue d'ensemble ... 193
2.9.6.2 Interpolation circulaire avec centre de cercle et point final (G2/G3, X... Y... Z..., I... J... K...) ... 194
2.9.6.3 Interpolation circulaire avec rayon et point final (G2/G3, X... Y... Z..., CR) ... 196
2.9.6.4 Interpolation circulaire avec angle au centre et point final / centre de cercle (G2/G3, X... Y... Z... / I... J... K..., AR) ... 198
2.9.6.5 Interpolation circulaire avec coordonnées polaires (G2/G3, AP, RP) ... 200
2.9.6.6 Interpolation circulaire avec point intermédiaire et point final (CIP, X... Y... Z..., I1... J1... K1...) ... 202
2.9.6.7 Interpolation circulaire avec transition tangentielle (CT, X... Y... Z...) ... 204
2.9.7 Interpolation hélicoïdale (G2/G3, TURN) ... 208
2.9.8 Interpolation de développante (INVCW, INVCCW) ... 210
2.9.9 Eléments de contour ... 215
2.9.9.1 Programmation d'éléments de contour ... 215
2.9.9.2 Eléments de contour : une droite ... 216
2.9.9.3 Eléments de contour : deux droites ... 218
2.9.9.4 Eléments de contour : trois droites ... 221
2.9.9.5 Eléments de contour : Programmation du point final avec un angle ... 225
2.9.10 Filetage ... 225
2.9.10.1 Filetage avec pas constant (G33, SF) ... 225
2.9.10.2 Filetage avec pas croissant ou décroissant (G34, G35) ... 232
2.9.10.3 Courses d'accélération et de freinage programmées pour G33, G34 et G35 (DITS, DITE) ... 233
2.9.10.4 Retrait rapide pendant un filetage (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN) ... 235
2.9.10.5 Filetages bombés (G335, G336) ... 239
2.9.11 Taraudage sans porte-taraud compensateur ... 244
2.9.11.1 Taraudage sans porte-taraud compensateur et déplacement de retrait (G331, G332) ... 244
2.9.11.2 Exemple : Taraudage avec G331 / G332 ... 246
2.9.11.3 Exemple : édition de la vitesse de taraudage programmée dans le rapport de transmission actuel ... 246
2.9.11.4 Exemple : application du deuxième jeu de paramètres machine des rapports de transmission ... 247

2.9.11.5 Exemple : Aucune programmation de vitesse de rotation, surveillance du rapport de transmission ... 247
2.9.11.6 Exemple : Changement du rapport de transmission impossible, surveillance du rapport de transmission ... 248
2.9.11.7 Exemple : programmation sans SPOS ... 248
2.9.12 Taraudage avec porte-taraud compensateur ... 249
2.9.12.1 Taraudage avec porte-taraud compensateur et déplacement de retrait (G63) ... 249
2.9.13 Chanfrein, arrondissement (CHF, CHR, RND, RNDM, FRC, FRCM) ... 250

2.10 Corrections de rayon d'outil ... 256
2.10.1 Correction du rayon d'outil (G40, G41, G42, OFFN) ... 256
2.10.2 Accostage et retrait du contour (NORM, KONT, KONTC, KONTT) ... 265
2.10.3 Correction aux angles saillants (G450, G451, DISC) ... 272
2.10.4 Accostage et retrait en douceur ... 275
2.10.4.1 Accostage et retrait (G140 à G143, G147, G148, G247, G248, G347, G348, G340, G341, DISR, DISCL, DISRP, FAD, PM, PR) ... 275
2.10.4.2 Accostage et retrait avec des stratégies de retrait étendues (G460, G461, G462) ... 286
2.10.5 Activer/désactiver la surveillance de collision ("détection des goulets d'étranglement") (CDON, CDOF, CDOF2) : ... 290
2.10.6 Correction d'outil 2D ½ (CUT2D, CUT2DD, CUT2DF, CUT2DFD) ... 291
2.10.7 Maintenir constante la correction du rayon d'outil (CUTCONON, CUTCONOF) ... 294
2.10.8 Outils à position de tranchant définie ... 296

2.11 Modes de déplacement ... 299
2.11.1 Arrêt précis (G60, G9, G601, G602, G603) ... 299
2.11.2 Contournage (G64, G641, G642, G643, G644, G645, ADIS, ADISPOS) ... 301

2.12 Transformations des coordonnées (Frames) ... 311
2.12.1 Frames ... 311
2.12.2 Instructions de frame ... 313
2.12.3 Décalage d'origine programmable (TRANS, ATRANS) ... 316
2.12.4 Décalage d'origine programmable (G58, G59) ... 320
2.12.5 Rotation programmable (ROT, AROT, RPL) ... 322
2.12.6 Programmation de rotations de frames avec des angles solides (ROTS, AROTS, CROTS) ... 328
2.12.7 Facteur d'échelle programmable (SCALE, ASCALE) ... 331
2.12.8 Fonction miroir programmable (MIRROR, AMIRROR) ... 334
2.12.9 Génération d'un frame selon l'orientation de l'outil (TOFRAME, TOROT, PAROT) ... 339
2.12.10 Désactivation du frame (G53, G153, SUPA, G500) ... 342
2.12.11 Programmation : Désélection de corrections spécifique à un axe (CORROF) ... 342
2.12.12 Désélection de décalages d'origine additifs (DRFOF) ... 345
2.12.13 Décalages d'origine spécifiques à la rectification (GFRAME0, GFRAME1 ... GFRAME100) ... 347

2.13 Sorties de fonctions auxiliaires ... 348
2.13.1 Fonctions M ... 350

2.14 Instructions complémentaires ... 354
2.14.1 Transmission de messages (MSG) ... 354
2.14.2 Écrire la chaîne de caractères dans la variable OPI (WRTPR) ... 355
2.14.3 Limitation de la zone de travail ... 357
2.14.3.1 Limitation de la zone de travail dans le SCB (G25/G26, WALIMON, WALIMOF) ... 357
2.14.3.2 Limitation de la zone de travail dans le SCP/SCR (WALCS0 ... WALCS10) ... 360
2.14.4 Accostage du point de référence (G74) ... 363
2.14.5 Accostage d'un point fixe (G75) ... 364
2.14.6 Accostage d'une butée (FXS, FXST, FXSW) ... 368
2.14.7 Arrêt temporisé (G4) ... 373

2.14.8 Arrêt interne du prétraitement des blocs ... 375

2.15 Informations diverses ... 376
2.15.1 Axes ... 376
2.15.1.1 Axes principaux / Axes géométriques ... 376
2.15.1.2 Axes supplémentaires ... 378
2.15.1.3 Broche principale, broche maître ... 378
2.15.1.4 Axes machine ... 378
2.15.1.5 Axes de canal ... 379
2.15.1.6 Axes à interpolation ... 379
2.15.1.7 Axes de positionnement ... 379
2.15.1.8 axes synchrones ... 380
2.15.1.9 Axes de commande ... 381
2.15.1.10 Axes AP ... 381
2.15.1.11 les axes Link ... 381
2.15.1.12 Axes Lead-Link ... 383
2.15.2 De l'instruction de déplacement au déplacement de la machine ... 385
2.15.3 Calcul de la distance ... 385
2.15.4 Adresses ... 386
2.15.5 Noms ... 388
2.15.6 Constante ... 390
2.15.7 Opérateurs et fonctions de calcul ... 391

3 Notions complémentaires ... 395

3.1 Programmation CN flexible ... 395
3.1.1 Variables ... 395
3.1.1.1 Données système ... 396
3.1.1.2 Variables utilisateur prédéfinies : Paramètres de calcul spécifiques à un canal (R) ... 398
3.1.1.3 Variables utilisateur prédéfinies : Paramètres de calcul globaux (RG) ... 400
3.1.1.4 Variables utilisateur prédéfinies : variables Link ... 401
3.1.1.5 Définition de variables utilisateur (DEF) ... 403
3.1.1.6 Redéfinition de données système, données utilisateur et instructions de langage CN (REDEF) ... 409
3.1.1.7 Attribut : Valeur d'initialisation ... 413
3.1.1.8 Attribut : valeurs limites (LLI, ULI) ... 416
3.1.1.9 Attribut : unité physique (PHU) ... 417
3.1.1.10 Attribut : droits d'accès (APR, APW, APRP, APWP, APRB, APWB) ... 420
3.1.1.11 Vue d'ensemble des attributs définissables et redéfinissables ... 424
3.1.1.12 Définition et initialisation de variables de tableau (DEF, SET, REP) ... 426
3.1.1.13 Types de données ... 432
3.1.1.14 Minimum, maximum et plage de variables (MINVAL, MAXVAL, BOUND) ... 433
3.1.1.15 Vérification de l'existence d'une variable (ISVAR) ... 434
3.1.1.16 Lecture des valeurs d'attribut / du type de données (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP) ... 436
3.1.1.17 Conversions de types possibles ... 442
3.1.2 Programmation indirecte ... 443
3.1.2.1 Programmation indirecte d'adresses ... 443
3.1.2.2 Programmation indirecte d'instructions G ... 445
3.1.2.3 Programmation indirecte d'attributs de position (GP) ... 446
3.1.2.4 Programmation indirecte de lignes de programme pièce (EXECSTRING) ... 448
3.1.3 Opérations ... 450
3.1.3.1 Fonctions de calcul ... 450
3.1.3.2 Opérations relationnelles et opérations logiques ... 452

3.1.3.3 Priorité des opérations ... 454
3.1.3.4 Correction de la précision pour les erreurs relationnelles (TRUNC) ... 454
3.1.3.5 Arrondissement (ROUNDUP) ... 456
3.1.4 Opérations sur les chaînes de caractères ... 457
3.1.4.1 Conversion de type en type STRING (AXSTRING) ... 458
3.1.4.2 Conversion de type à partir de STRING (NUMBER, ISNUMBER, AXNAME) ... 458
3.1.4.3 Concaténation de chaînes de caractères (<<) ... 459
3.1.4.4 Conversion en minuscules/majuscules (TOLOWER, TOUPPER) ... 460
3.1.4.5 Déterminer la longueur d'une chaîne de caractères (STRLEN) ... 461
3.1.4.6 Recherche de caractères/chaînes de caractères dans une chaîne (INDEX, RINDEX, MINDEX, MATCH) ... 462
3.1.4.7 Sélection d'une chaîne de caractères partielle (SUBSTR) ... 463
3.1.4.8 Lecture et écriture de caractères individuels ... 464
3.1.4.9 Formatage d'une chaîne (SPRINT) ... 465
3.1.5 Sauts de programme ... 473
3.1.5.1 Retour au début du programme (GOTOS) ... 473
3.1.5.2 Sauts de programme sur repères de saut (IF, GOTOB, GOTOF, GOTO, GOTOC) ... 474
3.1.5.3 Saut de programme (CASE ... OF ... DEFAULT ...) ... 477
3.1.6 Répétition de sections de programme (REPEAT, REPEATB, ENDLABEL, P) ... 479
3.1.7 Structures de contrôle ... 484
3.1.7.1 Instruction conditionnelle et branchement (IF, ELSE, ENDIF) ... 486
3.1.7.2 Boucle de programme sans fin (LOOP, ENDLOOP) ... 488
3.1.7.3 Boucle de comptage (FOR ... TO ..., ENDFOR) ... 489
3.1.7.4 Boucle de programme avec condition en début de boucle (WHILE, ENDWHILE) ... 490
3.1.7.5 Boucle de programme avec condition en fin de boucle (REPEAT, UNTIL) ... 491
3.1.7.6 Exemple de programme avec des structures de contrôle imbriquées ... 491
3.1.8 Coordination de programmes dans tous les canaux (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) ... 492
3.1.9 Macroprogrammation (DEFINE ... AS) ... 498

3.2 Technique des sous-programmes ... 501
3.2.1 Généralités ... 501
3.2.1.1 Sous-programme ... 501
3.2.1.2 Nom de sous-programme ... 502
3.2.1.3 Imbrication de sous-programmes ... 503
3.2.1.4 Chemin de recherche ... 504
3.2.1.5 Paramètres formels et paramètres effectifs ... 505
3.2.1.6 Transfert de paramètres ... 505
3.2.2 Définition d'un sous-programme ... 507
3.2.2.1 Sous-programme sans transfert de paramètres ... 507
3.2.2.2 Sous-programme avec transfert de paramètres Call-by-Value (PROC) ... 508
3.2.2.3 Sous-programme avec transfert de paramètres Call-by-Reference (PROC, VAR) ... 510
3.2.2.4 Sauvegarde des fonctions G modales (SAVE) ... 512
3.2.2.5 Inhibition de l'exécution bloc par bloc (SBLOF, SBLON) ... 513
3.2.2.6 Inhibition de l'affichage du bloc courant (DISPLOF, DISPLON, ACTBLOCNO) ... 518
3.2.2.7 Identifier les sous-programmes avec prétraitement (PREPRO) ... 521
3.2.2.8 Retour de sous-programme M17 ... 522
3.2.2.9 Retour de sous-programme RET ... 523
3.2.2.10 Retour paramétrable dans les sous-programmes (RET ...) ... 524
3.2.2.11 Retour paramétrable dans les sous-programmes (RETB...) ... 530
3.2.3 Appel d'un sous-programme ... 534
3.2.3.1 Appel de sous-programme sans transfert de paramètres ... 534
3.2.3.2 Appel d'un sous-programme avec transfert de paramètres (EXTERN) ... 536

3.2.3.3 Nombre de répétitions de programme (P) ... 538
3.2.3.4 Appel modal d'un sous-programme (MCALL) ... 540
3.2.3.5 Appel indirect de sous-programme (CALL) ... 542
3.2.3.6 Appel indirect d'un sous-programme avec indication de la section de programme à exécuter (CALL BLOCK ... TO ...) ... 543
3.2.3.7 Appel indirect d'un programme programmé en langage ISO (ISOCALL) ... 544
3.2.3.8 Appel de sous-programme avec chemin d'accès et paramètres (PCALL) ... 545
3.2.3.9 Extension du chemin de recherche pour l'appel de sous-programmes (CALLPATH) ... 546
3.2.3.10 Exécution de sous-programme externe (EXTCALL) ... 547

3.3 Routine d'interruption (ASUP) ... 551
3.3.1 Fonction d'une routine d'interruption ... 551
3.3.2 Création d'une routine d'interruption ... 552
3.3.3 Affectation et démarrage d'une routine d'interruption (SETINT, PRIO, BLSYNC) ... 552
3.3.4 Désactivation/activation de l'affectation d'une routine d'interruption (DISABLE, ENABLE) ... 555
3.3.5 Suppression d'une affectation de routine d'interruption (CLRINT) ... 555
3.3.6 Relèvement rapide du contour (SETINT LIFTFAST, ALF) ... 556
3.3.7 Sens de retrait rapide du contour ... 558
3.3.8 Séquence de déplacement avec des routines d'interruption ... 561

3.4 Gestion des fichiers et programmes ... 563
3.4.1 Mémoire de programmes ... 563
3.4.1.1 Mémoire de programmes dans le NCK ... 563
3.4.1.2 Mémoires de programmes externes ... 565
3.4.1.3 Adressage des fichiers de la mémoire de programmes ... 567
3.4.1.4 Chemin de recherche pour l'appel de sous-programmes ... 571
3.4.1.5 Interrogation du chemin et du nom de fichier ... 573
3.4.2 Mémoire de travail (CHANDATA, COMPLETE, INITIAL) ... 574

3.5 Gestion de fichiers ... 578
3.5.1 Ecriture d'un fichier (WRITE) ... 578
3.5.2 Suppression d'un fichier (DELETE) ... 581
3.5.3 Lecture des lignes d'un fichier (READ) ... 582
3.5.4 Vérification de l'existence d'un fichier (ISFILE) ... 584
3.5.5 Lecture des informations fichier (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) ... 585

3.6 Zones de protection ... 588
3.6.1 Définition des zones de protection (CPROTDEF, NPROTDEF) ... 588
3.6.2 Activation/désactivation des zones de protection (CPROT, NPROT) ... 592
3.6.3 Contrôle des dépassements de zone de protection, des limitations de la zone de travail et des fins de course logicielles (CALCPOSI) ... 595

3.7 Instructions de déplacement spéciales ... 606
3.7.1 Accostage de positions codées (CAC, CIC, CDC, CACP, CACN) ... 606
3.7.2 Interpolation de type spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) ... 607
3.7.3 Groupe spline (SPLINEPATH) ... 617
3.7.4 Activer/désactiver le compactage de bloc CN (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF) ... 618
3.7.5 Interpolation polynomiale (POLY, POLYPATH, PO, PL) ... 619
3.7.6 Référence de trajectoire réglable (SPATH, UPATH) ... 624
3.7.7 Mesure spécifique au canal (MEAS, MEAW) ... 626
3.7.8 Mesure spécifique à l'axe (MEASA, MEAWA, MEAC) (option) ... 629
3.7.9 Fonctions spéciales pour l'utilisateur OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) ... 638

3.7.10 Réduction de l'avance avec décélération aux angles (FENDNORM, G62, G621) 639
3.7.11 Critère programmable de fin de déplacement (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA) 640

3.8 Transformations des coordonnées (frames) 643
3.8.1 Transformation de coordonnées par variables frames 643
3.8.1.1 Variable frame prédéfinie ($P_CHBFRAME, $P_IFRAME, $P_PFRAME, $P_ACTFRAME) 645
3.8.2 Affectations de valeurs aux frames 648
3.8.2.1 Affectation de valeurs directes (valeur d'axe, angle, échelle) 648
3.8.2.2 Lecture et modification de composantes de frame (TR, FI, RT, SC, MI) 650
3.8.2.3 Calcul avec des frames 652
3.8.2.4 Définition des variables frame (DEF FRAME) 653
3.8.3 Décalage grossier et fin (CTRANS, CFINE) 654
3.8.4 Décalage d'origine externe ($AA_ETRANS) 655
3.8.5 Forçage de valeurs réelles avec perte de l'état de référencement (PRESETON) 656
3.8.6 Forçage de valeurs réelles sans perte de l'état de référencement (PRESETONS) 658
3.8.7 Calcul du frame à partir de 3 points mesurés dans l'espace (MEAFRAME) 659
3.8.8 Frames à définition globale 663
3.8.8.1 Frames spécifiques à un canal ($P_CHBFR, $P_UBFR) 664
3.8.8.2 Frames actifs dans un canal 665

3.9 Transformations 670
3.9.1 Programmation générale des types de transformation 670
3.9.1.1 Mouvements d'orientation lors des transformations 672
3.9.1.2 Aperçu de la transformation d'orientation TRAORI 676
3.9.2 Transformation trois, quatre et cinq axes (TRAORI) 678
3.9.2.1 Corrélations générales de tête de fraisage de type cardan 678
3.9.2.2 Transformation trois, quatre et cinq axes (TRAORI) 681
3.9.2.3 Variantes de programmation d'orientation et position initiale (ORIRESET) 682
3.9.2.4 Programmation de l'orientation de l'outil (A..., B..., C..., LEAD, TILT) 684
3.9.2.5 Fraisage en bout (A4, B4, C4, A5, B5, C5) 690
3.9.2.6 Référence des axes d'orientation (ORIWKS, ORIMKS) 691
3.9.2.7 Programmation des axes d'orientation (ORIAXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) 693
3.9.2.8 Programmation de l'orientation le long d'une enveloppe de cône (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) 695
3.9.2.9 Définition de l'orientation de deux points de contact (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=) 698
3.9.3 Polynômes d'orientation (PO[angles], PO[coordonnée]) 700
3.9.4 Rotations de l'orientation de l'outil (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) 702
3.9.5 Orientations par rapport à la trajectoire 704
3.9.5.1 Types d'orientation par rapport à la trajectoire 704
3.9.5.2 Rotation de l'orientation de l'outil relative à la trajectoire (ORIPATH, ORIPATHS, angle de rotation) 706
3.9.5.3 Interpolation relative à la trajectoire de la rotation de l'outil (ORIROTC, THETA) 707
3.9.5.4 Lissage du tracé d'orientation (ORIPATHS A8=, B8=, C8=) 710
3.9.6 Compactage de l'orientation (COMPON, COMPCURV, COMPCAD, COMPSURF) 711
3.9.7 Activation/désactivation du lissage du tracé d'orientation (ORISON, ORISOF) 713
3.9.8 Transformation cinématique 714
3.9.8.1 Activation de la transformation de face frontale (TRANSMIT) 714
3.9.8.2 Activer la transformation de surfaces cylindriques (TRACYL) 715
3.9.8.3 Activation de la transformation à angle oblique avec angle programmable (TRAANG) 718
3.9.8.4 Plongée oblique sur les rectifieuses (G5, G7) 719

3.9.9 Activer la transformation concaténée (TRACON) ... 721
3.9.10 Déplacement PTP cartésien ... 722
3.9.10.1 Activer/désactiver le déplacement PTP cartésien (PTP, PTPG0, PTPWOC, CP) ... 722
3.9.10.2 Indiquer la position des articulations (STAT) ... 724
3.9.10.3 Indiquer les signes des angles d'axe (TU) ... 728
3.9.10.4 Exemple 1 : Déplacement PTP d'un robot à 6 axes avec la transformation ROBX ... 730
3.9.10.5 Exemple 2 : Déplacement PTP avec une transformation générique 5 axes ... 732
3.9.10.6 Exemple 3 : PTPG0 et TRANSMIT ... 732
3.9.11 Conditions marginales pour l'activation d'une transformation ... 734
3.9.12 Désactivation de la transformation (TRAFOOF) ... 735

3.10 Chaînes cinématiques ... 736
3.10.1 Suppression de composants (DELOBJ) ... 736
3.10.2 Détermination de l'indice par le nom (NAMETOINT) ... 740

3.11 Prévention des collisions avec chaînes cinématiques ... 742
3.11.1 Contrôle de la présence de paire de collision (COLLPAIR) ... 742
3.11.2 Demander un nouveau calcul du modèle de machine pour la prévention des collisions (PROTA) ... 743
3.11.3 Réglage de l'état des zone de protection (PROTS) ... 744
3.11.4 Détermination de la distance entre deux zones de protection (PROTD) ... 745

3.12 Transformations avec chaînes cinématiques ... 747
3.12.1 Activation de la transformation (TRAFOON) ... 747
3.12.2 Modification de la transformation d'orientation après mesurage de la machine (CORRTRAFO) ... 748

3.13 Corrections d'outils ... 756
3.13.1 Mémoire de correcteurs ... 756
3.13.2 Corrections additives ... 758
3.13.2.1 Sélection des corrections additives (DL) ... 758
3.13.2.2 Définition des valeurs d'usure et de réglage ($TC_SCPxy[t,d], $TC_ECPxy[t,d]) ... 760
3.13.2.3 Effacer les corrections additives (DELDL) ... 760
3.13.3 Correcteur d'outil : Interventions spéciales ... 761
3.13.3.1 Application de la fonction miroir aux longueurs d'outil ... 763
3.13.3.2 Exploitation du signe de l'usure ... 763
3.13.3.3 Système de coordonnées de l'usinage actif (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS) ... 764
3.13.3.4 Longueurs d'outil et changement de plan ... 767
3.13.4 Correction d'outil en ligne ... 768
3.13.4.1 Définir une fonction polynôme (FCTDEF) ... 768
3.13.4.2 Ecriture en continu de la correction d'outil en ligne (PUTFTOCF) ... 770
3.13.4.3 Ecriture discrète de la correction d'outil en ligne (PUTFTOC) ... 771
3.13.4.4 Activer/désactiver la correction d'outil en ligne (FTOCON/FTOCOF) ... 772
3.13.5 Correction du rayon d'outil 3D ... 772
3.13.5.1 Sélection de la correction 3D du rayon d'outil pour le fraisage périphérique 3D (CUT3DC, CUT3DCD, ISD) ... 772
3.13.5.2 Sélection de la correction 3D du rayon d'outil pour le fraisage en bout 3D (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD) ... 777
3.13.5.3 Fraisage périphérique 3D avec prise en compte d'une surface de délimitation (CUT3DCC, CUT3DCCD) ... 783
3.13.6 Orientation de l'outil (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) ... 788
3.13.7 Numéros D libres, numéro de tranchant ... 794
3.13.7.1 Numéros D libres, numéro de tranchant (adresse CE) ... 794

3.13.7.2 Libre affectation des numéros D : contrôle des numéros D (CHKDNO) 794
3.13.7.3 Libre affectation des numéros D : modification des numéros D (GETDNO, SETDNO) 795
3.13.7.4 Libre affectation des numéros D : détermination du numéro T correspondant au numéro D prescrit (GETACTTD) 796
3.13.7.5 Libre affectation des numéros D : numéros D déclarés non valides (DZERO) 796
3.13.8 Cinématique de l'organe porte-outil 797
3.13.9 Correction de longueur d'outil pour organes porte-outils orientables (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ) 802
3.13.10 Modification des organes porte-outil orientables après mesurage de la machine (CORRTC) 807
3.13.11 Correction en ligne de la longueur d'outil (TOFFON, TOFFOF) 811
3.13.12 Modification des données de correction pour outils orientables 813
3.13.12.1 Calcul de l'orientation (ORISOLH) 813
3.13.12.2 Activation de la modification des données de correction pour outils orientables (CUTMOD, CUTMODK) 824
3.13.13 Utilisation des environnements d'outils 832
3.13.13.1 Enregistrement de l'environnement d'outil (TOOLENV) 832
3.13.13.2 Suppression de l'environnement d'outil (DELTOOLENV) 835
3.13.13.3 Lecture des numéros T, D et DL (GETTENV) 836
3.13.13.4 Lecture des informations relatives aux environnements d'outil archivés ($P_TOOLENVN, ($P_TOOLENV) 837
3.13.13.5 Lecture de longueurs d'outil ou de composantes de longueur d'outil (GETTCOR) 837
3.13.13.6 Modification des composantes d'outil (SETTCOR) 844
3.13.14 Lecture de l'affectation des longueurs d'outil L1, L2, L3 par rapport aux axes de coordonnées (LENTOAX) 857

3.14 Modes de déplacement 860
3.14.1 Variation de l'avance (FNORM, FLIN, FCUB, FPO) 860
3.14.2 Comportement à l'accélération 865
3.14.2.1 Mode d'accélération (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA) 865
3.14.2.2 Influence de l'accélération dans le cas des axes asservis (VELOLIMA, ACCLIMA, JERKLIMA) 867
3.14.2.3 Activation de valeurs dynamiques spécifiques à la technologie (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH, DYNPREC) 869
3.14.3 Déplacement avec commande anticipatrice (FFWON, FFWOF) 870
3.14.4 Précision de contour programmable (CPRECON, CPRECOF) 871
3.14.5 Exécution du programme avec une mémoire tampon (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE) 873
3.14.6 Définition de sections d'arrêt temporisé (DELAYFSTON, DELAYFSTOF) 875
3.14.7 Inhibition d'une partie de programme pour SERUPRO (IPTRLOCK, IPTRUNLOCK) 878
3.14.8 Réaccostage du contour (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) 880
3.14.9 Correction du pilotage des déplacements 888
3.14.9.1 Adaptation de la vitesse d'axe ou de la vitesse de broche maximale (VELOLIM) 888
3.14.9.2 Adaptation de l'à-coup d'axe maximal (JERKLIM) 890
3.14.9.3 Adaptation de la vitesse tangentielle maximale (FLIM) 891
3.14.9.4 Adaptation de l'accélération maximale sur trajectoire (PACCLIM) 893
3.14.10 Programmation d'une tolérance de contour/orientation (CTOL, OTOL, ATOL) 894
3.14.11 Mode de changement de bloc pour couplage actif (CPBC) 899

3.15 Fonctions d'axe 900
3.15.1 Permutation d'axe, permutation de broche (RELEASE, GET, GETD) 900
3.15.2 Transmettre un axe à un autre canal (AXTOCHAN) 904
3.15.3 Fonctions d'axe (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) 905
3.15.4 Axes géométriques permutables (GEOAX) 907

3.15.5 Conteneur d'axes (AXCTSWE, AXCTSWED, AXCTSWEC) ... 912
3.15.6 Attente de la position d'axe définitive (WAITENC) ... 914
3.15.7 Commutation de jeu de paramètres programmable (SCPARA) ... 915
3.15.8 Activation/désactivation de l'adaptation (CADAPTON, CADAPTOF) ... 916
3.15.9 Adaptation du filtre d'à-coup FIR au mode dynamique (CALCFIR) ... 919

3.16 couplages d'axes ... 922
3.16.1 Déplacements conjugués (TRAILON, TRAILOF) ... 922
3.16.2 Tables de courbes (CTAB) ... 926
3.16.2.1 Définition de tables de courbes (CTABDEF, CATBEND) ... 927
3.16.2.2 Vérification de l'existence d'une table de courbes (CTABEXISTS) ... 933
3.16.2.3 Suppression d'une table de courbes (CTABDEL) ... 933
3.16.2.4 Verrouillage de tables de courbes contre la suppression et l'écrasement (CTABLOCK, CTABUNLOCK) ... 934
3.16.2.5 Tables de courbes : détermination des propriétés de la table (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD) ... 936
3.16.2.6 Lecture des valeurs de la table de courbes (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) ... 937
3.16.2.7 Tables de courbes : Vérification de l'utilisation des ressources (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) ... 942
3.16.3 Couplage de deux axes par valeur pilote (LEADON, LEADOF) ... 943
3.16.4 Réducteur électronique (EG) ... 948
3.16.4.1 Définition d'un réducteur électronique (EGDEF) ... 949
3.16.4.2 Activation du réducteur électronique (EGON, EGONSYN, EGONSYNE) ... 950
3.16.4.3 Désactivation du réducteur électronique (EGOFS, EGOFC) ... 953
3.16.4.4 Suppression de la définition d'un réducteur électronique (EGDEL) ... 953
3.16.4.5 Avance par tour (G95) / réducteur électronique (FPR) ... 954
3.16.5 Broche synchrone ... 954
3.16.5.1 Broche synchrone : Programmation (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) ... 955
3.16.6 Couplage générique (CP...) ... 965
3.16.7 Positionnement tangentiel ... 973
3.16.7.1 Définir le couplage (TANG) ... 973
3.16.7.2 Activer la génération de bloc intermédiaire (TLIFT) ... 975
3.16.7.3 Activer le couplage (TANGON) ... 976
3.16.7.4 Désactiver le couplage (TANGOF) ... 977
3.16.7.5 Supprimer un couplage (TANGDEL) ... 978
3.16.8 Couplage maître/esclave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS) ... 979

3.17 Actions synchrones ... 983
3.17.1 Définition d'une action synchrone ... 983

3.18 Oscillation ... 984
3.18.1 Oscillation asynchrone (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) ... 984
3.18.2 Oscillation commandée par des actions synchrones (OSCILL) ... 989

3.19 Poinçonnage et grignotage ... 997
3.19.1 Activation/désactivation ... 997
3.19.1.1 Activation/désactivation du poinçonnage et du grignotage (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) ... 997
3.19.2 Segmentation automatique des déplacements ... 1002
3.19.2.1 Segmentation des déplacements pour axes à interpolation ... 1005
3.19.2.2 Segmentation du déplacement dans le cas d'axes individuels ... 1006

3.20 Rectification 1008
3.20.1 Activer/désactiver la surveillance d'outil spécifique à la rectification (TMON, TMOF) 1008

3.21 Arrêt étendu et retrait (ESR) 1009
3.21.1 ESR assisté par CN 1010
3.21.1.1 Retrait assisté par CN (POLF, POLFA, POLFMASK, POLFMLIN) 1010
3.21.1.2 Arrêt assisté par CN 1014
3.21.2 ESR contrôlé par le variateur 1015
3.21.2.1 Configuration d'un arrêt contrôlé par le variateur (ESRS) 1015
3.21.2.2 Configuration d'un retrait contrôlé par le variateur (ESRR) 1015

3.22 Temps d'exécution du programme / Compteur de pièces 1017
3.22.1 Temps d'exécution de programme 1017
3.22.2 Compteurs de pièces 1020

3.23 Simulation de programme 1022
3.23.1 Fonction 1022
3.23.2 Simulation de programme associée aux cycles de compilation 1022

3.24 Autres fonctions 1024
3.24.1 Activation des paramètres machine (NEWCONF) 1024
3.24.2 Contrôler le langage CN existant (STRINGIS) 1025
3.24.3 Appel interactif des fenêtres depuis le programme pièce (MMC) 1028
3.24.4 Process DataShare - Sortie sur un appareil/fichier externe (EXTOPEN, WRITE, EXTCLOSE) .. 1033
3.24.5 Alarmes (SETAL) 1037
3.24.6 Définition d'une pièce brute (WORKPIECE) 1039
3.24.7 Basculement du mode de langage (G290, G291) 1043

3.25 Programmes de chariotage personnalisés 1046
3.25.1 Fonctions additionnelles pour le chariotage 1046
3.25.2 Création d'une table de contour (CONTPRON) 1046
3.25.3 Création d'une table de contour codée (CONTDCON) 1052
3.25.4 Détermination de l'intersection entre deux éléments de contour (INTERSEC) 1055
3.25.5 Exécution des éléments de contour d'une table bloc par bloc (EXECTAB) 1057
3.25.6 Calcul de données de cercles (CALCDAT) 1057
3.25.7 Désactivation de la préparation du contour (EXECUTE) 1059

3.26 Programmation externe de cycles 1061
3.26.1 Cycles technologiques 1061
3.26.1.1 Introduction 1061
3.26.1.2 Vue d'ensemble spécifique à la technologie 1062
3.26.1.3 HOLES1 - Modèle de positions Rangée 1064
3.26.1.4 HOLES2 - Modèle de positions Cercle complet ou partiel 1065
3.26.1.5 POCKET3 - Poche rectangulaire 1067
3.26.1.6 POCKET4 - Poche circulaire 1069
3.26.1.7 SLOT1 - Rainure rectiligne 1072
3.26.1.8 SLOT2 - Rainure sur cercle 1074
3.26.1.9 LONGHOLE - Trou oblong 1077
3.26.1.10 CYCLE60 - Gravure 1079
3.26.1.11 CYCLE61 - Surfaçage 1081
3.26.1.12 CYCLE62 - Appel de contour 1083
3.26.1.13 CYCLE63 - Fraisage de poche de contour / Matière restante du contour de poche / Fraisage du contour de tourillon / Matière restante du contour de tourillon 1084
3.26.1.14 CYCLE64 - Perçage ébauche d'une poche de contour 1087
3.26.1.15 CYCLE70 - Fraisage de filetage 1088

3.26.1.16 CYCLE72 - Fraisage en contournage ... 1090
3.26.1.17 CYCLE76 - Tourillon rectangulaire ... 1094
3.26.1.18 CYCLE77 - Tourillon circulaire ... 1096
3.26.1.19 CYCLE78 - Fraisage de filetages ... 1098
3.26.1.20 CYCLE79 - Polyèdre ... 1101
3.26.1.21 CYCLE81 - Perçage, centrage ... 1103
3.26.1.22 CYCLE82 - Perçage, lamage ... 1104
3.26.1.23 CYCLE83 - Perçage profond 1 ... 1106
3.26.1.24 CYCLE84 - Taraudage sans porte-taraud compensateur ... 1109
3.26.1.25 CYCLE85 - Alésage à l'alésoir ... 1112
3.26.1.26 CYCLE86 - Alésage au tour ... 1113
3.26.1.27 CYCLE92 - Tronçonnage ... 1114
3.26.1.28 CYCLE95 - Chariotage au contour ... 1116
3.26.1.29 CYCLE98 - Concaténation de filetages ... 1118
3.26.1.30 CYCLE99 - Filetage au tour ... 1122
3.26.1.31 CYCLE435 - Définir le système de coordonnées du dresseur ... 1127
3.26.1.32 CYCLE495 - Profilage ... 1127
3.26.1.33 CYCLE782 - Adapter à la charge ... 1129
3.26.1.34 CYCLE800 - Pivotement plan / Pivotement outil / Orientation outil ... 1132
3.26.1.35 CYCLE801 - Modèle de positions Réseau ou cadre ... 1136
3.26.1.36 CYCLE802 - Positions quelconques ... 1137
3.26.1.37 CYCLE806 - Tournage par interpolation ... 1140
3.26.1.38 CYCLE830 - Perçage profond 2 ... 1141
3.26.1.39 CYCLE832 - Réglages High Speed ... 1147
3.26.1.40 CYCLE840 - Taraudage avec porte-taraud compensateur ... 1151
3.26.1.41 CYCLE899 - Rainure borgne ... 1153
3.26.1.42 CYCLE930 - Gorge ... 1156
3.26.1.43 CYCLE940 - Dégagement forme E et F / Dégagement filet ... 1158
3.26.1.44 CYCLE951 - Chariotage ... 1161
3.26.1.45 CYCLE952 - Chariotage / Chariotage reste / Plongée / Plongée reste / Plongée G +D / Plongée G+D reste ... 1164
3.26.1.46 CYCLE4071 - Rectification longitudinale avec pénétration au point d'inversion ... 1170
3.26.1.47 CYCLE4072 - Rectification longitudinale avec pénétration au point d'inversion et signal d'abandon ... 1172
3.26.1.48 CYCLE4073 - Rectification longitudinale avec pénétration continue ... 1176
3.26.1.49 CYCLE4074 - Rectification longitudinale avec pénétration continue et signal d'abandon ... 1177
3.26.1.50 CYCLE4075 - Rectification plane avec pénétration au point d'inversion ... 1180
3.26.1.51 CYCLE4077 - Rectification plane avec pénétration au point d'inversion et signal d'abandon. 1183
3.26.1.52 CYCLE4078 - Rectification plane avec pénétration continue ... 1187
3.26.1.53 CYCLE4079 - Rectification plane avec pénétration intermittente ... 1189
3.26.1.54 GROUP_BEGIN - Début du bloc de programme ... 1192
3.26.1.55 GROUP_END - Fin du bloc de programme ... 1192
3.26.1.56 GROUP_ADDEND - Fin du supplément de mise au point ... 1193
3.26.1.57 Conditions marginales ... 1193
3.26.2 Vue d'ensemble des paramètres des cycles de mesure ... 1195
3.26.2.1 Paramètres des cycles de mesure CYCLE973 ... 1195
3.26.2.2 Paramètres des cycles de mesure CYCLE974 ... 1197
3.26.2.3 Paramètres des cycles de mesure CYCLE994 ... 1200
3.26.2.4 Paramètres des cycles de mesure CYCLE976 ... 1203
3.26.2.5 Paramètres des cycles de mesure CYCLE978 ... 1205
3.26.2.6 Paramètres des cycles de mesure CYCLE998 ... 1209
3.26.2.7 Paramètres des cycles de mesure CYCLE977 ... 1211

3.26.2.8 Paramètres des cycles de mesure CYCLE961 ... 1215
3.26.2.9 Paramètres des cycles de mesure CYCLE979 ... 1217
3.26.2.10 Paramètres de cycles de mesure CYCLE997 ... 1220
3.26.2.11 Paramètres du cycle de mesure CYCLE995 ... 1223
3.26.2.12 Paramètres des cycles de mesure CYCLE996 ... 1225
3.26.2.13 Paramètres de cycle de mesure CYCLE9960 ... 1228
3.26.2.14 Paramètres des cycles de mesure CYCLE982 ... 1230
3.26.2.15 Paramètres des cycles de mesure CYCLE971 ... 1233
3.26.2.16 Paramètres de cycle de mesure CYCLE150 ... 1236
3.26.3 Cycles de mesure ... 1237
3.26.3.1 Vue d'ensemble des paramètres des cycles de mesure ... 1237

4 Tableaux ... 1281

4.1 Instructions ... 1281

4.2 Adresses ... 1323
4.2.1 Lettres adresses ... 1323
4.2.2 Adresses fixes ... 1324
4.2.3 Adresses réglables ... 1329

4.3 Instructions G ... 1337
4.3.1 Instructions G ... 1337
4.3.2 Groupe G 1 : instructions de déplacement modales ... 1337
4.3.3 Groupe G 2 : instructions de déplacement non modales, arrêt temporisé ... 1338
4.3.4 Groupe G 3 : frame programmable, limitation de la zone de travail et programmation de pôle ... 1338
4.3.5 Groupe G 4 : FIFO ... 1339
4.3.6 Groupe G 6 : Sélection du plan ... 1339
4.3.7 Groupe G 7 : Correction du rayon d'outil ... 1340
4.3.8 Groupe G 8 : décalage d'origine réglable ... 1340
4.3.9 Groupe G 9 : Inhibition des corrections de frame et d'outil ... 1340
4.3.10 Groupe G 10 : arrêt précis - contournage ... 1341
4.3.11 Groupe G 11 : Arrêt précis à effet non modal ... 1341
4.3.12 Groupe G 12 : critères de changement de bloc pour arrêt précis (G60/G9) ... 1341
4.3.13 Groupe G 13 : cotes de la pièce en système anglo-saxon / métrique ... 1341
4.3.14 Groupe G 14 : cotes de la pièce en absolu/relatif ... 1342
4.3.15 Groupe G 15 : type d'avance ... 1342
4.3.16 Groupe G 16 : correction de l'avance au niveau des courbures concave et convexe ... 1343
4.3.17 Groupe G 17 : correction d'outil pour comportement d'accostage / de retrait ... 1343
4.3.18 Groupe G 18 : correction d'outil pour comportement aux angles ... 1343
4.3.19 Groupe G 19 : raccordement en début de courbe de type spline ... 1343
4.3.20 Groupe G 20 : raccordement en fin de courbe de type spline ... 1344
4.3.21 Groupe G 21 : profil d'accélération ... 1344
4.3.22 Groupe G 22 : type de correction d'outil ... 1344
4.3.23 Groupe G 23 : surveillance anticollision sur contours intérieurs ... 1345
4.3.24 Groupe G 24 : commande anticipatrice ... 1345
4.3.25 Groupe G 25 : référence pour l'orientation de l'outil ... 1345
4.3.26 Groupe G 26 : mode de réaccostage pour REPOS (modal) ... 1346
4.3.27 Groupe G 27 : correction d'outil pour changement d'orientation au niveau des angles saillants ... 1346
4.3.28 Groupe G 28 : limitation de la zone de travail ... 1346
4.3.29 Groupe G 29 : programmation du rayon/diamètre ... 1346
4.3.30 Groupe G 30 : compression du bloc CN ... 1347

4.3.31 Groupe G 31 : Instructions G OEM ... 1347
4.3.32 Groupe G 32 : Instructions G OEM ... 1348
4.3.33 Groupe G 33 : correction fine d'outil réglable ... 1348
4.3.34 Groupe G 34 : lissage pour orientation de l'outil ... 1348
4.3.35 Groupe G 35 : poinçonnage et grignotage ... 1349
4.3.36 Groupe G 36 : poinçonnage avec temporisation ... 1349
4.3.37 Groupe G 37 : profil d'avance ... 1349
4.3.38 Groupe G 38 : affectation des entrées/sorties rapides pour poinçonnage/grignotage ... 1349
4.3.39 Groupe G 39 : précision de contour programmable ... 1350
4.3.40 Groupe G 40 : correction constante du rayon d'outil ... 1350
4.3.41 Groupe G 41 : possibilité d'interruption du filetage ... 1350
4.3.42 Groupe G 42 : organe porte-outil ... 1350
4.3.43 Groupe G 43 : sens d'accostage ARD ... 1351
4.3.44 Groupe G 44 : segmentation des déplacements ARD ... 1351
4.3.45 Groupe G 45 : référence de trajectoire des axes FGROUP ... 1351
4.3.46 Groupe G 46 : choix du plan pour le relèvement rapide ... 1351
4.3.47 Groupe G 47 : commutation de mode pour code CN externe ... 1352
4.3.48 Groupe G 48 : comportement d'accostage/retrait pour la correction du rayon d'outil ... 1352
4.3.49 Groupe G 49 : déplacement point à point ... 1352
4.3.50 Groupe G 50 : programmation de l'orientation ... 1352
4.3.51 Groupe G 51 : programmation de l'orientation du mode d'interpolation ... 1353
4.3.52 Groupe G 52 : rotation de frame par rapport à la pièce ... 1353
4.3.53 Groupe G 53 : rotation de frame par rapport à l'outil ... 1354
4.3.54 Groupe G 54 : rotation vectorielle pour la programmation polynomiale ... 1354
4.3.55 Groupe G 55 : déplacement en rapide avec/sans interpolation linéaire ... 1354
4.3.56 Groupe G 56 : prise en compte du calcul de l'usure de l'outil ... 1355
4.3.57 Groupe G 57 : décélération aux angles ... 1355
4.3.58 Groupe G 59 : mode dynamique pour interpolation de trajectoire ... 1355
4.3.59 Groupe G 60 : limitation de la zone de travail ... 1356
4.3.60 Groupe G 61 : lissage pour orientation de l'outil ... 1356
4.3.61 Groupe G 62 : mode de réaccostage pour REPOS (non modal) ... 1356
4.3.62 Groupe G 64 : frames de rectification ... 1357
4.4 Procédures prédéfinies ... 1358
4.5 Procédures prédéfinies dans les actions synchrones ... 1380
4.6 Fonctions prédéfinies ... 1382
4.7 Langue actuelle dans l'IHM ... 1396
A Annexe ... 1397
A.1 Liste des abréviations ... 1397
A.2 Vue d'ensemble de la documentation ... 1402
Index ... 1403

Consignes de sécurité élémentaires

1

1.1 Consignes de sécurité générales

⚠ ATTENTION

Le non respect des consignes de sécurité et le manque de prise en compte des risques résiduels peuvent entraîner la mort

Le non respect des consignes de sécurité et des remarques relatives aux risques résiduels dans la documentation du matériel peut conduire à des accidents susceptibles d'entraîner la mort ou de causer des blessures graves.

- Respecter les consignes de sécurité figurant dans la documentation du matériel.
- Tenir compte des risques résiduels pour l'évaluation des risques.

⚠ ATTENTION

Danger de mort lié à des dysfonctionnements de la machine suite à un paramétrage incorrect ou modifié

Un paramétrage incorrect ou modifié peut entraîner des dysfonctionnements sur les machines, susceptibles de provoquer des blessures, voire la mort.

- Protéger le paramétrage contre l'accès non autorisé.
- Prendre les mesures appropriées pour palier aux défauts éventuels (p. ex. un arrêt ou une coupure d'urgence).

1.2 Garantie et responsabilité pour les exemples d'application

Les exemples d'application sont sans engagement et n'ont aucune prétention d'exhaustivité concernant la configuration, les équipements et les éventualités de toutes sortes. Les exemples d'application ne constituent pas des solutions client spécifiques, mais ont uniquement pour objet d'apporter une aide dans la résolution de problèmes typiques.

L'utilisateur est seul responsable de la mise en œuvre des produits selon les règles de l'art. Les exemples d'application ne vous dispensent pas des obligations de précaution lors de l'utilisation, de l'installation, de l'exploitation et de la maintenance.

1.3 Note relative à la sécurité

Siemens commercialise des produits et solutions comprenant des fonctions de sécurité industrielle qui contribuent à une exploitation sûre des installations, systèmes, machines et réseaux.

Pour garantir la sécurité des installations, systèmes, machines et réseaux contre les cybermenaces, il est nécessaire de mettre en œuvre - et de maintenir en permanence - un concept de sécurité industrielle global et de pointe. Les produits et solutions de Siemens constituent une partie de ce concept.

Il incombe aux clients d'empêcher tout accès non autorisé à ses installations, systèmes, machines et réseaux. Ces systèmes, machines et composants doivent uniquement être connectés au réseau d'entreprise ou à Internet si et dans la mesure où cela est nécessaire et seulement si des mesures de protection adéquates (ex: pare-feu et/ou segmentation du réseau) ont été prises.

Pour plus d'informations sur les mesures de protection pouvant être mises en œuvre dans le domaine de la sécurité industrielle, rendez-vous sur https://www.siemens.com/industrialsecurity (https://www.siemens.com/industrialsecurity).

Les produits et solutions Siemens font l'objet de développements continus pour être encore plus sûrs. Siemens recommande vivement d'effectuer des mises à jour dès que celles-ci sont disponibles et d'utiliser la dernière version des produits. L'utilisation de versions qui ne sont plus prises en charge et la non-application des dernières mises à jour peut augmenter le risque de cybermenaces pour nos clients.

Pour être informé des mises à jour produit, abonnez-vous au flux RSS Siemens Industrial Security à l'adresse suivante:
https://www.siemens.com/industrialsecurity (https://new.siemens.com/global/en/products/services/cert.html#Subscriptions)

Plus d'informations, voir sur Internet :

Manuel de configuration Industrial Security (https://support.industry.siemens.com/cs/ww/fr/view/108862708/en)

⚠ ATTENTION

États de fonctionnement non sûrs suite à une manipulation du logiciel

Les manipulations des logiciels, p. ex. les virus, chevaux de Troie ou vers, peuvent provoquer des états de fonctionnement non sûrs de l'installation, susceptibles de causer la mort, des blessures graves et des dommages matériels.

- Les logiciels doivent être maintenus à jour.
- Intégrer les composants d'entraînement et d'automatisation dans un concept global de sûreté industrielle (Industrial Security) de l'installation ou de la machine selon l'état actuel de la technique.
- Tenir compte de tous les produits utilisés dans le système global de sûreté industrielle (Industrial Security).
- Il convient de protéger les données stockées sur les supports de mémoire amovibles contre les logiciels nuisibles avec les mesures de protection appropriées, par exemple avec un antivirus.
- Contrôler à l'issue de la mise en service toutes les fonctions relatives à la sécurité.

Notions de base

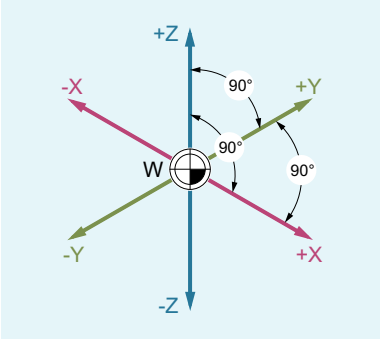
2

2.1 Notions géométriques de base

2.1.1 Positions des pièces

2.1.1.1 Système de référence des indications de position

Pour permettre à la machine, c'est-à-dire à la commande, d'exploiter les positions indiquées dans le programme CN, celles-ci doivent se rapporter à un système de référence qui soit applicable aux axes de déplacement de la machine. On utilise ainsi, pour les machines-outils, des systèmes de coordonnées orthogonaux de sens direct (coordonnées cartésiennes) conformes à la norme DIN 66217 comme système de coordonnées pièce.



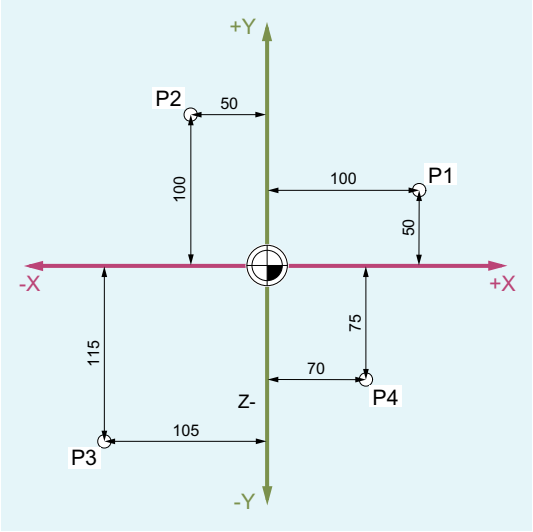
L'origine pièce (W) se confond avec l'origine du système de coordonnées pièce.

2.1.1.2 Coordonnées cartésiennes

Les axes du système de coordonnées sont cotés. Ainsi, il est possible de définir de manière univoque chaque point du système de coordonnées, et donc chaque position de pièce, par la direction (X, Y et Z) et trois valeurs numériques. L'origine pièce a toujours les coordonnées X0, Y0 et Z0.

Indications de position sous forme de coordonnées cartésiennes

Pour des raisons de simplification, nous considérerons, dans l'exemple suivant, un seul plan du système de coordonnées, à savoir le plan X/Y :

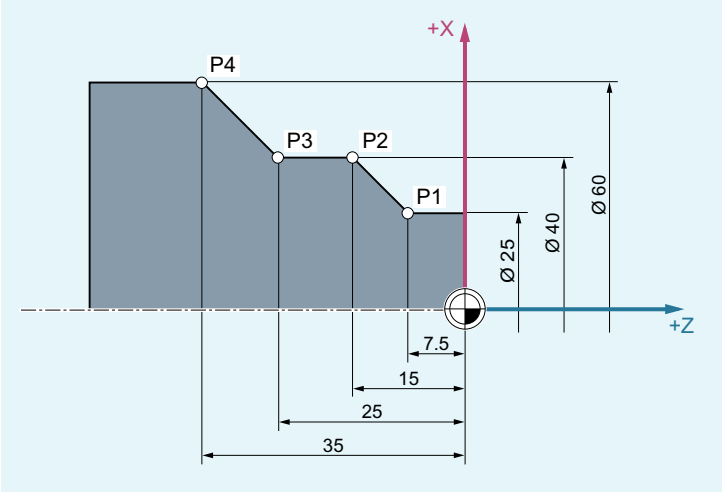


Les points P1 à P4 ont les coordonnées suivantes :

Position	Coordonnées
P1	X100 Y50
P2	X-50 Y100
P3	X-105 Y-115
P4	X70 Y-75

Exemple : Positions des pièces lors du tournage

Dans le cas des tours, un seul plan suffit pour définir le contour :

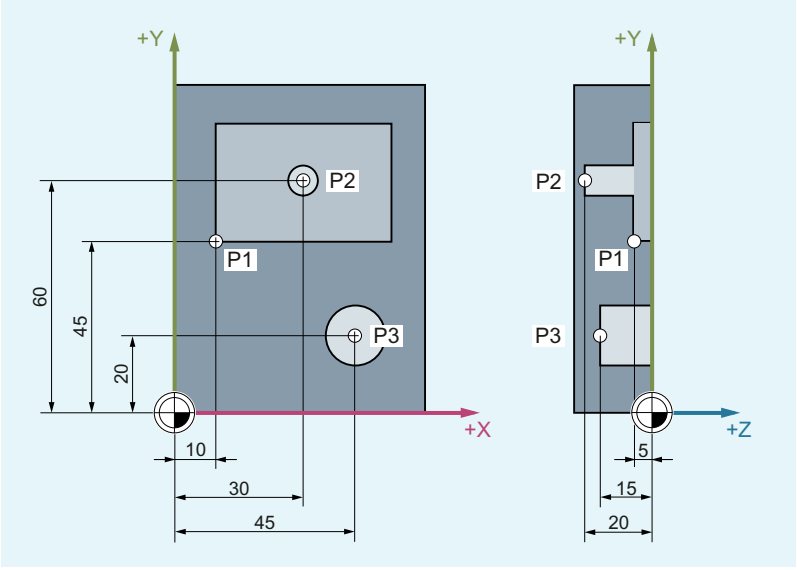


Les points P1 à P4 ont les coordonnées suivantes :

Position	Coordonnées
P1	X25 Z-7.5
P2	X40 Z-15
P3	X40 Z-25
P4	X60 Z-35

Exemple : Positions des pièces lors du fraisage

Pour les opérations de fraisage, la profondeur de passe doit également être décrite, autrement dit il est également nécessaire d'affecter une valeur numérique à la troisième coordonnée (Z dans ce cas).



Les points P1 à P3 ont les coordonnées suivantes :

Position	Coordonnées
P1	X10 Y45 Z-5
P2	X30 Y60 Z-20
P3	X45 Y20 Z-15

2.1.1.3 Coordonnées polaires

Pour la description des positions de pièce, il est également possible d'utiliser des coordonnées polaires au lieu des coordonnées cartésiennes. Ceci est intéressant lorsqu'une pièce ou une partie de pièce est cotée avec des rayons et des angles. Le point qui sert d'origine pour la cotation s'appelle "pôle".

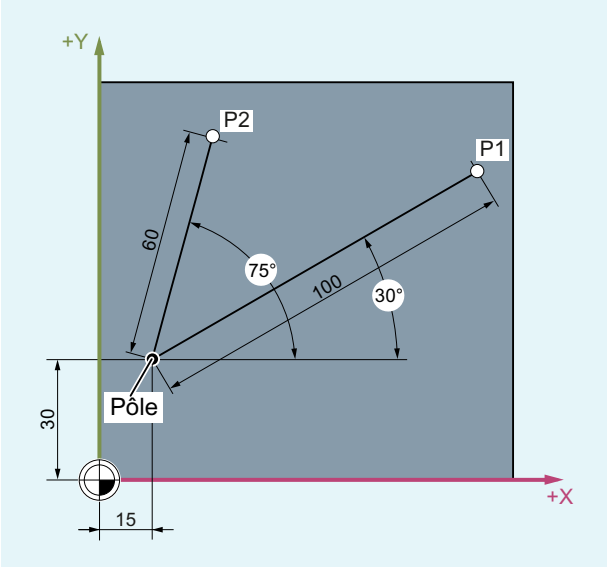
Indication des positions sous forme de coordonnées polaires

Les coordonnées polaires sont constituées du **rayon polaire** et de l'**angle polaire**.

Le rayon polaire correspond à la distance entre le pôle et la position.

L'angle polaire est l'angle compris entre le rayon polaire et l'axe horizontal du plan de travail. Un angle polaire négatif est défini dans le sens horaire et un angle positif dans le sens antihoraire.

Exemple



La description des points P1 et P2 peut se faire de la façon suivante, par rapport au pôle :

Position	Coordonnées polaires
P1	RP=100 AP=30
P2	RP=60 AP=75
RP : Rayon polaire AP : Angle polaire	

2.1.1.4 Cote absolue

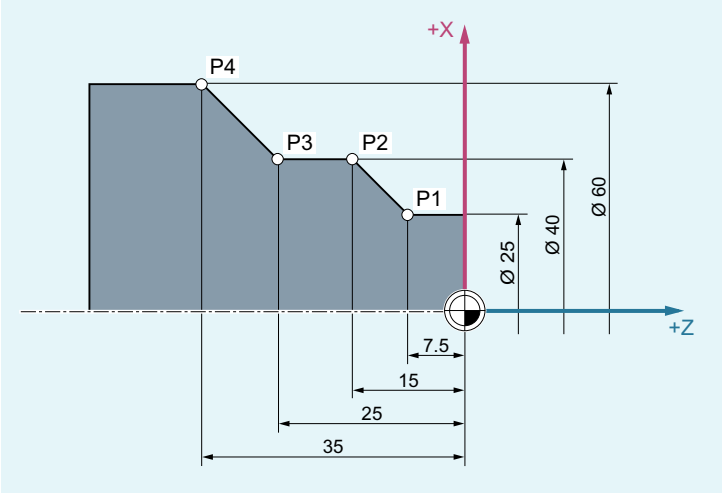
Indication des positions en cotes absolues

Dans le cas de la programmation en cotes absolues, toutes les indications de position se rapportent à l'origine du système de coordonnées.

Pour le déplacement de l'outil, cela signifie que :

La cote absolue décrit la position que doit atteindre l'outil.

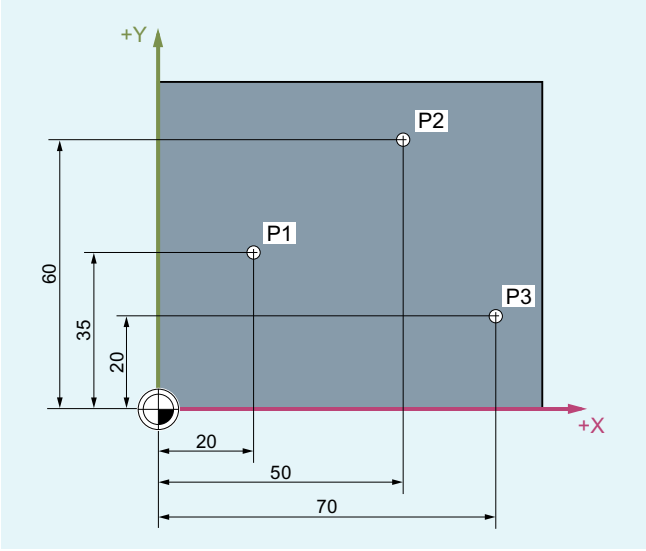
Exemple : Tournage



En cotes absolues, les positions des points P1 à P4 sont les suivantes :

Position	Indication des positions en cotes absolues
P1	X25 Z-7,5
P2	X40 Z-15
P3	X40 Z-25
P4	X60 Z-35

Exemple : Fraisage



En cotes absolues, les positions des points P1 à P3 sont les suivantes :

Position	Indication des positions en cotes absolues
P1	X20 Y35
P2	X50 Y60
P3	X70 Y20

2.1.1.5 Cote relative

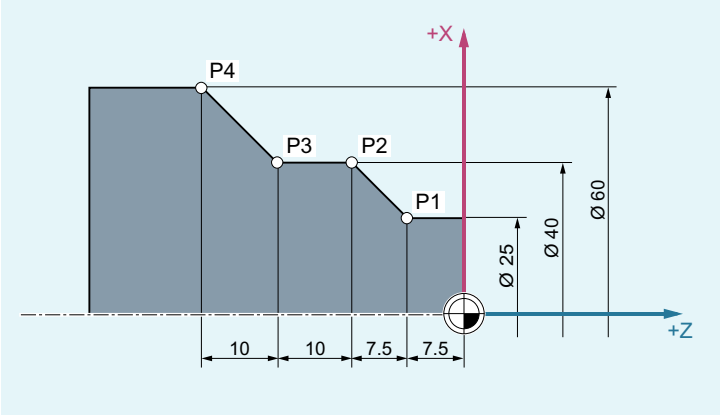
Indications de position en cotes relatives

Sur les dessins d'exécution, il est fréquent que la cote ne se rapporte pas à l'origine, mais à un autre point de la pièce. Afin de ne pas être obligé de recalculer ces cotes, il est possible d'utiliser les cotes relatives, dites aussi cotes incrémentales. Dans ce type de cote, une indication de position se rapporte au point précédent.

Pour le déplacement de l'outil, cela signifie que :

L'indication de cote relative décrit la valeur du déplacement que doit réaliser l'outil.

Exemple : Tournage



En cotes relatives, les indications de position des points P2 à P4 sont les suivantes :

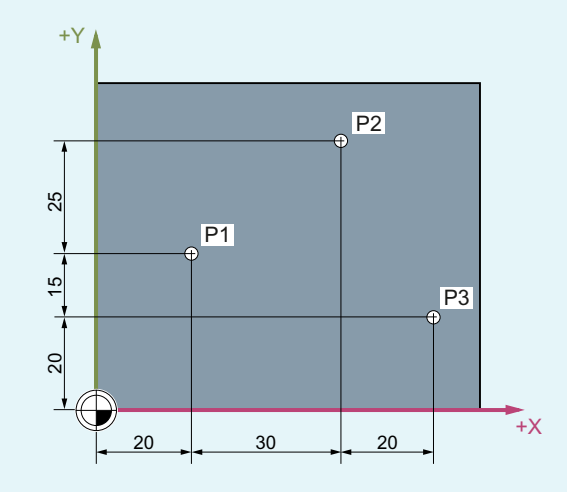
Position	Indication de position en cote relative	L'indication se rapporte à :
P2	X15 Z-7,5	P1
P3	Z-10	P2
P4	X20 Z-10	P3

Remarque

En présence de DIAMOF ou de DIAM90 (Page 166), la programmation de la consigne de déplacement pour l'indication de cotes relatives (G91) s'effectue au rayon.

Exemple : Fraisage

Les indications de position pour les points P1 à P3 sont, en cotes relatives :



En cotes relatives, les indications de position pour les points P1 à P3 sont les suivantes :

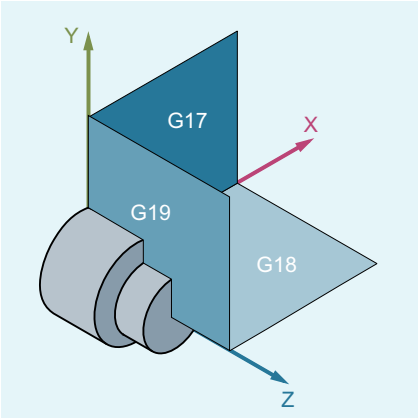
Position	Indication de position en cote relative	L'indication se rapporte à :
P1	X20 Y35	Point d'origine
P2	X30 Y20	P1
P3	X20 Y-35	P2

2.1.2 Plans de travail

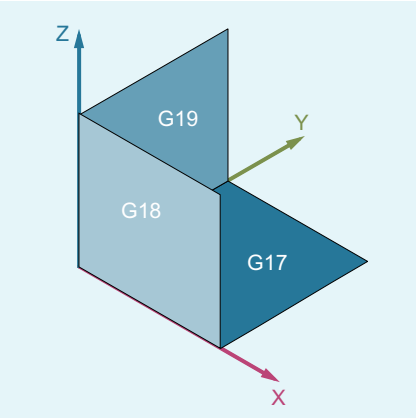
Un programme CN requiert l'indication du plan dans lequel l'usinage est effectué. C'est ce qui permet à la commande de prendre correctement en compte les valeurs de correction d'outil, par exemple. De plus, l'indication du plan de travail est requise pour certains types de programmation de cercle et dans le cas de coordonnées polaires.

Le plan de travail est défini par deux axes de coordonnées dans le système de coordonnées cartésiennes pièce de base. Le troisième axe de coordonnées est perpendiculaire à ce plan de travail et détermine le sens d'approche de l'outil (par exemple, pour usinage 2D).

Plans de travail pour le tournage / fraisage



Plans de travail pour le tournage



Plans de travail pour le fraisage

Activation d'un plan de travail

Dans le programme CN, les plans de travail sont activés au moyen des instructions G G17, G18 et G19. La corrélation est définie comme suit :

Instruction G	Plan de travail	Abscisse	Ordonnée	Cote ≙ Direction de pénétration
G17	X / Y	X	Y	Z
G18	Z / X	Z	X	Y
G19	Y / Z	Y	Z	X

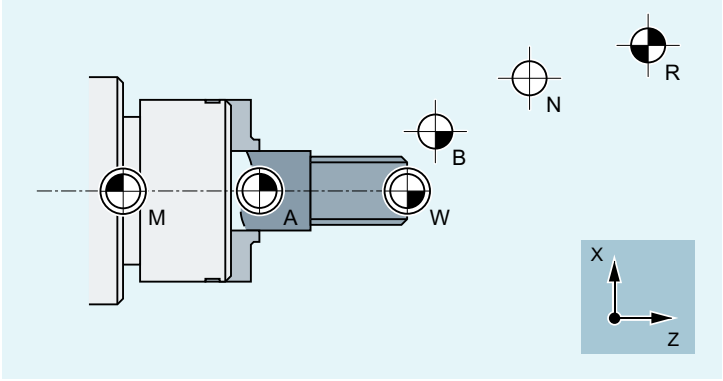
2.1.3 Origines et points de référence

Différentes origines et différents points de référence sont définis sur la machine-outil à CN :

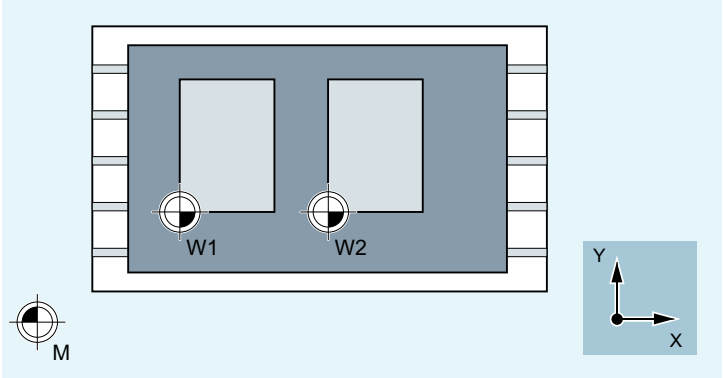
Origines		
	M	Origine machine L'origine machine sert à définir le système de coordonnées machine (SCM). Tous les autres points de référence se rapportent à l'origine machine.
	W	Origine pièce = Origine programme L'origine pièce définit le système de coordonnées pièce par rapport à l'origine machine.
	A	Point de butée Le point de butée peut se confondre avec l'origine pièce (sur les tours uniquement).

Points de référence		
	R	Point de référence Position définie par came et système de mesure. La distance à l'origine de la machine **M** doit être connu afin que la position de l'axe puisse être positionnée exactement sur cette valeur.
	B	Point de départ A définir pour chaque programme. Le premier outil commence ici l'usinage.
	T	Point de référence du porte-outil Le point de référence du porte-outil se trouve sur le logement du porte-outil. A partir des longueurs d'outil saisies, la commande calcule la distance entre la pointe de l'outil et le point de référence du porte-outil.
	N	Point de changement d'outil

Origines et points de référence pour le tournage



Origines pour le fraisage



2.1.4 Systèmes de coordonnées

On distingue les systèmes de coordonnées suivants :

- Système de coordonnées machine (SCM) (Page 36) avec l'origine machine **M**
- Système de coordonnées de base (SCB) (Page 38)
- Système de coordonnées de base décalé (SCD) (Page 39)
- Système de coordonnées réglable (SCR) (Page 40)
- Système de coordonnées pièce (SCP) (Page 41) avec l'origine pièce **W**

2.1.4.1 Système de coordonnées machine (SCM)

Le système de coordonnées machine est formé par tous les axes machine physiquement présents.

Dans le système de coordonnées machine sont définis des points de référence, des points de changement d'outil et de palette (points fixes machine).

Lors de la programmation directe dans le système de coordonnées machine (c'est possible pour quelques instructions G), les axes physiques de la machine entrent en action directement. L'existence d'un éventuel dispositif d'ablocage de pièce n'est pas prise en compte dans ce cas.

Remarque

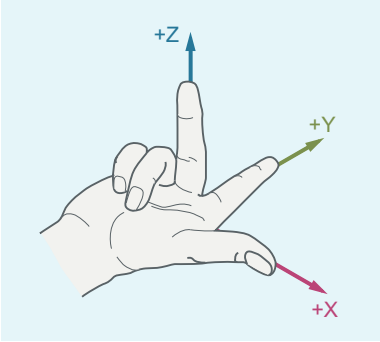
S'il existe différents systèmes de coordonnées machine (par exemple transformation 5 axes), la cinématique de la machine est reproduite, par le biais d'une transformation interne, sur le système de coordonnées dans lequel est effectuée la programmation.

Règle des trois doigts

L'orientation relative du système de coordonnées sur la machine dépend du type de machine. L'orientation des axes correspond à la "règle des trois doigts" de la main **droite** (selon DIN 66217).

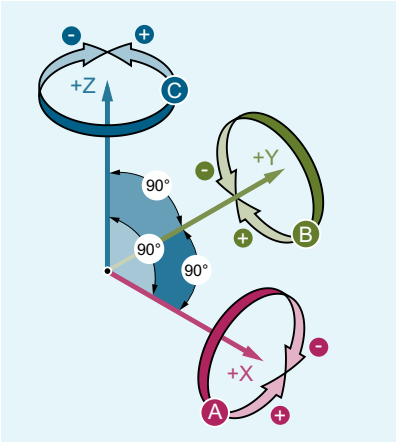
Si l'on se place devant la machine, le majeur de la main droite pointé dans le sens opposé à l'axe de pénétration de la broche principale, on a alors :

- le pouce dans le sens +X
- l'index dans le sens +Y
- le majeur dans le sens +Z



Les mouvements de rotation autour des axes de coordonnées X, Y et Z sont définis par A, B et C. Le sens de rotation est déduit du sens du mouvement de rotation en regardant dans le sens positif de l'axe de coordonnées :

Sens du mouvement de rotation	Sens de rotation
Dans le sens horaire	positif
Sens antihoraire (= sens trigonométrique)	négatif

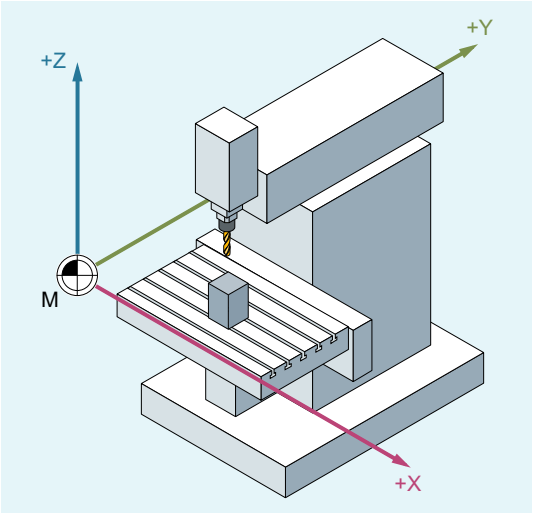


X, Y, Z Axes de coordonnées orthogonaux

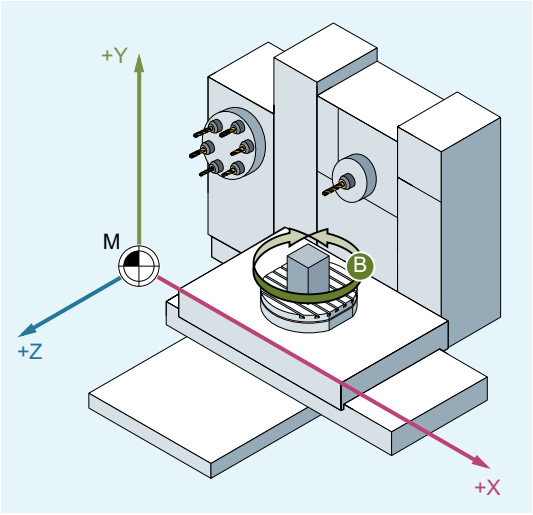
A, B, C Axes rotatifs autour de X, Y, Z

Position du système de coordonnées sur différents types de machine

La position du système de coordonnées, qui résulte de la "règle des trois doigts", peut varier en fonction du type de machine, comme le montrent les deux exemples suivants :



Fraiseuse verticale à 3 axes



Fraiseuse horizontale à 4 axes

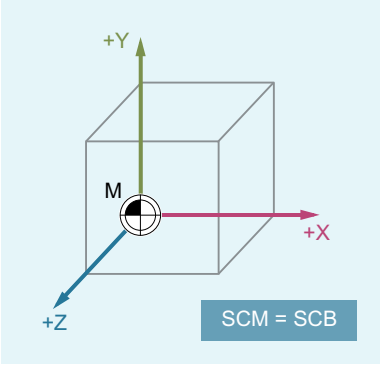
2.1.4.2 Système de coordonnées de base (SCB)

Le système de coordonnées de base (SCB) est composé de trois axes perpendiculaires (axes géométriques) et d'axes supplémentaires sans rapport géométrique.

Machines-outils sans transformation cinématique

Le SCB et le SCM se confondent toujours lorsque le SCB peut être reproduit sur le SCM sans transformation cinématique (p. ex. une transformation 5 axes, TRANSMIT / TRACYL / TRAANG).

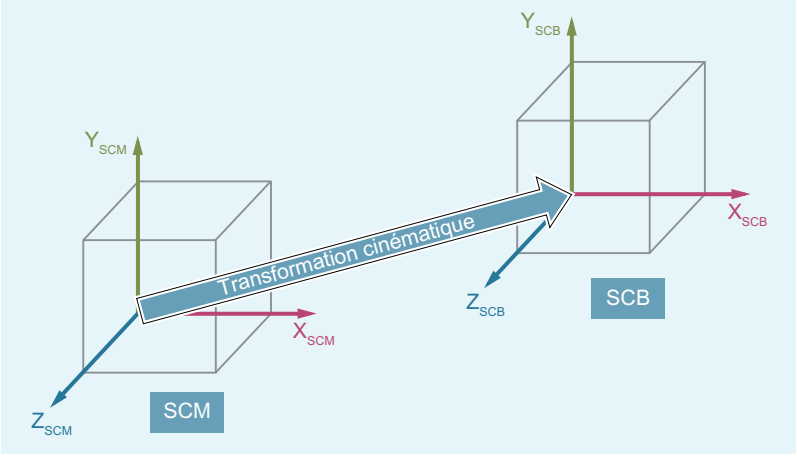
Sur ces machines, les noms des axes machine et des axes géométriques peuvent être identiques.



Machines-outils avec transformation cinématique

Le SCB et le SCM ne se confondent pas lorsque le SCB est reproduit sur le SCM par transformation cinématique (p. ex. une transformation 5 axes, TRANSMIT / TRACYL / TRAANG).

Sur ces machines, les noms des axes machine et des axes géométriques doivent être différents.

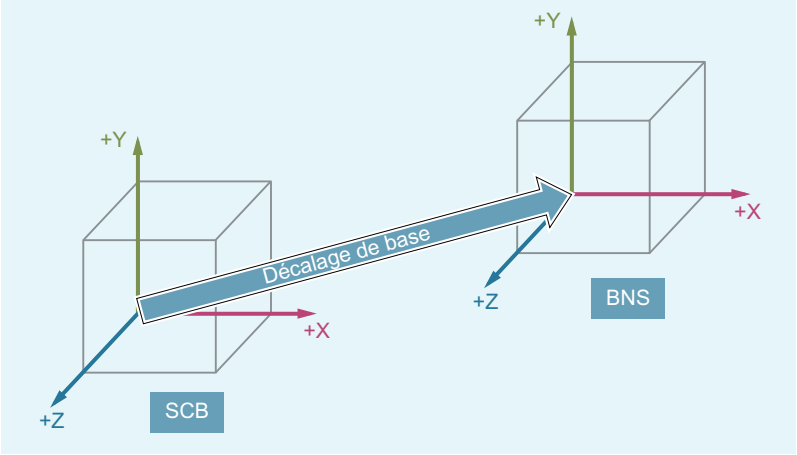


Cinématique de machine

La pièce est toujours programmée dans un système de coordonnées cartésiennes à deux ou à trois dimensions (SCP). Pour l'usinage de ces pièces, on utilise cependant de plus en plus souvent des machines-outils avec des axes rotatifs ou des axes linéaires qui ne sont pas cartésiens. Pour la représentation des coordonnées (cartésiennes) programmées dans le SCP en mouvements d'axe machine réels, on utilise la transformation cinématique.

2.1.4.3 Système de coordonnées de base décalé (SCD)

Le système de coordonnées décalé de base (SCD) résulte du système de coordonnées de base et du décalage de base.



Décalage de base

Le décalage de base décrit la transformation des coordonnées entre le SCB et le SCD. Elle permet, par exemple, de définir l'origine palette.

Le décalage de base se compose comme suit :

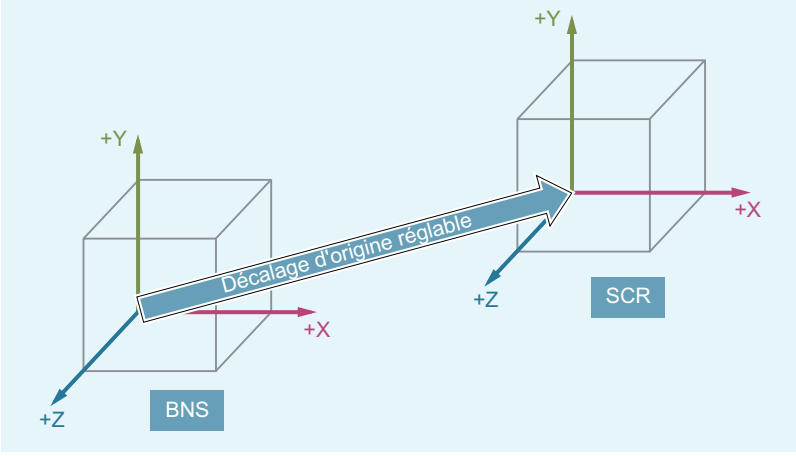
- Décalage d'origine externe
- Décalage DRF
- Déplacement forcé
- Frames système concaténés
- Frames de base concaténés

2.1.4.4 Système de coordonnées réglable (SCR)

Décalage d'origine réglable

Le décalage d'origine réglable permet d'obtenir le "système de coordonnées décalé réglable" (SCR) à partir du système de coordonnées décalé de base (SCD).

Dans le programme CN, les décalages d'origine réglables sont activés au moyen des instructions G G54...G57 et G505...G599.



Si aucune transformation de coordonnées programmable (frame) n'est active, le "système de coordonnées réglable" est le système de coordonnées pièce (SCP).

Transformations de coordonnées programmables (frames)

Dans certains cas, il peut s'avérer préférable, voire nécessaire, de procéder à l'intérieur d'un programme CN à un déplacement ou à une rotation du système de coordonnées pièce (ou du "système de coordonnées décalé réglable") choisi initialement, voire d'appliquer une fonction miroir ou un changement d'échelle. Ceci s'effectue par des transformations de coordonnées (frames) (Page 311) programmables.

Remarque

Les transformations de coordonnées programmables (frames) se rapportent toujours au "système de coordonnées décalé réglable".

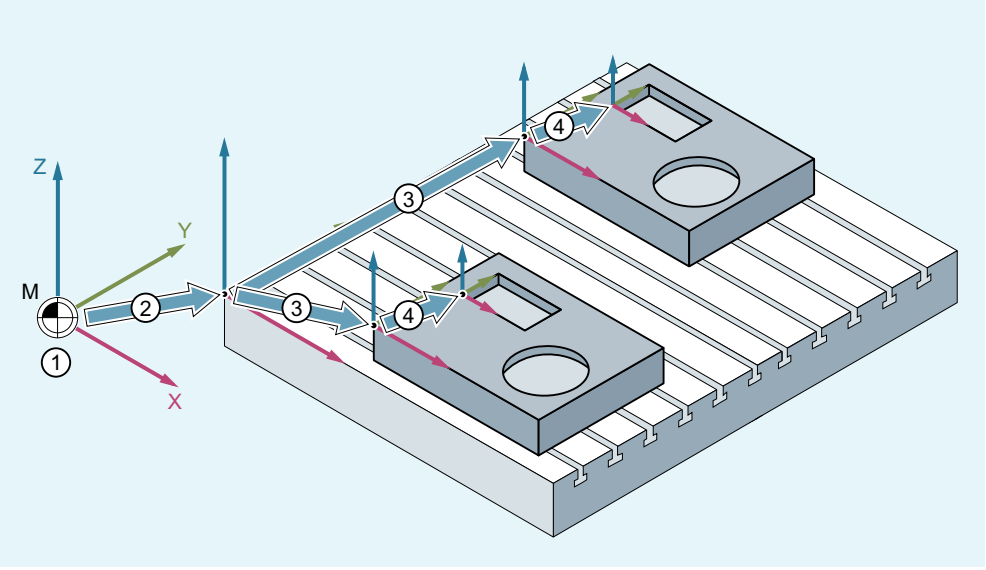
2.1.4.5 Système de coordonnées pièce (SCP)

La géométrie d'une pièce est décrite dans le système de coordonnées pièce (SCP). Autrement dit : les indications contenues dans le programme CN se rapportent au système de coordonnées pièce.

Le système de coordonnées pièce est toujours un système de coordonnées cartésiennes et toujours affecté à une pièce bien définie.

2.1.4.6 Quels sont les rapports entre les différents systèmes de coordonnées ?

L'exemple suivant illustre de manière explicite les rapports entre les différents systèmes de coordonnées :



① Aucune transformation cinématique n'est active. Cela signifie que le système de coordonnées machine et le système de coordonnées de base sont confondus.

② Le système de coordonnées de base décalé (SCD), avec l'origine palette, résulte du décalage de base.

③ Le décalage d'origine réglable G54 (ou G55) définit le "système de coordonnées réglable" (SCR) de la pièce 1 (respectivement de la pièce 2).

④ Le système de coordonnées pièce (SCP) est obtenu par une transformation de coordonnées programmable.

2.2 Bases de la programmation de la commande numérique

Remarque

DIN 66025 est la directive relative à la programmation CN.

2.2.1 Dénomination d'un programme CN

Règles

À chaque programme CN doit être affecté un nom de programme (descripteur) lors de sa création. Le nom du programme peut être choisi librement dans la mesure où les règles suivantes sont respectées :

- Caractères autorisés :
  - Caractères alphabétiques : A ... Z, a ... z
  - Chiffres : 0 ... 9
  - Tiret bas : **(_)**
- Les deux premiers caractères doivent être deux lettres, ou bien un tiret bas suivi d'une lettre.

  Remarque

  Si cette condition est remplie, un programme CN peut être appelé, par simple indication du nom du programme, comme sous-programme depuis un autre programme. En revanche, si le nom du programme commence par des chiffres, l'appel en tant que sous-programme est possible uniquement avec l'instruction CALL.

- Longueur maximale : 24 caractères

Remarque

Majuscules/minuscules

Dans le langage CN SINUMERIK, **aucune** distinction n'est faite entre majuscules et minuscules.

Remarque

Noms de sous-programme interdits

Pour éviter tout problème avec les applications Windows, les noms de programme suivants ne doivent **pas** être utilisés :

- CON, PRN, AUX, NUL
- COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9
- LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8, LPT9

Pour plus de restrictions, voir "Noms (Page 388)".

Extensions internes à la commande

Un préfixe et un postfixe sont ajoutés en interne à la commande comme extensions au nom de programme, attribué lors la création de celui-ci :

- Préfixe : _N_
- Postfixe :
  - Programmes principaux :_MPF
  - Sous-programmes : _SPF

Fichier en format de bande perforée

Les fichiers programmes créés en externe et devant être chargés via l'interface V.24 doivent être disponibles en format de bande perforée.

Les règles complémentaires suivantes s'appliquent pour le nom de fichier programme en format de bande perforée :

- Premier caractère : %
- Pour finir, une extension de fichier de quatre caractères : _xxx

Exemples :

- %_N_ARBRE123_MPF
- %Bride3_MPF

2.2.2 Structure et contenu d'un programme CN

2.2.2.1 Blocs et composantes de blocs

Blocs

Un programme CN se compose d'une suite de blocs CN. Chaque bloc contient les données pour l'exécution d'une opération d'usinage.

Composantes de bloc

Les blocs CN sont formés des composantes suivantes :

- Instructions selon DIN 66025
- Eléments du langage évolué CN

Instructions selon DIN 66025

Les instructions selon DIN 66025 sont constitués d'un symbole d'adresse et d'un chiffre ou d'une suite de chiffres qui décrit une valeur arithmétique.

Symbole d'adresse (adresse)

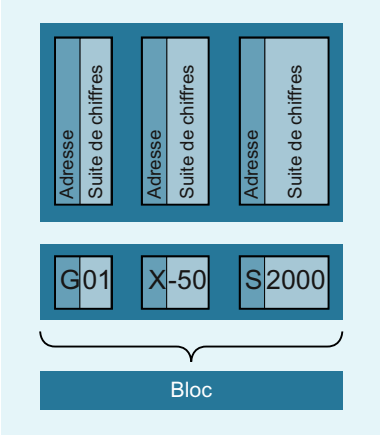
Le symbole d'adresse (généralement une lettre alphabétique) définit la signification de l'instruction.

Exemples :

Symbole d'adresse	Signification
G	Instruction G (fonction préparatoire)
X	Information de déplacement pour l'axe X
S	Vitesse de rotation de broche

Suite de chiffres

La suite de chiffres est la valeur affectée au symbole d'adresse. La suite de chiffres peut contenir un signe et un point décimal, le signe étant toujours placé entre la lettre adresse et la suite de chiffres. Les signes positifs (+) et les zéros de tête (0) n'ont pas besoin d'être écrits.



Eléments du langage évolué CN

Le jeu d'instructions selon DIN 66025 ne suffisant plus à la programmation des opérations d'usinage complexes des machines-outils modernes, il a été complété par les éléments du langage évolué CN.

Il s'agit, entre autres, des éléments suivants :

- Instructions du langage évolué CN
  Contrairement aux instructions selon DIN 66025, les instructions du langage évolué CN se composent de plusieurs lettres d'adresse. Par Exemple :
  - `OVR` pour la correction de vitesse de rotation
  - `SPOS` pour le positionnement de broche
- Descripteurs (noms définis) pour :
  - Variables système
  - Variables définies par l'utilisateur
  - Sous-programmes
  - Mots-clés
  - Marques de saut
  - Macros

  Remarque

  Un descripteur doit être univoque et ne pas être utilisé pour plusieurs objets.
- Opérateurs relationnels
- Opérateurs logiques
- Fonctions de calcul
- Structures de contrôle

Prise d'effet des instructions

Les instructions peuvent être avec prise d'effet modale ou non modale :

- modale
  Les instructions à effet modal restent valides avec la valeur programmée au-delà du bloc dans lequel elles sont programmées, jusqu'à :
  - la programmation d'une nouvelle valeur sous la même instruction
  - la programmation d'une instruction qui annule la validité de l'instruction qui était valide jusque-là
- Non modal
  Les instructions à effet non modal sont valides uniquement dans le bloc dans lequel elles ont été programmées.

Fin du programme

Le dernier bloc des séquences d'exécution contient un mot spécifique pour la fin du programme : `M2`, `M17` ou `M30`.

2.2.2.2 Règles valables pour les blocs

Début de bloc

Les blocs CN peuvent être identifiés en début de bloc par des numéros de bloc composés du caractère "N" et d'un nombre entier positif, comme par exemple :
`N40 ...`

Les numéros de blocs peuvent être définis dans un ordre quelconque, mais une numérotation en ordre croissant est fortement recommandée.

Remarque

A l'intérieur d'un programme, les numéros de blocs doivent être définis sans ambiguïté pour que la recherche de bloc puisse être efficace.

Fin de bloc

Un bloc se termine par le caractère LF (LINE FEED = nouvelle ligne).

Remarque

Le caractère LF ne doit pas être écrit. Il est automatiquement généré par le retour à la ligne.

Longueur de bloc

Un bloc peut contenir jusqu'à **512 caractères** (commentaire et caractère de fin de bloc LF compris).

Remarque

En général, l'écran affiche trois blocs à raison de 66 caractères au maximum par bloc. Les commentaires sont affichés également. Les messages apparaissent dans une fenêtre qui leur est propre.

Ordre des instructions

Afin d'obtenir une structure de bloc claire, il est conseillé de placer les instructions d'un bloc dans l'ordre suivant :
`N... G... X... Y... Z... F... S... T... D... M... H...`

Adresse	Signification
`N`	Adresse du numéro de bloc
`G`	Fonction préparatoire
`X,Y,Z`	Information de déplacement
`F`	Avance
`S`	Vitesse de rotation

T	Outil
D	Numéro de correction d'outil
M	Fonction supplémentaire
H	Fonction auxiliaire

Remarque

Quelques adresses peuvent être utilisées plusieurs fois au sein d'un même bloc, comme par exemple :

G..., M..., H...

2.2.2.3 Affectation de valeurs

On peut affecter des valeurs aux adresses. Les règles applicables sont les suivantes :

- Il faut écrire le caractère "=" entre l'adresse et la valeur qui suit si :
  - l'adresse se compose de plusieurs lettres,
  - la valeur se compose de plusieurs constantes.

  On peut omettre le signe "=" si l'adresse est constitué d'une seule lettre et la valeur à affecter d'une seule constante.
- Les signes sont autorisés.
- Les caractères de séparation sont permis après les lettres de l'adresse.

Exemples :

X10	Affectation d'une valeur (10) à l'adresse X, "=" n'est pas requis
X1=10	Affectation d'une valeur (10) à l'adresse (X) avec extension numérique (1), "=" requis
X=10*(5+SIN(37.5))	Affectation d'une valeur par le biais d'une expression numérique, "=" requis

Remarque

Une extension numérique doit toujours être suivie d'un des caractères spéciaux "=", "(", "[", ")", "]", "," ou d'un opérateur, pour faire la distinction entre l'adresse avec extension numérique et une lettre adresse avec valeur.

2.2.2.4 Commentaires

Pour qu'un programme CN soit plus compréhensible, il est possible d'ajouter des commentaires aux blocs CN.

Un commentaire se situe à la fin d'un bloc et est séparé de la partie programme du bloc CN par un point virgule (";").

Exemple 1 :

Code de programme	Commentaire
N10 G1 F100 X10 Y20	; commentaire explicatif du bloc CN

Exemple 2 :

Code de programme	Commentaire
N10	; société G&S, commande n° 12A71
N20	; programme créé par H. Müller, service TV 4, le 21/11/94
N50	; pièce n° 12, boîtier pour pompe plongeante type TP23A

Remarque

Les commentaires sont mémorisés et apparaissent à l'écran au moment du traitement du bloc en question.

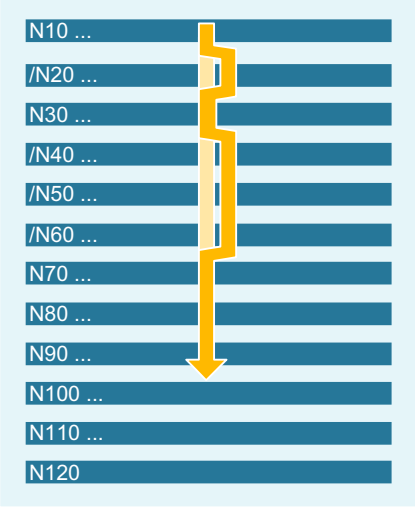
2.2.2.5 Masquage de blocs

Pour le traitement, il est possible de masquer les blocs CN qui ne doivent pas être exécutés à chaque passage du programme. Cette fonction peut, par exemple, être appliquée lors du test ou de la mise au point de nouveaux programmes.

Blocs optionnels

Les blocs à masquer sont marqués avec le caractère "/" (trait oblique) placé avant le numéro de bloc. Il est aussi possible de masquer plusieurs blocs consécutifs. Les instructions figurant dans les blocs masqués ne sont pas exécutées, le programme se poursuit avec le bloc suivant non masqué.

Exemple :



Code de programme	Commentaire
N10 …	; en cours de traitement
/N20 …	; masqué
N30 …	; en cours de traitement
/N40 …	; masqué
/N50 …	; masqué
/N60 …	; masqué
N70 …	; en cours de traitement
...	

Niveaux d'inhibition

Les blocs peuvent être affectés à des niveaux d'inhibition (10 au maximum), activables via l'interface utilisateur ou le programme utilisateur AP.

L'affectation s'effectue dans le programme CN via l'insertion d'une barre oblique suivie du numéro du niveau d'inhibition. Un seul niveau d'inhibition peut être indiqué par bloc.

Exemple :

Code de programme	Commentaire
/ ...	; le bloc est sauté (1er niveau d'inhibition)
/0 ...	; le bloc est sauté (1er niveau d'inhibition)
/1 N010...	; le bloc est sauté (2e niveau d'inhibition)
/2 N020...	; le bloc est sauté (3e niveau d'inhibition)
...	
/7 N100...	; le bloc est sauté (8e niveau d'inhibition)
/8 N080...	; le bloc est sauté (9e niveau d'inhibition)
/9 N090...	; le bloc est sauté (10e niveau d'inhibition)

Remarque

Les modifications de niveaux à inhiber ne sont possibles qu'à l'état Stop/Reset de la commande.

Remarque

Le nombre de niveaux d'inhibition utilisables dépend d'un PM de visualisation.

Remarque

L'inhibition des blocs est aussi active lors de la recherche de bloc.

Remarque

Le déroulement d'un programme peut également être modifié en procédant à des sauts conditionnels à l'aide de variables système et utilisateur.

2.3 Création d'un programme CN

2.3.1 Procédure générale

Lors de la réalisation d'un programme CN, la programmation, à savoir la transposition des opérations élémentaires d'usinage en langage CN, ne représente souvent qu'une petite partie du travail de programmation.

Avant toute programmation proprement dite, il est important de planifier et de préparer méticuleusement les opérations d'usinage. Plus votre préparation aura été précise quant à la structure de votre programme CN, plus la programmation proprement dite sera simple et rapide et moins vous aurez d'erreurs dans le programme terminé. Des programmes clairs s'avéreront par ailleurs d'autant plus avantageux le jour où des modifications devront être entreprises.

Une pièce ne ressemblant pas à une autre, il n'est pas judicieux de créer chaque programme selon la même méthode. Mais la procédure suivante sera appropriée dans la plupart des cas.

Procédure

1. **Préparer le dessin de la pièce**
   - Définir l'origine de la pièce
   - Indiquer le système de coordonnées
   - Eventuellement calculer les coordonnées manquantes.
2. **Définir le déroulement des opérations d'usinage**
   - Quels sont les outils à mettre en œuvre, à quel moment et pour le traitement de quel contour ?
   - Dans quel ordre les différents éléments de la pièce devront-ils être usinés ?
   - Quels sont les éléments qui se répètent (éventuellement pivotés) et qui devraient figurer dans un sous-programme ?
   - Est-ce que d'autres programmes pièce ou sous-programmes contiennent des contours susceptibles d'être utilisés pour la pièce actuelle ?
   - Quels sont les cas où des décalages d'origine, des rotations, des fonctions miroir ou des agrandissements / réductions sont justifiés, voire nécessaires (concept FRAME) ?

3. **Définir la gamme de fabrication**
   Définir pas à pas toutes les phases d'opération de la machine, par exemple :
   - Déplacements à vitesse rapide pour le positionnement
   - Changement d'outil
   - Définition du plan d'usinage
   - Dégagement pour les mesures
   - Mise en marche / arrêt de la broche, de l'arrosage
   - Appel des données d'outil
   - Approche de l'outil
   - Correction de trajectoire
   - Accostage du contour
   - Retrait de l'outil
   - etc.
4. **Traduire les opérations dans le langage de programmation**
   - Transcrire chaque opération sous la forme d'un bloc CN (ou de blocs CN).
5. **Regrouper toutes les opérations en un programme**

2.3.2 Caractères disponibles

Pour l'écriture de programmes CN , on dispose des caractères suivants :

- Majuscules :
  A, B, C, D, E, F, G, H, I, J, K, L, M, N,(O),P, Q, R, S, T, U, V, W, X, Y, Z
- Minuscules :
  a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z
- Chiffres :
  0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Caractères spéciaux :
  voir le tableau suivant

Caractère spécial	Signification
%	Caractère de début de programme (uniquement pour programmation sur PC externe)
(	Mise entre parenthèses de paramètres ou dans des expressions
)	Mise entre parenthèses de paramètres ou dans des expressions
[	Mise entre crochets d'adresses ou d'indices de tableau
]	Mise entre crochets d'adresses ou d'indices de tableau
<	inférieur à
>	supérieur à
:	Bloc principal, fin d'étiquette, opérateur de concaténation

Caractère spécial	Signification
=	Affectation, partie d'une égalité
/	Division, saut de bloc optionnel
*	Multiplication
+	Addition
-	Soustraction, signe négatif
"	Guillemet, identificateur de chaîne de caractères
'	Apostrophe, identificateur de valeurs numériques particulières : hexadécimales, binaires
$	Identificateur de variables propres au système
s_	Trait de soulignement, fait partie des lettres
?	Réservé
!	Réservé
.	Point décimal
,	Virgule, séparateur de paramètres
;	Début de commentaire
&	Caractère de mise en forme, effet identique à un caractère d'espacement
LF	Fin de bloc
Tabulateur	Séparateur
Espace	Séparateur (blanc)

Remarque

Ne pas confondre la lettre "O" avec le chiffre "0".

Remarque

Aucune différence n'est faite entre les minuscules et les majuscules (exception : appel d'outil).

Remarque

Les caractères non représentables sont traités comme des caractères d'espacement.

2.3.3 En-tête du programme

Les blocs CN qui précèdent les blocs de déplacement destinés à la réalisation du contour de la pièce sont désignés comme en-tête du programme.

L'en-tête du programme contient des informations et des instructions concernant les points suivants :

- Changement d'outil
- Corrections d'outil
- Mouvement de broche

- Régulation d'avance
- Paramètres géométriques (décalage d'origine, sélection du plan de travail)

En-tête de programme de tournage

L'exemple suivant montre la structure typique d'un en-tête de programme CN de tournage :

Code de programme	**Commentaire**
N10 G0 G153 X200 Z500 T0 D0	; Retirer l'organe porte-outil avant le pivotement de la tourelle porte-outil.
N20 T5	; Mettre en position l'outil 5.
N30 D1	; Activer le bloc de données du tranchant de l'outil.
N40 G96 S300 LIMS=3000 M4 M8	; Vitesse de coupe constante (Vc) = 300 m/min, limitation de la vitesse de rotation = 3000 tr/min, sens de rotation antihoraire, arrosage en marche.
N50 DIAMON	; Axe X programmé sur le diamètre.
N60 G54 G18 G0 X82 Z0.2	; Appeler le décalage d'origine et le plan de travail, accoster la position de départ.
...	

En-tête de programme de fraisage

L'exemple suivant montre la structure typique d'un en-tête de programme CN de fraisage :

Code de programme	**Commentaire**
N10 T="SF12"	; Alternative : T123
N20 M6	; Déclencher un changement d'outil
N30 D1	; Activer le bloc de données du tranchant de l'outil
N40 G54 G17	; Décalage d'origine et plan de travail
N50 G0 X0 Y0 Z2 S2000 M3 M8	; Mouvement d'accostage de la pièce, broche et arrosage en marche
...	

Lorsque l'on travaille avec une orientation de l'outil ou des transformations de coordonnées, il est recommandé de supprimer les transformations qui pourraient encore être actives au début du programme.

Code de programme	**Commentaire**
N10 CYCLE800()	; Réinitialisation du plan orienté
N20 TRAFOOF	; Réinitialisation de TRAORI, TRANSMIT, TRACYL...
...	

2.3.4 Exemples de programmes

2.3.4.1 Exemple 1 : Premières opérations de programmation

L'exemple de programme 1 doit permettre d'effectuer les premières opérations de programmation et de les tester.

Marche à suivre

1. Créer un programme pièce (nom)
2. Éditer le programme pièce
3. Sélectionner le programme pièce
4. Activer le bloc par bloc.
5. Démarrer le programme pièce

Remarque

Pour que le programme soit exécutable sur la machine, des paramètres machine doivent être définis en conséquence (→ voir les indications du constructeur de la machine !).

Remarque

Lors du test d'un programme, des alarmes peuvent apparaître. Avant toute chose, elles doivent être réinitialisées.

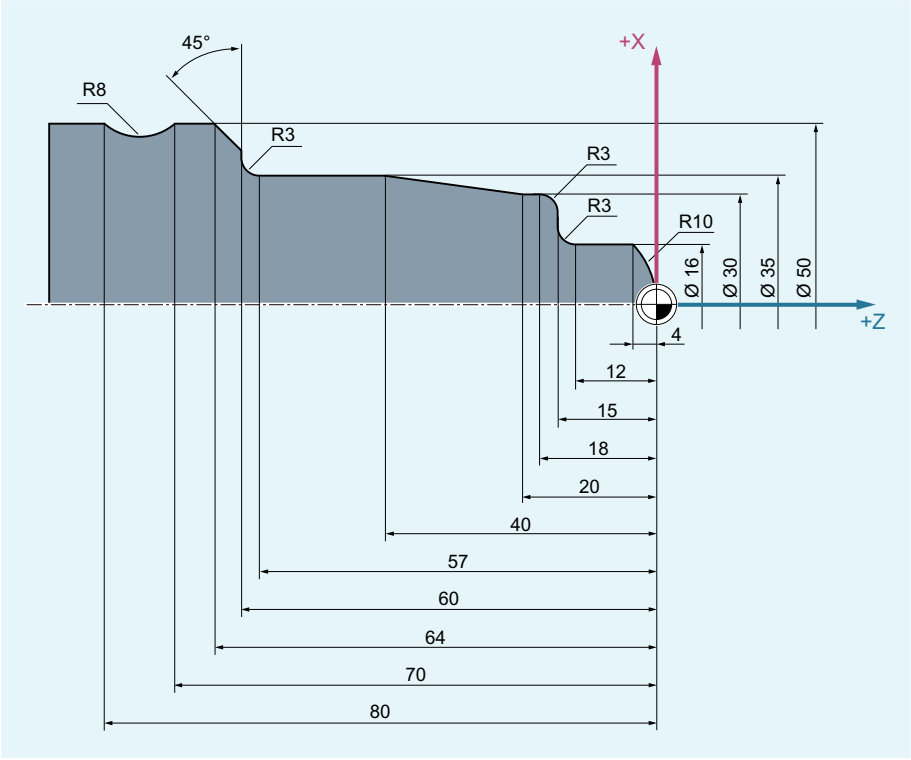
Programme CN

Code de programme	**Commentaire**
`N10 MSG("CECI EST MON PROGRAMME CN")`	`; Affichage du message "CECI EST MON PROGRAMME CN" dans la ligne d'alarme`
`N20 F200 S900 T1 D2 M3`	`; Avance, broche, outil, correction d'outil, broche sens horaire`
`N30 G0 X100 Y100`	`; Accoster position en vitesse rapide`
`N40 G1 X150`	`; Rectangle avec avance, droite en X`
`N50 Y120`	`; Droite en Y`
`N60 X100`	`; Droite en X`
`N70 Y100`	`; Droite en Y`
`N80 G0 X0 Y0`	`; Retrait en vitesse rapide`
`N100 M30`	`; Fin de bloc`

2.3.4.2 Exemple 2 : Programme CN de tournage

L'exemple de programme 2 est prévu pour l'usinage d'une pièce sur un tour. Il contient une programmation radiale et une correction du rayon d'outil.

Dessin coté de la pièce



Programme CN

Code de programme	Commentaire
N5 G0 G53 X280 Z380 D0	; Point de départ
N10 TRANS X0 Z250	; Décalage d'origine
N15 LIMS=4000	; Limitation de la vitesse de rotation (G96)
N20 G96 S250 M3	; Sélection de la vitesse de coupe constante
N25 G90 T1 D1 M8	; Sélection de l'outil et du correcteur
N30 G0 G42 X-1.5 Z1	; Activation de la correction du rayon d'outil et approche de l'outil
N35 G1 X0 Z0 F0.25	
N40 G3 X16 Z-4 I0 K-10	; Tournage du rayon 10
N45 G1 Z-12	
N50 G2 X22 Z-15 CR=3	; Tournage du rayon 3
N55 G1 X24	
N60 G3 X30 Z-18 I0 K-3	; Tournage du rayon 3
N65 G1 Z-20	
N70 X35 Z-40	
N75 Z-57	
N80 G2 X41 Z-60 CR=3	; Tournage du rayon 3
N85 G1 X46	

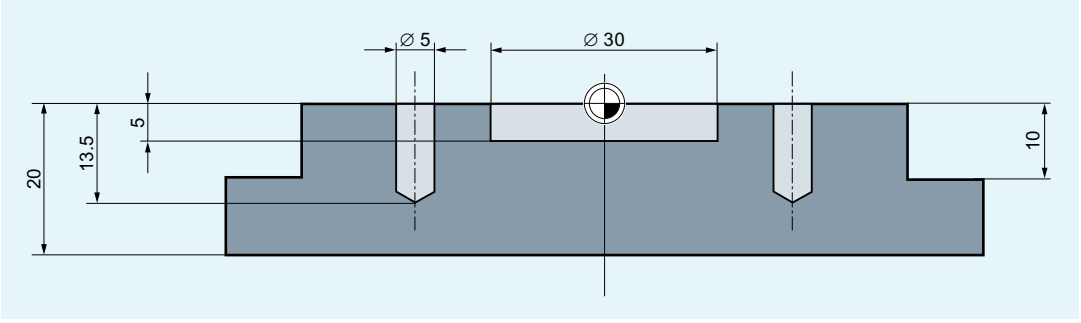
Code de programme	Commentaire
N90 X52 Z-63	
N95 G0 G40 G97 X100 Z50 M9	; Désactivation de la correction du rayon d'outil et accostage de point de changement d'outil
N100 T2 D2	; Appel de l'outil et sélection du correcteur
N105 G96 S210 M3	; Sélection de la vitesse de coupe constante
N110 G0 G42 X50 Z-60 M8	; Activation de la correction du rayon d'outil et approche de l'outil
N115 G1 Z-70 F0.12	; Tournage du diamètre 50
N120 G2 X50 Z-80 I6.245 K-5	; Tournage du rayon 8
N125 G0 G40 X100 Z50 M9	; Relève de l'outil et désactivation de la correction du rayon d'outil
N130 G0 G53 X280 Z380 D0 M5	; Accostage du point de changement d'outil
N135 M30	; Fin du programme

2.3.4.3 Exemple 3 : Programme CN de fraisage

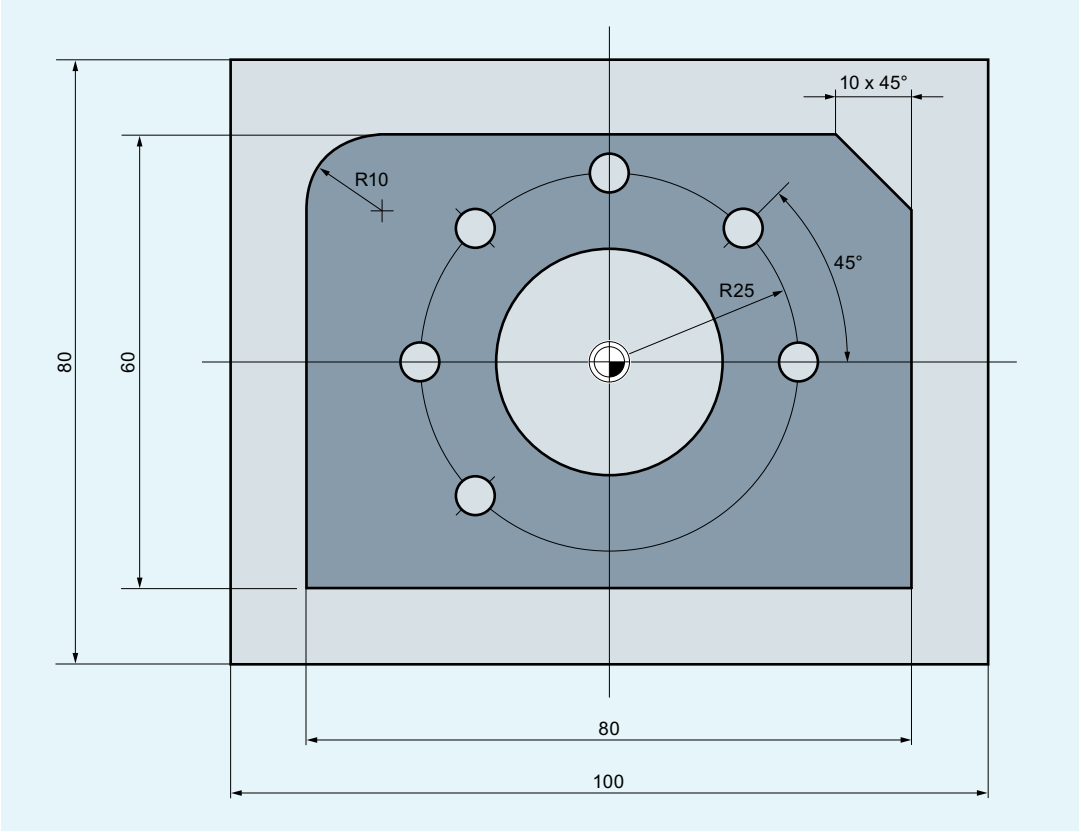
L'exemple de programme 3 est prévu pour l'usinage d'une pièce sur une fraiseuse verticale. Il contient des surfaçages, des fraisages de faces latérales, ainsi que des perçages.

Dessin coté de la pièce

Vue latérale



Vue de dessus



Programme CN

Code de programme	Commentaire
N10 T="PF60"	; Présélection de l'outil avec le nom PF60.
N20 M6	; Mise en place de l'outil dans la broche.
N30 S2000 M3 M8	; Vitesse de rotation, sens de rotation, refroidissement en marche.
N40 G90 G64 G54 G17 G0 X-72 Y-72	; Réglages de base de la géométrie et accostage du point de départ.
N50 G0 Z2	; Axe Z à la distance de sécurité.
N60 G450 CFTCP	; Comportement avec G41/G42 activé.
N70 G1 Z-10 F3000	; Fraise en profondeur de passe avec une avance = 3000 mm/min.
N80 G1 G41 X-40	; Activation de la correction du rayon de fraise.
N90 G1 X-40 Y30 RND=10 F1200	; Parcours du contour avec une avance = 1200 mm/min.
N100 G1 X40 Y30 CHR=10	
N110 G1 X40 Y-30	

Code de programme	**Commentaire**
N120 G1 X-41 Y-30	
N130 G1 G40 Y-72 F3000	; Désactivation de la correction du rayon de fraise.
N140 G0 Z200 M5 M9	; Retrait de la fraise, arrêt de la broche et du refroidissement.
N150 T="SF10"	; Présélection de l'outil avec le nom SF10.
N160 M6	; Mise en place de l'outil dans la broche.
N170 S2800 M3 M8	; Vitesse de rotation, sens de rotation, refroidissement en marche.
N180 G90 G64 G54 G17 G0 X0 Y0	; Réglages de base de la géométrie et accostage du point de départ.
N190 G0 Z2	
N200 POCKET4(2,0,1,-5,15,0,0,0,0,0,800,1300,0,21,5,,,2,0.5)	; Appel du cycle de fraisage de poche.
N210 G0 Z200 M5 M9	; Retrait de la fraise, arrêt de la broche et du refroidissement.
N220 T="ZB6"	; Appel du foret à centrer de 6 mm.
N230 M6	
N240 S5000 M3 M8	
N250 G90 G60 G54 G17 X25 Y0	; Arrêt précis G60 pour positionnement précis.
N260 G0 Z2	
N270 MCALL CYCLE82(2,0,1,-2.6,,0)	; Appel modal du cycle de perçage.
N280 POSITION:	; Marque de saut pour répétition.
N290 HOLES2(0,0,25,0,45,6)	; Modèle de positions pour réseau de trous.
N300 ENDLABEL:	; Identificateur de fin pour répétition.
N310 MCALL	; Réinitialisation de l'appel modal.
N320 G0 Z200 M5 M9	
N330 T="SPB5"	; Appel du foret hélicoïdal D 5 mm.
N340 M6	
N350 S2600 M3 M8	
N360 G90 G60 G54 G17 X25 Y0	
N370 MCALL CYCLE82(2,0,1,-13.5,,0)	; Appel modal du cycle de perçage.
N380 REPEAT POSITION	; Répétition de la description de la position du centrage.
N390 MCALL	; Réinitialisation du cycle de perçage.
N400 G0 Z200 M5 M9	
N410 M30	; Fin du programme.

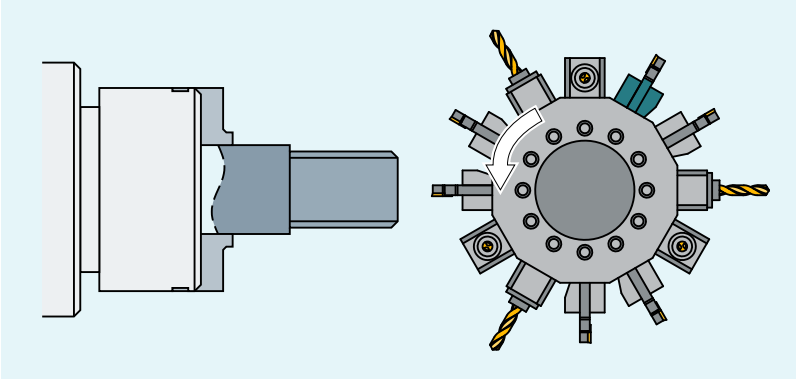
2.4 Changement d'outil

Type de changement d'outil

Remarque

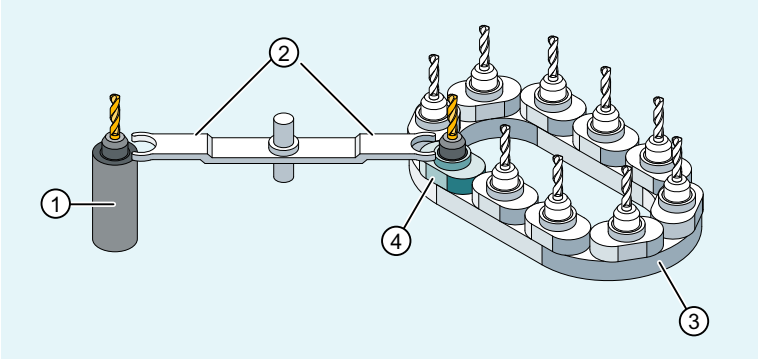
Le type de changement d'outil est défini par le constructeur de la machine lors de la mise en service.

Changement d'outil sur les tours à tourelle revolver



Pour les tourelles revolver, le changement d'outil, c'est-à-dire la recherche et le changement de l'outil, est uniquement effectué avec l'instruction T.

Changement d'outil sur les machines-outils avec magasins à chaîne, à plateau tournant ou à râtelier



① Broche
② Préhenseur
③ Magasin (ici : magasin à chaîne)
④ Point de changement pour broche

Dans le cas de magasins à chaîne, à plateau tournant et à râtelier, le changement d'outil s'effectue normalement en deux opérations :

1. L'instruction T recherche l'outil dans le magasin.
2. Puis l'instruction M charge l'outil dans la broche.

Programmation du changement d'outil lorsque la gestion des outils est active / inactive

La programmation du changement d'outil diffère selon que la gestion des outils est active ou inactive. Par conséquent, les deux variantes sont décrites :

→ Changement d'outil lorsque la gestion des outils est active (Page 61)

→ Changement d'outil lorsque la gestion des outils est inactive (Page 65)

Programmation du plan de travail

Le plan de travail (Page 33) correspondant doit être programmé avec le changement d'outil (position de base : G18). Ceci est nécessaire pour que la correction de longueur d'outil soit affectée à l'axe adéquat.

Activation de la correction d'outil

Le changement d'outil active les valeurs de correction d'outil enregistrées sous un numéro D (Page 81).

2.4.1 Changement d'outil lorsque la gestion des outils est active

Gestion des outils

La fonction "Gestion des outils" garantit à tout moment que l'outil correct est au bon endroit de la machine et que les données affectées à un outil sont à jour. De plus, elle permet un remplacement rapide des outils, et évite les rebuts par la surveillance du temps d'utilisation des outils et du temps d'arrêt de la machine en prenant en compte les outils de remplacement.

Nom d'outil

Sur une machine-outil avec une gestion d'outils active, les outils doivent avoir un nom et un numéro permettant leur identification univoque (par exemple "foret", "3").

L'appel d'outil est alors réalisable au moyen du nom d'outil, par exemple :
`T="foret"`

Remarque

Le nom d'outil ne doit pas contenir de caractères spéciaux.

2.4.1.1 Changement d'outil lorsque la gestion des outils est active avec la fonction T=emplacement

La programmation de la fonction T=emplacement conduit à un changement direct d'outil.

Application

Tours à tourelle revolver

Syntaxe

Sélectionner l'outil
```
T=<n°>
T=<nom>
T<n>=<n°>
T<n>=<nom>
```

Désélectionner l'outil
```
T0
```

Signification

`T=` :	Adresse pour la sélection de l'outil, y compris le changement d'outil et l'activation de la correction d'outil Il est possible d'affecter les valeurs suivantes :	
	`<n°>` :	Numéro de l'emplacement de magasin
	`<nom>` :	Nom de l'outil **Remarque :** Dans la programmation, la notation d'un nom d'outil (majuscules/minuscules) doit être correcte.
`<n>` :	Numéro de broche en tant qu'extension d'adresse **Remarque :** La possibilité de programmer un numéro de broche comme extension d'adresse dépend de la configuration de la machine (→ voir les indications du constructeur de machine).	
`T0` :	Désélection d'outil (emplacement de magasin inoccupé)	

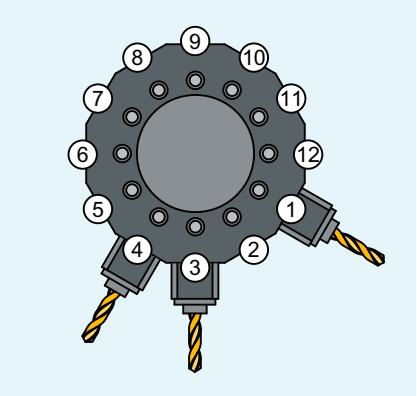
Remarque

Si l'emplacement de magasin sélectionné est inoccupé dans un magasin d'outils, l'instruction d'outil agit comme T0. La sélection de l'emplacement de magasin inoccupé peut être utilisée pour positionner l'emplacement vide.

Exemple

Une tourelle revolver comporte les emplacements 1 à 12 occupés par les outils suivants :

Emplacement	Outil	Numéro T interne	État
1	Foret, numéro de frère = 1	T15	Verrouillé
2	Inoccupé		
3	Foret, numéro de frère = 2	T10	Débloqué
4	Foret, numéro de frère = 3	T1	Activé
5 ... 12	Inoccupé		



① ... ⑫ Numéros d'emplacement de magasin

L'appel d'outil suivant est programmé dans le programme CN :

```
N10 T=1
```

L'appel est traité de la manière suivante :

1. L'emplacement de magasin 1 est pris en compte et le descripteur de l'outil est déterminé.
2. La gestion des outils reconnaît que cet outil est verrouillé et donc inutilisable.
3. Une recherche d'outil selon T="foret" est lancée conformément à la stratégie de recherche définie :
   "Rechercher l'outil actif, sinon prendre celui portant le numéro de frère immédiatement supérieur".
4. L'outil utilisable ayant été trouvé est :
   "foret" numéro de frère 3 (emplacement de magasin 4)
   La sélection d'outil est terminée et le changement d'outil est lancé.

Remarque

Avec la stratégie de recherche "Prendre le premier outil disponible du groupe", l'ordre des outils n'est pas défini. Dans ce cas, T10 **ou** T1 est sélectionné.

Avec la stratégie de recherche "Prendre le premier outil ayant l'état 'actif' dans le groupe", T1 est mis en place.

2.4.1.2 Changement d'outil lorsque la gestion des outils est active avec M6

La programmation de l'instruction T provoque la sélection de l'outil. Cependant, l'outil n'est activé que par M6 (correction d'outil incluse).

Application

Pour les machines-outils avec magasins à chaîne, à plateau tournant ou à râtelier.

Syntaxe

Sélectionner l'outil
```
T=<n°>
T=<nom>
T<n>=<n°>
T<n>=<nom>
```

Changement d'outil
```
M6
```

Désélectionner l'outil
```
T0
```

Signification

`T=` :	Adresse pour la sélection de l'outil Il est possible d'affecter les valeurs suivantes :	
	`<n°>` :	Numéro de l'emplacement de magasin
	`<nom>` :	Nom de l'outil **Remarque :** Dans la programmation, la notation d'un nom d'outil (majuscules/minuscules) doit être correcte.
`<n>` :	Numéro de broche en tant qu'extension d'adresse **Remarque :** La possibilité de programmer un numéro de broche comme extension d'adresse dépend de la configuration de la machine (→ voir les indications du constructeur de machine).	
`M6` :	Fonction M pour le changement d'outil (selon DIN 66025) M6 active l'outil sélectionné (T...) et la correction d'outil (D...).	
`T0` :	Désélection d'outil (emplacement de magasin inoccupé)	

Exemple

Code de programme	**Commentaire**
`N10 T=1 M6`	`; Mise en place de l'outil de l'emplacement de magasin 1.`
`N20 D1`	`; Sélection de la correction de longueur d'outil.`

```
Code de programme          Commentaire
N30 G1 X10 ...             ; Usinage avec l'outil T=1.
...
N70 T="foret"              ; Présélection de l'outil avec le nom "foret".
N80 ...                    ; Usinage avec l'outil T=1.
...
N100 M6                    ; Mise en place de l'outil T="foret".
N140 D1 G1 X10 ...         ; Usinage avec l'outil T="foret".
...
```

2.4.2 Changement d'outil lorsque la gestion des outils est inactive

2.4.2.1 Changement d'outil lorsque la gestion des outils est inactive avec numéro T

La programmation du numéro T conduit à un changement direct d'outil.

Application

Tours à tourelle revolver

Syntaxe

Sélectionner l'outil
`T<n°>`
`T=<n°>`
`T<n>=<n°>`

Désélectionner l'outil
`T0`
`T0=<n°>`

Signification

`T`:	Adresse pour la sélection d'outil, y compris le changement d'outil et l'activation de la correction d'outil	
`<n°>` :	Numéro de l'outil	
	Plage de valeurs :	0 ... 32000
`<n>` :	Numéro de broche en tant qu'extension d'adresse **Remarque :** La possibilité de programmer un numéro de broche comme extension d'adresse dépend de la configuration de la machine (→ voir les indications du constructeur de machine).	
`T0` :	Désélection de l'outil actif	

Exemple

Code de programme	Commentaire
N10 T1 D1	; Mise en place de l'outil T1 et activation de la correction d'outil D1.
...	
N70 T0	; Désélection de l'outil T1.
...	

2.4.2.2 Changement d'outil lorsque la gestion des outils est inactive avec M6

La programmation de l'instruction T provoque la sélection de l'outil. Cependant, l'outil n'est activé que par M6 (correction d'outil incluse).

Application

Pour les machines-outils avec magasins à chaîne, à plateau tournant ou à râtelier.

Syntaxe

Sélectionner l'outil
T<n°>
T=<n°>
T<n>=<n°>

Changement d'outil
M6

Désélectionner l'outil
T0
T0=<n°>

Signification

T:	Adresse pour la sélection de l'outil	
<n°> :	Numéro de l'outil	
	Plage de valeurs :	0 ... 32000
<n> :	Numéro de broche en tant qu'extension d'adresse **Remarque :** La possibilité de programmer un numéro de broche comme extension d'adresse dépend de la configuration de la machine (→ voir les indications du constructeur de machines).	
M6 :	Fonction M pour le changement d'outil (selon DIN 66025) M6 active l'outil sélectionné (T...) et la correction d'outil (D...).	
T0 :	Désélection de l'outil actif	

Exemple

```
Code de programme                Commentaire
N10 T1 M6                        ; Mise en place de l'outil T1.
N20 D1                           ; Sélection de la correction de longueur d'outil.
N30 G1 X10 ...                   ; Usinage avec T1.
...
N70 T5                           ; Présélection de l'outil T5.
N80 ...                          ; Usinage avec T1.
...
N100 M6                          ; Mise en place de l'outil T5.
N110 D1 G1 X10 ...               ; Usinage avec l'outil T5
...
```

2.4.3 Réaction en cas de programmation erronée de T

La réaction à une programmation erronée de T dépend de la configuration de la machine :

MD22562 TOOL_CHANGE_ERROR_MODE		
Bit	**Valeur**	**Signification**
7	0	Préréglage Lors de la programmation de T, le système contrôle immédiatement si la CN connaît le numéro T. Si ce n'est pas le cas, une alarme est émise.
	1	Le numéro T programmé est seulement vérifié après que la sélection D soit effectuée. Si le nombre T n'est pas connu de la CN, une alarme est émise lors de la sélection D. Ce comportement est alors souhaité lorsque la programmation de T doit par exemple également provoquer un positionnement sans que les paramètres d'outil correspondant ne soient requis (magasin circulaire).

2.5 Corrections d'outils

2.5.1 Contour et trajectoire d'outil programmés

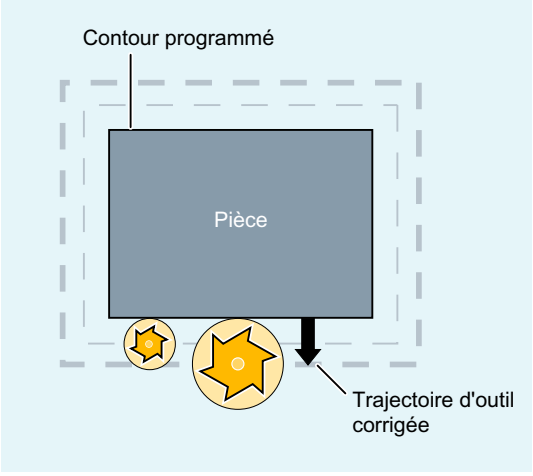
Les cotes de la pièce sont programmées directement (par exemple en fonction du dessin d'exécution). Par conséquent, les paramètres d'outil tels que le diamètre de la fraise, la position de l'arête tranchante (outil de tournage à droite/à gauche) et la longueur de l'outil ne doivent pas être pris en considération lors de la création d'un programme.

La commande corrige la trajectoire

Au cours de la fabrication d'une pièce, les déplacements d'outil doivent être commandés en fonction de la géométrie propre à l'outil, de manière à ce que le contour programmé puisse être réalisé quel que soit l'outil utilisé.

Pour que la commande puisse calculer les trajectoires d'outil, les paramètres d'outil doivent être introduits dans la mémoire de corrections d'outil de la commande. Le programme CN appelle uniquement l'outil requis (T...) et le bloc de données de correction correspondant (D...).

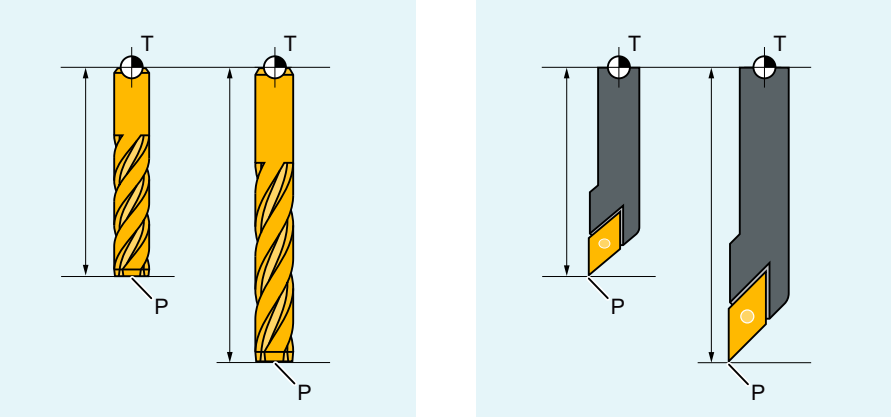
Pendant l'exécution du programme, la commande extrait les données de correction nécessaires de la mémoire de corrections d'outil et corrige la trajectoire d'outil individuellement, pour les différents outils :



2.5.2 Correction de longueur d'outil

La correction de longueur d'outil permet de compenser les différences de longueur entre les outils utilisés.

La longueur d'outil est la distance entre le point de référence de l'organe porte-outil et la pointe de l'outil :



T Point de référence de l'organe porte-outil

P Pointe de l'outil

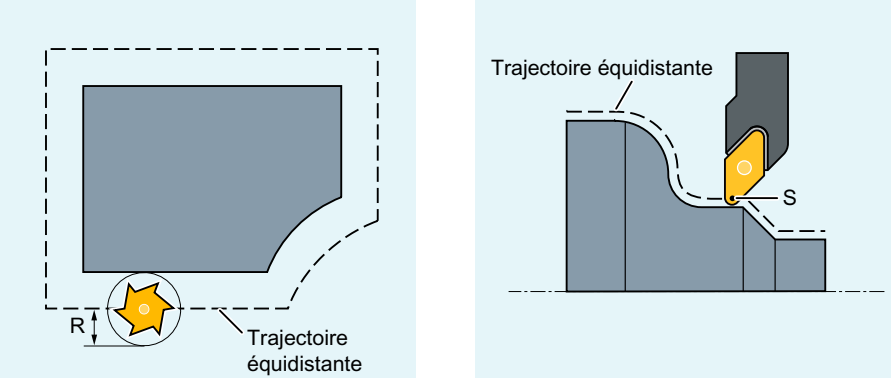
Cette longueur est mesurée et introduite dans la mémoire de corrections d'outil de la commande, avec des valeurs d'usure qui peuvent être prédéfinies. À partir de ces données, la commande calcule les déplacements dans le sens de l'axe de pénétration.

Remarque

La valeur de correction de la longueur d'outil dépend de l'orientation de l'outil dans l'espace.

2.5.3 Correction du rayon d'outil

Le contour et la trajectoire d'outil ne sont pas identiques. Le centre de la fraise ou du tranchant doit être déplacé selon le rayon d'outil en décrivant une trajectoire équidistante au contour (trajectoire du centre de l'outil). Pour cela, pendant l'exécution du programme, la commande décale la trajectoire programmée du centre de l'outil à partir du rayon d'outil de l'outil actif (mémoire de correction d'outil), de manière que le tranchant d'outil soit déplacé exactement selon le contour programmé.



R Rayon d'outil

S Centre du tranchant

La correction du rayon d'outil est décrite en détail au chapitre "Corrections de rayon d'outil (Page 256)".

Voir aussi

Correction d'outil 2D ½ (CUT2D, CUT2DD, CUT2DF, CUT2DFD) (Page 291)

2.5.4 Mémoire de corrections d'outil

Pour chaque tranchant d'outil, la mémoire de corrections d'outil de la commande doit contenir les données suivantes :

- Type d'outil
- Position du tranchant
- Dimensions géométriques de l'outil (longueur, rayon)

Ces données sont introduites en tant que paramètres d'outil (25 au maximum). Les paramètres requis pour un outil dépendent du type d'outil. Les paramètres qui ne sont pas nécessaires doivent être mis à "zéro" (valeur par défaut du système).

Remarque

Dès qu'une valeur est introduite dans la mémoire de corrections, elle est prise en compte à chaque appel d'outil.

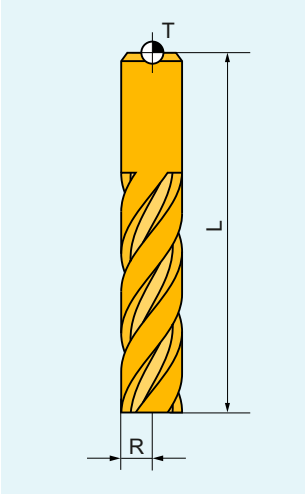
Type d'outil

Le type d'outil (foret, fraise ou outil de tournage) détermine quelles sont les données géométriques nécessaires et comment elles doivent être prises en compte.

Position du tranchant

La position du tranchant décrit la position de la pointe d'outil par rapport au centre du tranchant. La position et le rayon du tranchant sont nécessaires au calcul de la correction de rayon des outils de tournage (type d'outil 5xx) (Page 77).

Dimensions géométriques de l'outil (longueur, rayon)



T	Point de référence de l'organe porte-outil
R	Rayon d'outil
L	Longueur d'outil

Les dimensions géométriques de l'outil sont formées de plusieurs composantes (géométrie, usure). La commande calcule à partir de ces composantes une dimension résultante (par exemple longueur totale 1, rayon total). Chaque cote totale prend effet lors de l'activation de la mémoire de corrections.

La façon dont ces valeurs sont exploitées dans les axes dépend du type d'outil et du plan actuel (G17/G18/G19).

2.5.5 Types d'outil

2.5.5.1 Numéro de type d'outil et groupes d'outils

Un numéro de 3 chiffres univoque est affecté à chaque type d'outil. L'affectation de l'outil à l'une des technologies suivantes ou à l'un des groupes d'outils suivants s'effectue via le premier chiffre (position des centaines) :

Type d'outil	Groupe d'outils
1xy	Fraises (Page 72)
2xy	Forets (Page 74)
3xy	Réservé
4xy	Outils de rectification (Page 75)
5xy	Outils de tournage (Page 77)
6xy	Réservé
7xy	Outils spéciaux (Page 79)

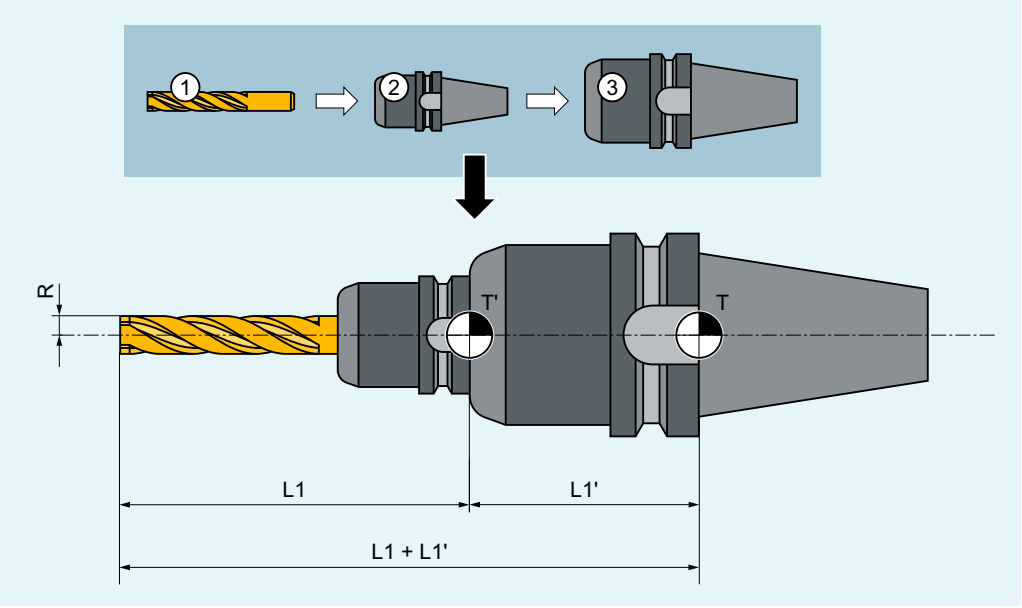
2.5.5.2 Fraises

Le groupe "Fraises" comprend les types d'outil suivants :

100	Fraise selon CLDATA (Cutter Location Data)
110	Fraise à bout rond
111	Fraise cylindrique à matrices
120	Fraise deux tailles à queue sans arrondi
121	Fraise deux tailles à queue avec arrondi
130	Fraise pour tête à renvoi d'angle sans arrondi
131	Fraise pour tête à renvoi d'angle avec arrondi
140	Fraise tourteau
145	Fraise à fileter
150	Fraise trois tailles
151	Scie
155	Fraise conique tronquée sans arrondi
156	Fraise conique tronquée avec arrondi
157	Fraise conique à matrices
160	Foret fraise à fileter

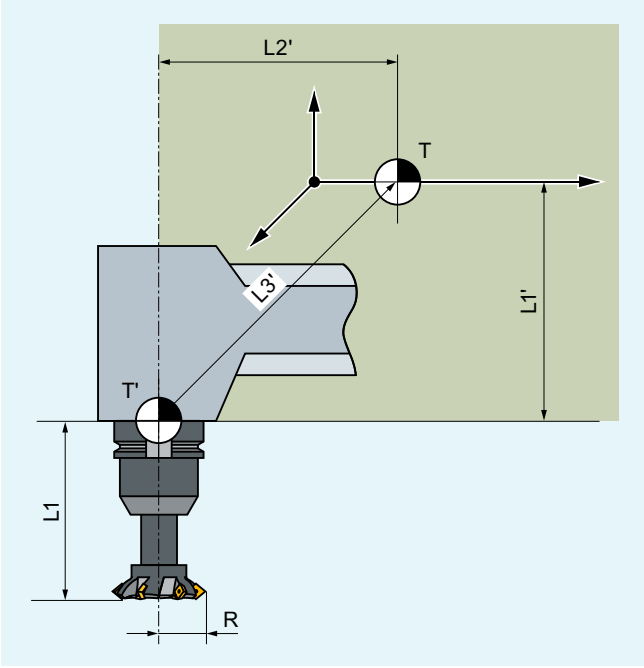
Paramètres d'outil

Les figures suivantes donnent un aperçu des paramètres d'outil enregistrés dans la mémoire de corrections pour les fraises :



①	Outil
②	Porte-outil
③	Adaptateur d'outil
T	Point de référence de l'adaptateur (lorsqu'un outil est inséré = point de référence du porte-outil)
T'	Point de référence du porte-outil
L1	Géométrie - Longueur 1
L1'	Cote d'adaptation - Longueur 1
L1 + L1'	Longueur totale L1
R	Rayon

Paramètres d'outil	Signification
$TC_DP1	Type d'outil 1xy
$TC_DP3	Géométrie - Longueur 1
$TC_DP6	Géométrie - Rayon
$TC_DP21	Cote d'adaptation - Longueur 1
• Valeurs d'usure en fonction des nécessités. • Les autres valeurs doivent être mises à 0.	



T	Point de référence de l'organe porte-outil
T'	Point de référence du porte-outil
L1	Géométrie - Longueur 1
R	Rayon d'outil
L1'	Cote de base - Longueur 1
L2'	Cote de base - Longueur 2
L3'	Cote de base - Longueur 3

Paramètres d'outil	Signification
$TC_DP1	Type d'outil
$TC_DP3	Géométrie - Longueur 1
$TC_DP6	Géométrie - Rayon
$TC_DP21	Cote de base - Longueur 1
$TC_DP22	Cote de base - Longueur 2
$TC_DP23	Cote de base - Longueur 3
• Valeurs d'usure en fonction des nécessités. • Les autres valeurs doivent être mises à 0.	

2.5.5.3 Forets

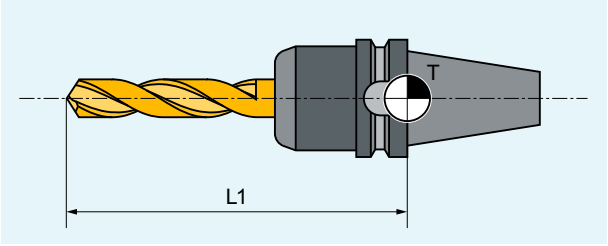
Le groupe "Forets" comprend les types d'outil suivants :

N°	Type d'outil
200	Foret hélicoïdal
205	Foret
210	Barre d'alésage

N°	Type d'outil
220	Foret à centrer
230	Foret conique à lamer
231	Foret cylindrique à lamer
240	Taraud à pas gros
241	Taraud à pas fin
242	Taraud à pas fin Withworth
250	Alésoir

Paramètres d'outil

La figure suivante donne un aperçu des paramètres d'outil enregistrés dans la mémoire de corrections pour les forets :



T Point de référence de l'organe porte-outil

L1 Longueur 1

Paramètres d'outil	Signification
$TC_DP1	Type d'outil
$TC_DP3	Géométrie - Longueur 1
• Valeurs d'usure en fonction des nécessités. • Les autres valeurs doivent être mises à 0.	

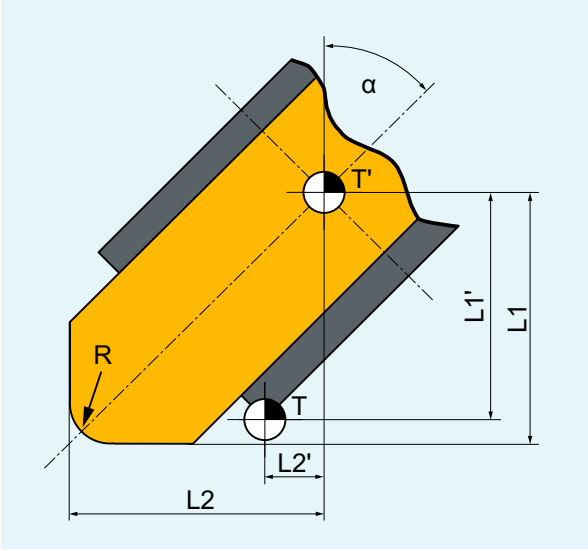
2.5.5.4 Outils de rectification

Le groupe "Outils de rectification" comprend les types d'outils suivants :

400	Meule tangentielle
401	Meule tangentielle avec surveillance
402	Meule tangentielle sans surveillance sans cote de base (gestion d'outils)
403	Meule tangentielle avec surveillance sans cote de base pour vitesse périphérique de meule VPM
410	Meule frontale
411	Plateau (gestion d'outils) avec surveillance
412	Plateau (gestion d'outils) sans surveillance
413	Plateau avec surveillance sans cote de base pour vitesse périphérique de meule VPM
490	Dresseur

Paramètres d'outil

La figure suivante donne un aperçu des paramètres d'outil enregistrés dans la mémoire de corrections pour les outils de rectification :



T	Point de référence de l'organe porte-outil
T'	Point de référence du porte-outil
L1	Géométrie - Longueur 1
L1'	Cote de base - Longueur 1
L2	Géométrie - Longueur 2
L2'	Cote de base - Longueur 2
R	Rayon
α	Angle de la meule inclinée

Paramètres spécifiques au tranchant	Signification
$TC_DP1	Type d'outil 4xy
$TC_DP2	Position du tranchant
$TC_DP3	Géométrie Longueur 1
$TC_DP4	Géométrie Longueur 2
$TC_DP6	Rayon
$TC_DP21	Cote de base Longueur 1
$TC_DP22	Cote de base Longueur 2
• Valeurs d'usure en fonction des nécessités. • Les autres valeurs doivent être mises à 0.	

Paramètres spécifiques à l'outil	Signification
$TC_TPG1	Numéro de broche
$TC_TPG2	Règle de concaténation [1)]
$TC_TPG3	Rayon minimal de la meule

Paramètres spécifiques à l'outil	Signification
$TC_TPG4	Largeur minimale de la meule
$TC_TPG5	Largeur courante de la meule
$TC_TPG6	Vitesse maximale
$TC_TPG7	Vitesse circonférentielle maxi.
$TC_TPG8	Angle de la meule inclinée
$TC_TPG9	N° de paramètre pour calcul du rayon
$TC_TPG_DRSPATH	Chemin de répertoire du programme de dressage
$TC_TPG_DRSPROG	Nom du programme de dressage

[1] Les corrections de longueur de la géométrie, de l'usure et de la cote de base peuvent être associées dans la correction côté droit et côté gauche de la meule. Autrement dit, quand les corrections de longueur sont modifiées pour le tranchant gauche, les valeurs sont automatiquement introduites pour le tranchant droit et inversement.

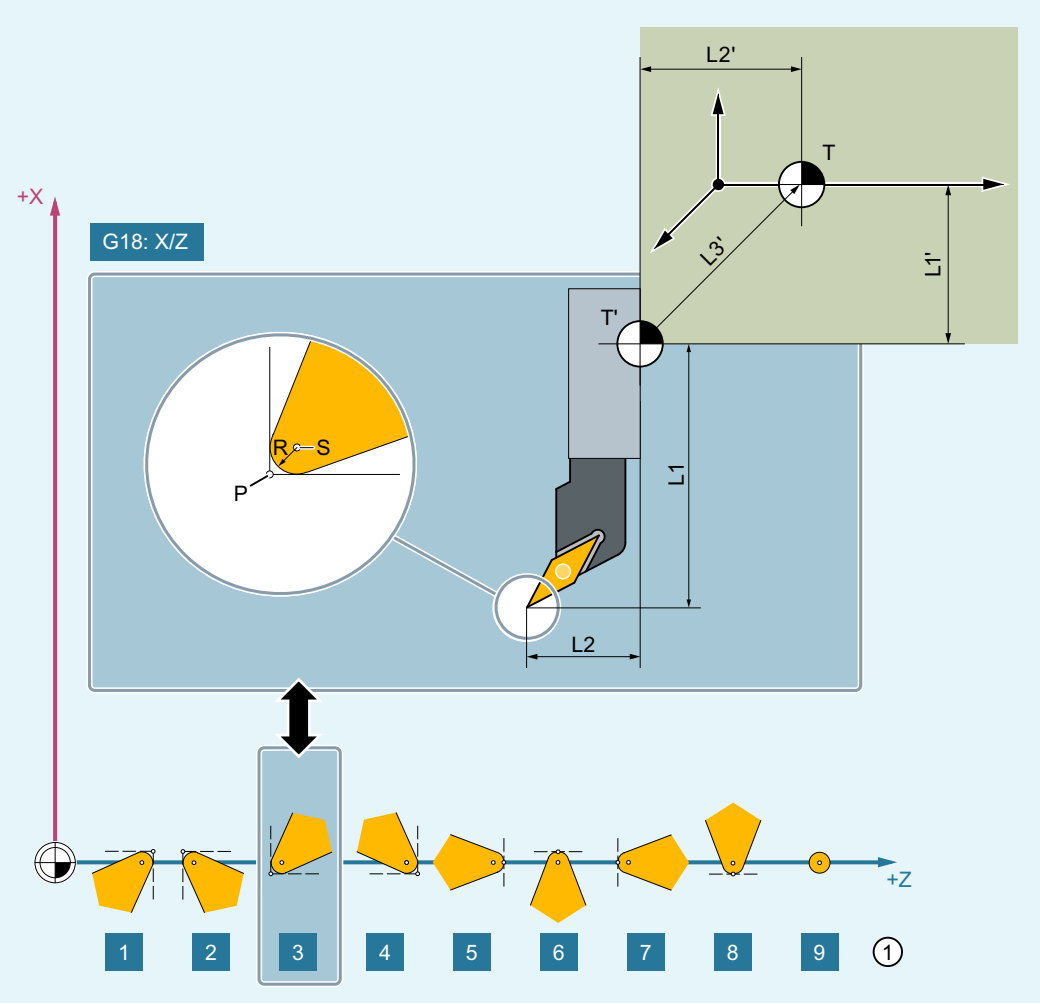
2.5.5.5 Outils de tournage

Le groupe "Outils de tournage" comprend les types d'outils suivants :

500	Outil d'ébauche
510	Outil de finition
520	Outil à plonger
530	Outil à tronçonner
540	Outil à fileter
550	Outil à plaquette ronde / outil de forme (gestion d'outils)
560	Foret à plaquette (ECOCUT)
580	Palpeur avec paramètre Position du tranchant

Paramètres d'outil

La figure suivante donne un aperçu des paramètres d'outil enregistrés dans la mémoire de corrections pour les outils de tournage :



①	Position du tranchant (1 - 9) pour l'usinage derrière le centre de rotation
P	Pointe de l'outil
S	Centre du tranchant
R	Rayon du tranchant
T	Point de référence de l'organe porte-outil
T'	Point de référence du porte-outil
L1	Géométrie - Longueur 1
L2	Géométrie - Longueur 2
L1'	Cote de base - Longueur 1
L2'	Cote de base - Longueur 2
L3'	Cote de base - Longueur 3

Paramètres d'outil	Signification
$TC_DP1	Type d'outil
$TC_DP2	Position du tranchant
$TC_DP3	Géométrie - Longueur 1
$TC_DP4	Géométrie - Longueur 2
$TC_DP6	Géométrie - Rayon (rayon du tranchant)
$TC_DP21	Cote de base - Longueur 1
$TC_DP22	Cote de base - Longueur 2
$TC_DP23	Cote de base - Longueur 3
• Valeurs d'usure en fonction des nécessités. • Les autres valeurs doivent être mises à 0.	

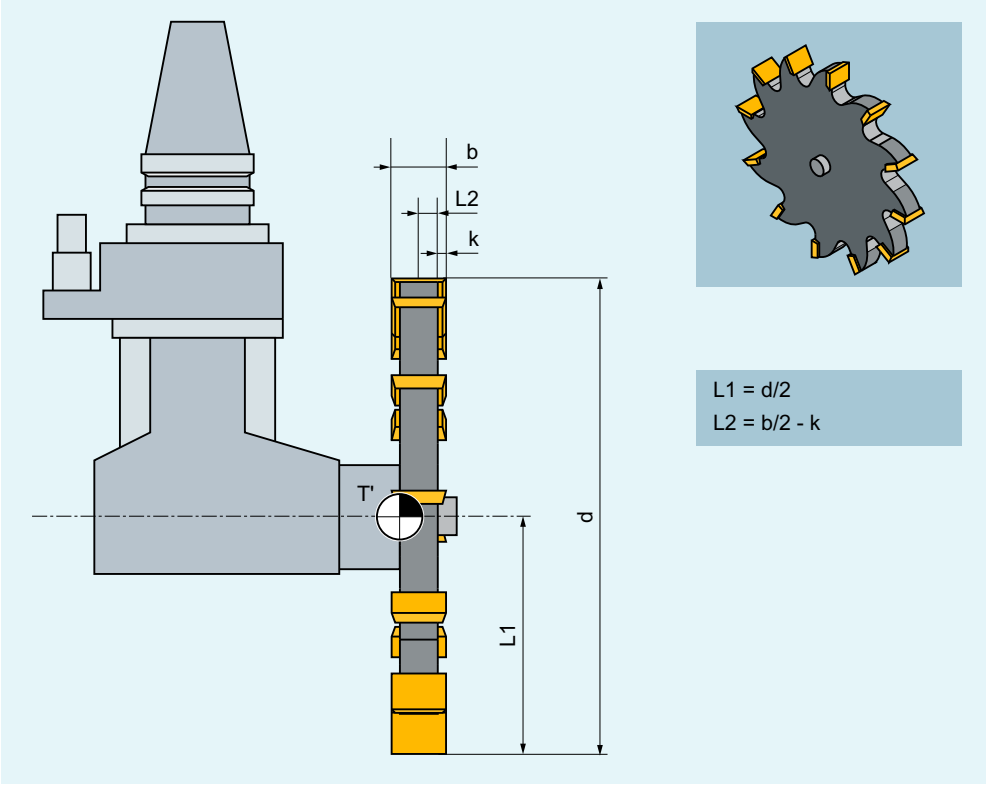
2.5.5.6 Outils spéciaux

Le groupe "Outils spéciaux" comprend les types d'outil suivants :

700	scie à rainurer
710	Palpeur 3D
711	Palpeur à arête
712	Palpeur unidirectionnel
713	Palpeur L
714	Palpeur étoilé
725	Outil étalon
730	Butée
731	Fourreau de broche
732	Lunette

Paramètres d'outil

La figure suivante donne un aperçu des paramètres d'outil enregistrés dans la mémoire de corrections pour le type d'outil "Scie à rainurer" :



T'	Point de référence du porte-outil
L1	Géométrie - Longueur 1
L2	Géométrie - Longueur 2
d	Diamètre
b	Largeur de rainure
k	Dépassement

Paramètres d'outil	Signification
$TC_DP1	Type d'outil
$TC_DP3	Géométrie - Longueur 1
$TC_DP4	Géométrie - Longueur 2
$TC_DP6	Diamètre
$TC_DP7	Largeur de rainure
$TC_DP8	Dépassement
$TC_DP21	Cote de base Longueur 1
$TC_DP22	Cote de base Longueur 2
$TC_DP23	Cote de base Longueur 3
• Valeurs d'usure en fonction des nécessités. • Les autres valeurs doivent être mises à 0.	

2.5.6 Activation/désactivation des corrections d'outil (D, D0)

Différents blocs de données de correction d'outil (par exemple valeurs de correction différentes pour les tranchants gauche et droit d'un outil pelle) peuvent être affectés aux tranchants 1 à 8 (avec gestion d'outils 12 actif) d'un outil.

Dans le cas d'un tranchant spécial, les données de correction (entre autres les données de correction de longueur d'outil) sont activées par appel du numéro D. Avec la programmation de D0, les corrections sont inopérantes pour l'outil en question.

Une correction de rayon d'outil doit être activée en outre par le biais de G41/G42.

Remarque

Les corrections de longueur d'outil sont opérantes
quand le numéro D est programmé. Si aucun numéro D n'est programmé et qu'il y a un changement d'outil, c'est le réglage par défaut défini dans les paramètres machine qui devient actif (→ voir les indications du constructeur de machines).

Syntaxe

```
D<N°>
X... Y... Z...
G41/G42 X... Y... Z...
G40
D0
```

Signification

D :	Adresse d'activation d'un bloc de données de correction pour l'outil actif La correction de longueur d'outil est effectuée lors de l'exécution du premier déplacement programmé de l'axe de correction de longueur associé. **Important :** une correction de longueur d'outil agit également sans programmation de D si l'activation automatique d'un tranchant d'outil a été configurée pour le changement d'outil (→ voir les indications du constructeur de la machine).	
<N°> :	Le numéro D indique le bloc de données de correction d'outil à activer. Le type de programmation de D dépend de la configuration de la machine (→ voir les indications du constructeur de la machine). Les options disponibles sont les suivantes : • Numéro D = numéro de tranchant Pour chaque outil T <N°> (sans gestion d'outils) ou T="nom" (avec gestion d'outils), il existe les numéros D de 1 à 12. Ces numéros D sont affectés directement aux tranchants des outils. À chaque numéro D (= numéro de tranchant) correspond un bloc de données de correction ($TC_DPx[<t>,<d>]). • Liberté de choix des numéros D Les numéros D peuvent être affectés librement aux numéros des tranchants d'un outil. La limite supérieure des numéros D utilisables est définie par un paramètre machine.	
	Plage de valeurs :	0 - 32000
D0 :	Désactivation du bloc de données de correction pour l'outil actif	
G41 :	Instruction d'activation de la correction du rayon d'outil avec sens d'usinage **à gauche** du contour	
G42 :	Instruction d'activation de la correction du rayon d'outil avec sens d'usinage **à droite** du contour	
G40 :	Instruction de désactivation de la correction du rayon d'outil	

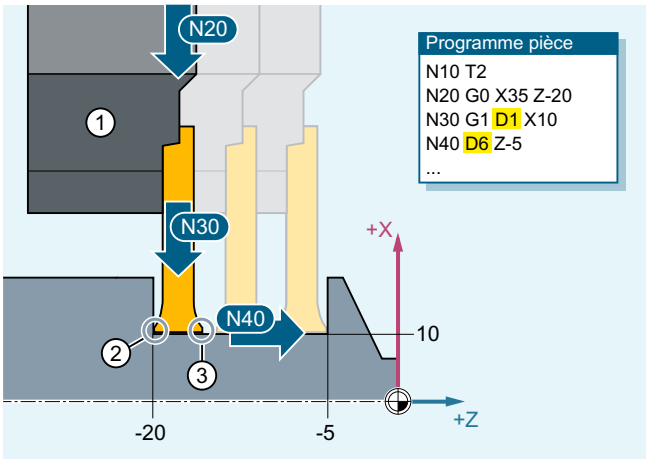
Des informations complémentaires sur G40/G41/G42 figurent au chapitre "Corrections de rayon d'outil (Page 256)".

Exemples

Exemple 1 : Changement d'outil avec instruction T (tournage)

Code de programme	Commentaire
N10 T1 D1	; Mise en place de l'outil T1 et activation du bloc de données de correction D1 de l'outil T1
N11 G0 X... Z...	; Exécution des déplacements correspondant aux corrections de longueur
N50 T4 D2	; Mise en place de l'outil T4 et activation du bloc de données de correction D2 de l'outil T4
...	
N70 G0 Z... D1	; Activation de l'autre tranchant D1 de l'outil T4

Exemple 2 : Valeurs de correction différentes pour les tranchants gauche et droit d'un outil pelle



① Outil de tournage en plongée (T2)
② Tranchant D1
③ Tranchant D6

Voir aussi

Correction du rayon d'outil (Page 69)

Informations complémentaires

Modification des données de correction d'outil

Dans le réglage standard, les modifications des données de correction d'outil ne prennent effet qu'après la reprogrammation de T ou de D.

Le paramètre machine suivant permet de définir que les données de correction d'outil saisies deviennent immédiatement effectives :

MD9440 $MM_ACTIVATE_SEL_USER

⚠ ATTENTION

Risque de collision

Lorsque MD9440 est activé, les corrections d'outil qui résultent de modifications de données de correction d'outil effectuées pendant l'arrêt du programme pièce sont exécutées au moment de la reprise du programme pièce.

2.5.7 Inhibition des corrections d'outil (SUPD)

La commande SUPD permet d'inhiber les corrections d'outil par bloc.

L'avantage par rapport à la désélection des corrections d'outil avec D0 est qu'avec SUPD, les informations de correction d'outil sont conservées, ce qui évite la réactivation des données de correction d'outil via la programmation du numéro D.

SUPD sert exclusivement à inhiber la correction de longueur d'outil, par exemple dans les blocs du programme pièce dans lequel le point d'origine doit être pris en compte, mais pas la longueur d'outil. Une utilisation avec G41/G42 actif pour une inhibition supplémentaire des corrections du rayon d'outil n'est pas recommandée.

Syntaxe

```
D<N°>
X... Y... Z...
X... Y... Z... SUPD
...
D0
```

Signification

SUPD:	Instruction G de désactivation du bloc de données de correction pour l'outil actif dans le bloc actif	
	Prise d'effet :	par bloc

Autres conditions à prendre en compte

- SUPD ne doit être utilisée que dans des blocs linéaires.
- SUPD ne peut pas être utilisée dans des actions synchrones.
- SUPD ne peut pas être utilisée avec les fonctions suivantes :
  - Correction du rayon d'outil 3D pour le fraisage en bout 3D (CUT3DFxx)
  - Tables de courbes (CTAB)
- L'utilisation de SUPD pour la correction de rayon d'outil active (G41/G42) est possible, mais pas recommandée.
  Si cette fonction est néanmoins utilisée, la donnée de réglage suivante doit être mise à 0 :
  SD42480 $SC_STOP_CUTCOM_STOPRE = 0
  Cela évite une interruption de programme en cas d'activation de G41/G42.

Exemple

Dans le sous-programme SUB_SUP, la longueur d'outil doit être inhibée lors du processus.

Programme pièce

Code de programme	Commentaire
...	
N300 $P_UIFR[1]=CTRANS(X,1000,Y,400,Z,-120)	
N310 T="BALL_D3"	

Code de programme	Commentaire
N320 M6	
N330 TRAFOOF	
N340 G54 G0 Z49 D1	
N350 G0 X1100 Y500 C0 A0	
N360 **SUB_SUP**	; appel du sous-programme
N370 G0 X1200	
N380 M30	

Sous-programme avec D0

Code de programme	Commentaire
N10 PROC **SUB_SUP**	
N20 DEF INT NUMBER	
N30 NUMBER=$P_TOOL	
N40 G0 Z49 **D0**	; désactivation des corrections d'outil
N50 D=NUMBER	; réactivation des corrections d'outil
N60 RET	

Sous-programme avec SUPD

Code de programme	Commentaire
N10 PROC **SUB_SUP**	
N40 G0 Z49 **SUPD**	; inhibition des corrections d'outil dans le bloc actif
N60 RET	

2.5.8 Offset de correction d'outil programmable (TOFFL, TOFF, TOFFR, TOFFLR)

Avec les adresses TOFFx, l'utilisateur a la possibilité de modifier la longueur d'outil effective et le rayon d'outil effectif dans le programme CN, sans modifier les données de correction d'outil enregistrées dans la mémoire de corrections.

À la fin du programme, les offsets programmés sont à nouveau supprimés.

Syntaxe

Offset de longueur d'outil

```
TOFFL=<valeur>
TOFFL[1]=<valeur> TOFFL[2]=<valeur> TOFFL[3]=<valeur>
TOFF[<GeoAx>]=<valeur>
```

La longueur d'outil peut être modifiée simultanément dans les trois composantes. Il est cependant interdit d'utiliser simultanément dans un même bloc des instructions du groupe

TOFFL/TOFFL[1..3] d'un côté, et du groupe TOFF[<GeoAx>] de l'autre. De même, l'écriture simultanée de TOFFL et de TOFFL[1] est interdite dans un même bloc.

Si les trois composantes de longueur d'outil ne sont pas toutes programmées dans un bloc, les composantes non programmées restent inchangées. Il est ainsi possible de structurer les corrections par blocs pour plusieurs composantes. Cependant, ceci est seulement valable tant que les composantes d'outil sont modifiées soit avec TOFFL soit avec TOFF. Un changement de la variante de programmation de TOFFL à TOFF ou inversement efface d'abord tous les offsets ayant éventuellement été programmés précédemment pour les longueurs d'outil (voir l'exemple 3).

Offset de rayon d'outil

```
TOFFR=<valeur>
```

Offset de rayon d'outil et offset de longueur d'outil simultanés

```
TOFFLR=<valeur>
```

Signification

<table>
<tr><td>TOFFL :</td><td colspan="2">Correction de la longueur d'outil effective
TOFFL peut être programmé avec ou sans indice :</td></tr>
<tr><td></td><td>TOFFL=...</td><td>La valeur offset programmée agit dans le même sens que la composante de longueur d'outil L1 enregistrée dans la mémoire de corrections.
Les instructions TOFFL et TOFFL[1] ont le même effet.</td></tr>
<tr><td></td><td>TOFFL[1]=...
TOFFL[2]=...
TOFFL[3]=...</td><td>La valeur offset programmée agit dans le même sens que la composante de longueur d'outil L1, L2 ou L3 enregistrée dans la mémoire de corrections.</td></tr>
<tr><td></td><td colspan="2">Remarque :
La façon dont les valeurs de correction de longueur d'outil sont exploitées dans les axes dépend du type d'outil et du plan de travail actuel (G17/G18/G19).</td></tr>
<tr><td>TOFF :</td><td colspan="2">Correction de la longueur d'outil dans la composante qui est parallèle à l'axe géométrique indiqué
TOFF agit dans le sens de la composante de longueur d'outil qui, lorsqu'il n'y a pas eu de rotation de l'outil (organe porte-outil orientable ou transformation d'orientation), est parallèle à l'axe géométrique indiqué dans l'indice.
Remarque :
Un frame n'influence pas l'affectation des valeurs programmées aux composantes de longueur d'outil. Autrement dit, l'affectation des composantes de longueur d'outil aux axes géométriques ne repose pas sur le système de coordonnées pièce (SCP), mais sur le système de coordonnées de l'outil en position de base.</td></tr>
<tr><td><GeoAx> :</td><td colspan="2">Descripteur de l'axe géométrique</td></tr>
<tr><td>TOFFR :</td><td colspan="2">Correction du rayon d'outil effectif
TOFFR modifie le rayon d'outil effectif lorsque la correction du rayon d'outil est activée, en fonction de la valeur offset programmée.</td></tr>
</table>

TOFFLR :	Correction de la longueur d'outil effective dans la composante L1 **et** correction du rayon d'outil effectif **Remarque :** Pour les outils avec arrondi (types 111, 121, 131 et 156), TOFFLR corrige également le rayon d'angle.	
<Valeur> :	Valeur de l'offset	
	Type :	REAL

Exemples

Exemple 1 : offset de longueur d'outil positif

L'outil actif est un foret d'une longueur L1 = 100 mm.

Le plan actif est G17. Le foret est donc orienté dans la direction Z.

La longueur effective du foret doit être prolongée de 1 mm. Les variantes suivantes sont disponibles pour la programmation de cet offset de la longueur d'outil :

- `TOFFL=1`
- `TOFFL[1]=1`
- `TOFF[Z]=1`

Exemple 2 : offset de longueur d'outil négatif

L'outil actif est un foret d'une longueur L1 = 100 mm.

Le plan actif est G18. Le foret est donc orienté dans la direction Y.

La longueur effective du foret doit être raccourcie de 1 mm. Les variantes suivantes sont disponibles pour la programmation de cet offset de la longueur d'outil :

- `TOFFL=-1`
- `TOFFL[1]=-1`
- `TOFF[Y]=1`

Exemple 3 : changement de la variante de programmation de TOFFL à TOFF

L'outil actif est une fraise. Le plan actif est G17.

```
Code de programme                    Commentaire
N10 TOFFL[1]=3 TOFFL[3]=5            ; Offsets actifs : L1=3, L2=0, L3=5
N20 TOFFL[2]=4                       ; Offsets actifs : L1=3, L2=4, L3=5
N30 TOFF[Z]=1.3                      ; Offsets actifs : L1=0, L2=0, L3=1.3
```

Exemple 4 : affectation des valeurs offsets après un changement de plan

```
Code de programme          Commentaire
N10 $TC_DP1[1,1]=120
N20 $TC_DP3[1,1]=100       ; Longueur d'outil L1=100 mm
N30 T1 D1 G17
N40 TOFF[Z]=1.0            ; Offset dans la direction Z (correspond à L1 avec G17).
```

Code de programme	Commentaire
N50 G0 X0 Y0 Z0	; Position d'axe machine X0 Y0 Z101
N60 G18 G0 X0 Y0 Z0	; Position d'axe machine X0 Y100 Z1
N70 G17	
N80 **TOFFL=1.0**	; Offset dans la direction L1 (correspond à Z avec G17).
N90 G0 X0 Y0 Z0	; Position d'axe machine X0 Y0 Z101.
N100 G18 G0 X0 Y0 Z0	; Position d'axe machine X0 Y101 Z0.

Dans cet exemple, l'offset de 1 mm dans l'axe Z est conservé dans le bloc N60 lors du passage à G18, la longueur d'outil effective dans l'axe Y étant la longueur d'outil inchangée de 100 mm.

Dans le bloc N100, l'offset agit cependant dans l'axe Y lors du passage à G18, car il a été affecté à la longueur d'outil L1 lors de la programmation et cette composante de longueur agit dans l'axe Y lorsque le plan est G18.

Exemple 5 : Offset de rayon d'outil et offset de longueur d'outil simultanés

a, fraise à queue **sans** arrondi (type d'outil 120) :

Code de programme	Commentaire
...	
TOFFLR=0.1	; Offsets actifs :
	; Offset de longueur d'outil (L1) = 5
	; Offset de rayon d'outil = 5
...	

b, fraise à queue **avec** arrondi (type d'outil 121) :

Code de programme	Commentaire
...	
TOFFLR=0.1	; Offsets actifs :
	; Offset de longueur d'outil (L1) = 0.1
	; Offset de rayon d'outil = 0.1
	; Offset de rayon d'angle = 0.1
...	

Informations complémentaires

Offsets de longueur d'outil

Selon le type de programmation, les offsets programmés pour les longueurs d'outil sont affectés soit aux composantes de longueur d'outil L1, L2 et L3 (TOFFL) enregistrées dans la mémoire de corrections, soit aux axes géométriques (TOFF). Les offsets programmés sont traités en conséquence en cas de changement de plan (G17/G18/G19 ↔ G17/G18/G19) :

- Si les valeurs offsets sont affectées aux composantes de longueur d'outil, les sens dans lesquels les offsets programmés agissent sont changés en conséquence.
- Si les valeurs offsets sont affectées aux axes géométriques, l'affectation relative aux axes de coordonnées n'est pas influencée par un changement de plan.

Les données de réglage suivantes sont exploitées pour l'affectation des valeurs offsets programmées aux composantes de longueur d'outil :

SD42940 $SC_TOOL_LENGTH_CONST (changement des composantes de longueur d'outil en cas de changement de plan)

SD42950 $SC_TOOL_LENGTH_TYPE (affectation de la correction de longueur d'outil indépendamment du type d'outil)

Si ces données de réglage ont des valeurs valides différentes de 0, celles-ci ont la priorité sur le contenu du groupe G 6 (sélection du plan G17/G18/G19) ou sur le type d'outil ($TC_DP1[<n° T>, <n° D>]) figurant dans les paramètres d'outil. Autrement dit, ces données de réglage influencent l'exploitation des offsets de la même manière que les composantes de longueur d'outil L1 à L3.

Offset de rayon d'outil

L'adresse TOFFR a presque le même effet que l'adresse OFFN (Page 256). Il y a une différence uniquement lorsque la transformation de courbe enveloppe (TRACYL) et la correction de flanc de rainure sont activées. Dans ce cas, OFFN agit sur le rayon d'outil avec un signe négatif, et TOFFR avec un signe positif.

OFFN et TOFFR peuvent être actives simultanément. Elles ont alors généralement un effet additif (sauf en cas de correction de flanc de rainure).

Changement d'outil

Toutes les valeurs offsets sont conservées en cas de changement d'outil (changement de tranchant). Elles s'appliquent alors au nouvel outil (nouveau tranchant).

Variables système pour la lecture des valeurs offsets courantes

Les offsets actifs courants peuvent être lus au moyen des variables système suivantes :

Variable système	Signification
$P_TOFFL [<n>] avec 0 ≤ n ≤ 3	Lecture de la valeur offset courante de TOFFL (avec n = 0) ou de TOFFL[1...3] (avec n = 1, 2, 3) dans le contexte de prétraitement.
$P_TOFF [<GeoAx>]	Lecture de la valeur offset courante de TOFF[<GeoAx>] dans le contexte de prétraitement.
$P_TOFFR	Lecture de la valeur offset courante de TOFFR dans le contexte de prétraitement.
$P_TOFFCR	Lecture de la valeur offset courante du rayon d'angle dans le contexte de prétraitement.
$AC_TOFFL[<n>] avec 0 ≤ n ≤ 3	Lecture de la valeur offset courante de TOFFL (avec n = 0) ou de TOFFL[1...3] (avec n = 1, 2, 3) dans le contexte d'exécution (actions synchrones)
$AC_TOFF[<GeoAx>]	Lecture de la valeur offset courante de TOFF[<GeoAx>] dans le contexte d'exécution (actions synchrones).
$AC_TOFFR	Lecture de la valeur offset courante de TOFFR dans le contexte d'exécution (actions synchrones).
$AC_TOFFCR	Lecture de la valeur offset courante du rayon d'angle dans le contexte d'exécution (actions synchrones).

Remarque

Les variables système $AC_TOFFL, $AC_TOFF, AC_TOFFR et AC_TOFFCR déclenchent un arrêt automatique du prétraitement des blocs en cas de lecture dans le contexte de prétraitement (programme CN).

Applications

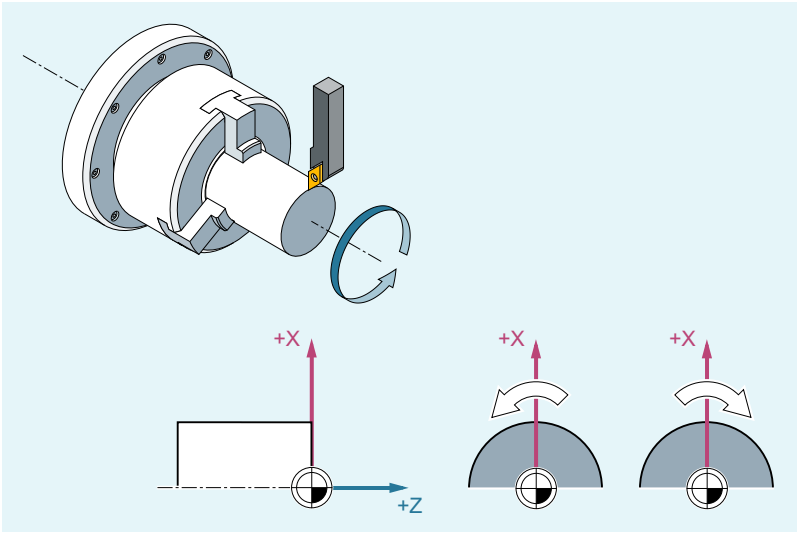
La fonction "Offset de correction d'outil programmable" est particulièrement intéressante pour les fraises à bout rond et les fraises avec un rayon d'arrondi, leur calcul étant généralement effectué par rapport au centre de la sphère et non pas par rapport à la pointe de la sphère dans le système FAO. Lors de la mesure de l'outil, on mesure cependant généralement la pointe de l'outil, qui est enregistrée dans la mémoire de corrections en tant que longueur d'outil.

Pour la correction du rayon d'outil 3D lorsqu'une fraise à bout rond est utilisée, il est utile de corriger simultanément la longueur et le rayon d'outil de la même valeur. Dans ce cas, l'utilisateur dispose de l'adresse TOFFLR.

2.6 Mouvement de broche

2.6.1 Vitesse de rotation de broche (S), sens de rotation (M3, M4, M5)

Les indications vitesse et sens de rotation de broche permettent d'appliquer un mouvement de rotation à la broche et fournissent la condition nécessaire à l'usinage par enlèvement des copeaux.



Outre la broche principale, il peut exister des broches supplémentaires (p. ex. la contre-broche ou un outil motorisé dans le cas des tours). En règle générale, la broche principale est déclarée comme broche maître par un paramètre machine. Cette affectation peut être modifiée par une instruction CN.

Syntaxe

`S...` / `S<n>=...`

`M3` / `M<n>=3`

`M4` / `M<n>=4`

`M5` / `M<n>=5`

`SETMS(<n>)`	
`...`	
`SETMS`	

Signification

`S...` :	Vitesse de rotation en tours/min pour la broche maître
`S<n>=...` :	Vitesse de rotation en tours/min pour broche <n>

	Remarque : La vitesse de rotation indiquée par S0=... s'applique à la broche maître.
`M3` :	Sens de rotation horaire pour broche maître
`M<n>=3` :	Sens de rotation horaire pour broche <n>
`M4` :	Sens de rotation antihoraire pour broche maître
`M<n>=4` :	Sens de rotation antihoraire pour broche <n>
`M5` :	Arrêt de broche pour broche maître
`M<n>=5` :	Arrêt de broche pour broche <n>
`SETMS(<n>)` :	La broche <n> doit faire office de broche maître
`SETMS` :	SETMS sans indication de broche entraîne un retour à la broche maître configurée

Remarque

Au maximum 3 valeurs S sont programmables dans un bloc CN, p. ex. :

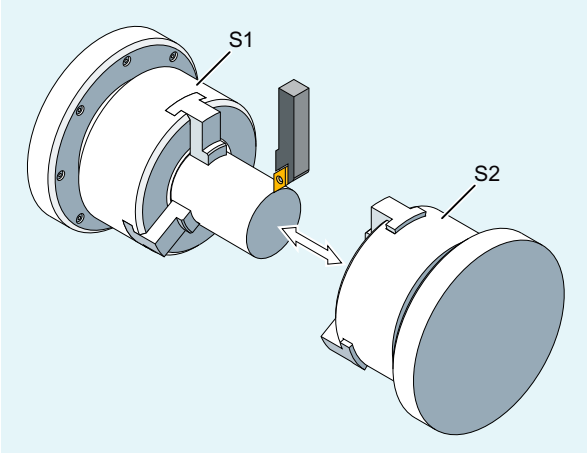
```
S... S2=... S3=...
```

Remarque

SETMS doit figurer dans un bloc spécifique.

Exemple

S1 est la broche maître, S2 est la seconde broche. La pièce tournée doit être usinée des deux côtés. Pour ce faire, une division des étapes est nécessaire. Après tronçonnage, le dispositif de synchronisation (S2) prend la pièce pour que l'usinage soit effectué côté tronçonné. Pour ce faire, cette broche S2 est définie comme broche maître et G95 s'applique alors à celle-ci.



Code de programme	**Commentaire**
`N10 S300 M3`	`; Vitesse et sens de rotation pour la broche d'entraînement = broche maître préréglée.`
`...`	`; Usinage du côté droit de la pièce.`
`N100 SETMS(2)`	`; S2 est maintenant la broche maître.`
`N110 S400 G95 F…`	`; Vitesse pour la nouvelle broche maître.`

Code de programme	Commentaire
...	; Usinage du côté gauche de la pièce.
N160 SETMS	; Retour à la broche maître S1.

Informations complémentaires

Interprétation de la valeur S pour la broche maître

Lorsque les fonctions G331 ou G332 sont actives dans le groupe G 1 (instructions de déplacement à effet modal), la valeur S programmée est toujours interprétée comme vitesse de rotation en tours/min. Sinon, l'interprétation de la valeur S dépend du groupe G 15 (mode d'avance) : lorsque G96, G961 ou G962 sont actifs, la valeur S est interprétée comme vitesse de coupe constante en m/min, dans les autres cas comme vitesse de rotation en tours/min.

En cas de changement de G96/G961/G962 à G331/G332, la valeur de la vitesse de coupe constante est réglée à zéro, en cas de changement de G331/G332 à une fonction différente de G331/G332 dans le groupe G 1, la valeur de la vitesse de rotation est réglée à zéro. Le cas échéant, les valeurs S concernées doivent être reprogrammées.

Fonctions M préréglées M3, M4, M5

Dans un bloc contenant des instructions d'axe, les fonctions M3, M4, M5 sont activées **avant** les déplacements d'axe (préréglage de la commande).

Exemple :

Code de programme	Commentaire
N10 G1 F500 X70 Y20 S270 M3	; La broche accélère jusqu'à 270 tr/min, puis les déplacements en X et Y sont exécutés.
N100 G0 Z150 M5	; Arrêt de la broche avant le déplacement de retrait en Z.

Remarque

Dans les paramètres machine, vous réglez si les déplacements d'axe doivent être effectués seulement après accélération de la broche, quand elle a atteint la valeur de consigne de vitesse, ou s'est arrêtée, ou bien immédiatement après les opérations de commutation programmées.

Usinage avec plusieurs broches

5 broches peuvent se trouver simultanément dans un canal (la broche maître plus 4 autres broches).

Une seule broche est définie par paramètre machine comme **broche maître**. Cette broche obéit à des fonctions spéciales comme par exemple le filetage à l'outil, le taraudage, l'avance par tour, l'arrêt temporisé. Pour les autres broches (p. ex. une seconde broche et un outil motorisé), il est nécessaire d'indiquer les numéros correspondants pour la vitesse et le sens de rotation/arrêt de la broche.

Exemple :

Code de programme	Commentaire
`N10 S300 M3 S2=780 M2=4`	`; Broche maître : 300 tr/min, rotation en sens horaire` `2nde broche : 780 tr/min, rotation en sens anti-horaire`

Basculement programmable de la broche maître

L'instruction `SETMS(<n>)` permet de définir toute broche comme broche maître dans le programme CN. `SETMS` doit figurer dans un bloc spécifique.

Exemple :

Code de programme	Commentaire
`N10 SETMS(2)`	`; La broche 2 est maintenant la broche maître.`

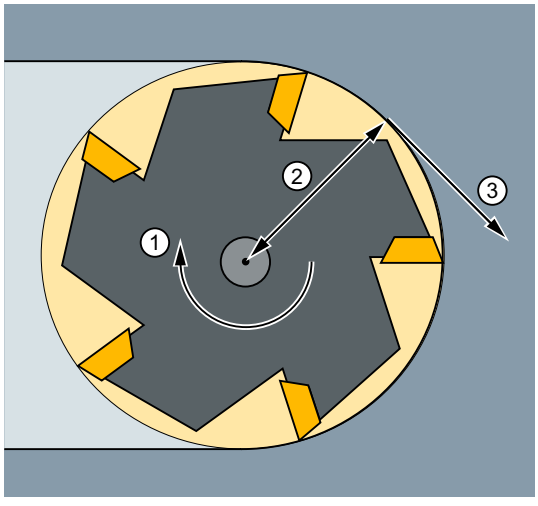
Remarque

La vitesse de rotation indiquée par `S...` ainsi que les fonctions programmées avec `M3`, `M4`, `M5` s'appliquent maintenant à la nouvelle broche maître déclarée.

Avec `SETMS` sans indication de broche, le retour à la broche maître définie dans les paramètres machine est réalisé.

2.6.2 Vitesse de coupe d'outil (SVC)

Outre la vitesse de rotation de la broche, il est également possible de programmer la vitesse de coupe de l'outil qui est utilisée en pratique pour le fraisage.



① Vitesse de rotation de la broche
② Rayon d'outil
③ Vitesse de coupe d'outil

En utilisant le rayon de l'outil actif, la commande calcule la vitesse de rotation active de la broche à partir de la vitesse de coupe programmée pour l'outil :

$S = (SVC * 1000) / (R_{Out.} * 2\pi)$

où : S : Vitesse de rotation de la broche en tr/min
SVC : Vitesse de coupe d'outil en m/min ou ft/min
$R_{Out.}$: Rayon de l'outil actif en mm

Le type ($TC_DP1) de l'outil actif n'est pas pris en compte.

La vitesse de coupe d'outil programmée est indépendante de l'avance tangentielle F ainsi que du groupe fonctionnel G 15 (type d'avance). Le sens de rotation et le départ de la broche sont définis par M3 ou M4, et l'arrêt de la broche par M5.

Une modification des paramètres du rayon de l'outil dans la mémoire de corrections prend effet à la prochaine sélection de la correction de l'outil ou à la prochaine actualisation des paramètres de correction actifs.

Un changement d'outil et une sélection/désélection d'un enregistrement de correction d'outil entraînent un nouveau calcul de la vitesse de rotation active de la broche.

Conditions

La programmation de la vitesse de coupe d'outil nécessite :

- les rapports géométriques d'un outil en rotation (outil de fraisage ou de perçage)
- un enregistrement de correction d'outil actif

Syntaxe

```
T... D... SVC[<n>]=<Value>
...
S... M3/M4
```

Signification

SVC :	Mot clé pour la programmation de la vitesse de coupe d'outil	
[<n>] :	Numéro de la broche Cette extension d'adresse indique la broche pour laquelle la vitesse de coupe programmée doit être active. En l'absence d'une extension d'adresse, l'indication se réfère toujours à la broche maître actuelle. **Remarque :** Il est possible de présélectionner une vitesse de coupe propre à chaque broche. **Remarque :** La programmation de SVC sans extension d'adresse suppose que la broche maître dispose de l'outil actif. En cas de changement de la broche maître, l'utilisateur doit sélectionner un outil correspondant.	
<Value> :	Valeur de la vitesse de coupe d'outil	
	Unité :	m/min (pour G71/G710) ou ft/min (pour G70/G700)

`T... D...:`	Le rayon de l'outil doit être connu dans l'enregistrement contenant SVC, c'est-à-dire qu'un outil correspondant, y compris l'enregistrement de correction de l'outil, doit être actif ou sélectionné dans l'enregistrement. L'ordre de sélection de SVC et de T/D est quelconque lors de la programmation dans le même bloc.
`S... M3/M4:`	La programmation de la vitesse de rotation de broche entraîne la désactivation de la vitesse de coupe d'outil. **Remarque :** Un changement entre la programmation SVC et S est toujours possible, même lorsque la broche est en rotation. La valeur non active respective est supprimée.

Remarque

La programmation de SVC n'est pas possible pour certains mouvements d'avance actifs de la broche, à savoir :

- Vitesse de coupe constante : G96/G961/G962 S... (Page 100)
- Vitesse circonférentielle de meule constante : VCM (Page 106)
- Positionnement de la broche : SPOS/SPOSA/M19 (Page 121)
- Basculement de la broche maître en mode axe : M70 (Page 121)

Inversement, la programmation de l'une de ces fonction entraîne la désélection de SVC (vitesse de coupe d'outil).

Remarque

Vitesse de rotation d'outil maximale

La variable système $TC_TP_MAX_VELO[<numéro T>] permet de présélectionner une vitesse de rotation d'outil maximale (vitesse de broche).
Si aucune limite de vitesse n'est définie, aucune surveillance n'est réalisée.

Remarque

Les trajectoires "d'outils normalisés", générées p. ex. par des systèmes CAD, qui prennent déjà en compte le rayon de l'outil et qui intègrent la différence par rapport à l'outil normalisé uniquement dans le rayon de coupe, ne sont pas prises en charge dans le cadre de la programmation SVC.

Exemples

Pour chaque exemple, est valable : porte-outil = broche (dans le cas du fraisage standard)

Exemple 1 : fraise de rayon 6 mm

Code de programme	Commentaire
`N10 G0 X10 T1 D1`	`; Sélection de l'outil de fraisage avec p. ex. $TC_DP6[1,1] = 6 (rayon d'outil = 6 mm)`

`N20 SVC=100 M3`	`; Vitesse de coupe = 100 m/min`
	`⇒ Vitesse de rotation de broche résultante :`
	`S = (100 m/min * 1000) / (6,0 mm * 2 * 3,14) = 2653,93 tr/min`

`N30 G1 X50 G95 FZ=0.03`	`; SVC et avance par dent`

Code de programme	Commentaire
...	

Exemple 2 : sélection d'outil et de SVC dans le même bloc

Code de programme	Commentaire
N10 G0 X20	
N20 T1 D1 SVC=100	; Sélection d'outil et d'enregistrement de correction avec SVC dans le bloc (ordre quelconque).
N30 X30 M3	; Type de broche avec sens de rotation horaire, vitesse de coupe 100 m/min
N40 G1 X20 F0.3 G95	; SVC et avance par tour

Exemple 3 : présélection des vitesses de coupe pour deux broches

Code de programme	Commentaire
N10 SVC[3]=100 M6 T1 D1	
N20 SVC[5]=200	; Le rayon de l'outil de la correction d'outil active est identique pour les deux broches, la vitesse de rotation effective est différente pour la broche 3 et la broche 5.

Exemple 4 :

Hypothèses :

le maître pour le changement d'outil est déterminé par les porte-outils :

PM20124 $MC_TOOL_MANAGEMENT_TOOLHOLDER > 1

Lors du changement d'outil, l'ancienne correction d'outil est conservée et une correction du nouvel outil ne devient active qu'avec la programmation de `D` :

PM20270 $MC_CUTTING_EDGE_DEFAULT = - 2

Code de programme	Commentaire
N10 $TC_MPP1[9998,1]=2	; Le magasin d'outils est le porte-outil
N11 $TC_MPP5[9998,1]=1	; Le magasin d'outils est le porte-outil 1
N12 $TC_MPP_SP[9998,1]=3	; Le porte-outil 1 est affecté à la broche 3
N20 $TC_MPP1[9998,2]=2	; Le magasin d'outils est le porte-outil
N21 $TC_MPP5[9998,2]=4	; Le magasin d'outils est le porte-outil 4
N22 $TC_MPP_SP[9998,2]=6	; Le porte-outil 4 est affecté à la broche 6
N30 $TC_TP2[2]="WZ2"	
N31 $TC_DP6[2,1]=5.0	; Rayon = 5,0 mm de T2, correction D1
N40 $TC_TP2[8]="WZ8"	
N41 $TC_DP6[8,1]=9.0	; Rayon = 9,0 mm de T8, correction D1
N42 $TC_DP6[8,4]=7.0	; Rayon = 7,0 mm de T8, correction D4
...	
N100 SETMTH(1)	; Définir le numéro du porte-outil maître
N110 T="WZ2" M6 D1	; L'outil T2 est mis en place et la correction D1 est activée.
N120 G1 G94 F1000 M3=3 SVC=100	; S3 = (100 m/min * 1000) / (5,0 mm * 2 * 3,14) = 3184,71 tr/min

Code de programme	Commentaire
N130 SETMTH(4)	; Définir le numéro du porte-outil maître
N140 T="WZ8"	; Correspond à T8="WZ8"
N150 M6	; Correspond à M4=6
	L'outil "WZ8" vient sur le porte-outil maître, mais puisque PM20270=–2, l'ancienne correction d'outil reste active.
N160 SVC=50	; S3 = (50 m/min * 1000) / (5,0 mm * 2 * 3,14) = 1592,36 tr/min
	La correction du porte-outil 1 est encore active et ce dernier est affecté à la broche 3.
N170 D4	; La correction D4 du nouvel outil "WZ8" est activée (sur le porte-outil 4).
N180 SVC=300	; S6 = (300 m/min * 1000) / (7,0 mm * 2 * 3,14) = 6824,39 tr/min
	La broche 6 est affectée au porte-outil 4.

Exemple 5 :

Hypothèses :

Les broches sont également des porte-outils :

PM20124 $MC_TOOL_MANAGEMENT_TOOLHOLDER = 0

Lors d'un changement d'outil, l'enregistrement de correction d'outil D4 est automatiquement sélectionné :

PM20270 $MC_CUTTING_EDGE_DEFAULT = 4

Code de programme	Commentaire
N10 $TC_MPP1[9998,1]=2	; Le magasin d'outils est le porte-outil
N11 $TC_MPP5[9998,1]=1	; Le magasin d'outils est le porte-outil 1 = broche 1
N20 $TC_MPP1[9998,2]=2	; Le magasin d'outils est le porte-outil
N21 $TC_MPP5[9998,2]=3	; Le magasin d'outils est le porte-outil 3 = broche 3
N30 $TC_TP2[2]="WZ2"	
N31 $TC_DP6[2,1]=5.0	; Rayon = 5,0 mm de T2, correction D1
N40 $TC_TP2[8]="WZ8"	
N41 $TC_DP6[8,1]=9.0	; Rayon = 9,0 mm de T8, correction D1
N42 $TC_DP6[8,4]=7.0	; Rayon = 7,0 mm de T8, correction D4
...	
N100 SETMS(1)	; Broche 1 = broche maître
N110 T="WZ2" M6 D1	; L'outil T2 est mis en place et la correction D1 est activée.
N120 G1 G94 F1000 M3 SVC=100	; S1 = (100 m/min * 1000) / (5,0 mm * 2 * 3,14) = 3184,71 tr/min
N200 SETMS(3)	; Broche 3 = broche maître
N210 M4 SVC=150	; S3 = (150 m/min * 1000) / (5,0 mm * 2 * 3,14) = 4777,07 tr/min
	Se rapporte à la correction d'outil D1 de T="WZ2", la rotation de S1 se poursuit avec l'ancienne vitesse de rotation.
N220 T="WZ8"	; Correspond à T8="WZ8"

Code de programme	Commentaire
N230 M4 SVC=200	; S3 = (200 m/min * 1000) / (5,0 mm * 2 * 3,14) = 6369,43 tr/min
	Se rapporte à la correction d'outil D1 de T="WZ2".
N240 M6	; Correspond à M3=6
	L'outil "WZ8" vient sur la broche maître, la correction d'outil D4 du nouvel outil devient active.
N250 SVC=50	; S3 = (50 m/min * 1000) / (7,0 mm * 2 * 3,14) = 1137,40 tr/min
	La correction D4 sur la broche maître est active.
N260 D1	; La correction D1 du nouvel outil "WZ8" est active.
N270 SVC[1]=300	; S1 = (300 m/min * 1000) / (9,0 mm * 2 * 3,14) = 5307,86 tr/min
	S3 = (50 m/min * 1000) / (9,0 mm * 2 * 3,14) = 884,64 tr/min
...	

plus d'informations

Rayon de l'outil

Les paramètres suivants de correction de l'outil (actif) ont une incidence sur le rayon de l'outil :

- $TC_DP6 (rayon - géométrie)
- $TC_DP15 (rayon - usure)
- $TC_SCPx6 (correction de $TC_DP6)
- $TC_ECPx6 (correction de $TC_DP6)

Ne sont pas pris en compte :

- les corrections d'outil en ligne
- La surépaisseur d'usinage par rapport au contour programmé (OFFN)

Correction du rayon d'outil (G41/G42)

La correction du rayon d'outil (G41/G42) et la vitesse de coupe d'outil (SVC) se rapportent toutes deux au rayon de l'outil, mais sont fonctionnellement découplées et indépendantes l'une de l'autre.

Taraudage sans porte-taraud compensateur, (G331, G332)

La programmation de SVC est également possible en association avec G331 ou G332.

Actions synchrones

La spécification de SVC n'est pas possible à partir d'actions synchrones.

Lecture de la vitesse de coupe et de la variante de programmation pour la vitesse de rotation de la broche

La lecture de la vitesse de coupe d'une broche et la variante de programmation pour la vitesse de rotation (vitesse de broche S ou vitesse de coupe d'outil SVC) peuvent être réalisées au moyen de variables système :

- Avec arrêt du prétraitement des blocs dans le programme pièce, au moyen des variables système :

$AC_SVC[<n>]	Vitesse de coupe active lors du traitement du bloc principal actuel pour la broche de numéro <n>.
$AC_S_TYPE[<n>]	Variante de programmation pour la vitesse de rotation de broche, active lors du traitement du bloc principal actuel pour la broche de numéro <n>.

Valeur :	**Signification** :
1	Vitesse de rotation de broche S en tr/min
2	Vitesse de coupe SVC en m/min ou ft/min

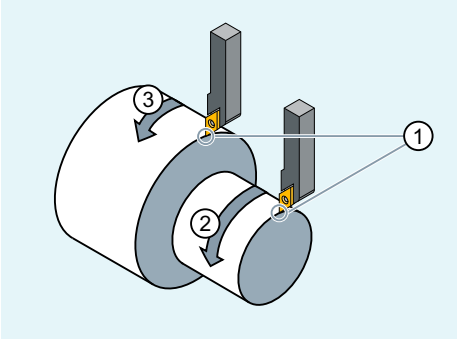
- Sans arrêt du prétraitement des blocs dans le programme pièce, au moyen des variables système :

$P_SVC[<n>]	Vitesse de coupe programmée pour la broche <n>
$P_S_TYPE[<n>]	Variante de programmation pour la vitesse de rotation programmée pour la broche <n>

Valeur :	**Signification** :
1	Vitesse de rotation de broche S en tr/min
2	Vitesse de coupe SVC en m/min ou ft/min

2.6.3 Vitesse de coupe constante (G96/G961/G962, G97/G971/G972, G973, LIMS, SCC)

Lorsque la fonction "Vitesse de coupe constante" est active, la vitesse de rotation de la broche est modifiée en fonction du diamètre de la pièce, de sorte que la vitesse de coupe S en m/min ou ft/min reste constante au niveau du tranchant de l'outil.



① Vitesse de coupe constante
② Vitesse de rotation de broche augmentée
③ Vitesse de rotation de broche réduite

Il en résulte les avantages suivants :

- Des profils de tournage réguliers et, par conséquent, une qualité de surface élevée
- Un traitement ménageant les outils

Syntaxe

Activation/désactivation de la vitesse de coupe constante pour la broche maître :

```
G96/G961/G962 S...
...
G97/G971/G972/G973
```

Limitation de la vitesse de rotation pour la broche maître :
`LIMS=<valeur>`
`LIMS[<broche>]=<valeur>`

Autre axe de référence pour G96/G961/G962 :
`SCC[<axe>]`

Remarque

SCC[<axe>] peut être programmé séparément ou avec G96/G961/G962.

Signification

`G96` :	Avance par tour (comme G95 (Page 108)) et vitesse de coupe constante G96 active automatiquement G95. Si G95 n'était pas encore activé, une nouvelle valeur d'avance F... doit être indiquée à l'appel de G96.	
`G961` :	Avance linéaire (comme G94 (Page 108)) et vitesse de coupe constante	
`G962` :	Avance linéaire ou avance par tour et vitesse de coupe constante	
`S...` :	En association avec G96, G961 ou G962, S... n'est pas interprétée comme vitesse de rotation de broche, mais comme vitesse de coupe. La vitesse de coupe s'applique toujours à la broche maître.	
	Unité :	m/min (pour G71/G710) ou ft/min (pour G70/G700)
	Plage de valeurs :	0,1 m/min ... 9999 9999,9 m/min
`G97` :	Avance par tour et vitesse de rotation de broche constante (vitesse de coupe constante désactivée)	
`G971` :	Avance linéaire et vitesse de rotation de broche constante (vitesse de coupe constante désactivée)	
`G972` :	Avance linéaire ou avance par tour et vitesse de rotation de broche constante (vitesse de coupe constante désactivée)	
`G973` :	Avance par tour sans limitation de vitesse de la broche et vitesse de rotation de broche constante (G97 sans LIMS pour mode ISO)	
	Remarque : Après G97 (ou G971 ... G973), S... est de nouveau interprétée comme vitesse de rotation de broche en tours/min. Si aucune nouvelle vitesse de rotation de broche n'est indiquée, la dernière vitesse de rotation réglée par G96 (ou G961 ou G962) est conservée.	

LIMS :	Limitation de la vitesse de rotation pour la broche maître (uniquement effective si G96/G961/G97 sont actifs) Pour les machines avec broche maître permutable, il est possible de programmer jusqu'à 4 limitations de broches avec différentes valeurs dans un bloc.	
	<broche> :	Numéro de la broche
	<valeur> :	Limite supérieure de la vitesse de rotation de broche en tours/min
SCC :	Lorsqu'une fonction G96/G961/G962 est active, SCC[<axe>] permet d'affecter un axe géométrique au choix comme axe de référence.	

Remarque

Lors de la première sélection de G96/G961/G962, une vitesse de coupe constante S... doit être spécifiée ; lors des sélections suivantes de G96/G961/G962, cette indication est optionnelle.

Remarque

La limitation de la vitesse de rotation programmée avec LIMS ne doit pas dépasser la vitesse de rotation limite programmée avec G26 ou définie dans les données de réglage.

Remarque

L'axe de référence pour G96/G961/G962 doit, au moment de la programmation de SCC[<axe>], être un axe géométrique connu dans le canal. La programmation de SCC[<axe>] est possible également lorsque G96/G961/G962 est actif.

Exemples

Exemple 1 : activation de la vitesse de coupe constante avec limitation de la vitesse de rotation

```
Code de programme                              Commentaire
N10 SETMS(3)
N20 G96 S100 LIMS=2500                         ; Vitesse de coupe constante = 100 m/
                                               min, vitesse de rotation max. = 2500 tr/
                                               min
...
N60 G96 G90 X0 Z10 F8 S100 LIMS=444            ; Vitesse de rotation max. = 444 tr/min
```

Exemple 2 : présélection de la limitation de vitesse de rotation pour 4 broches

Les limitations de vitesse de rotation sont définies pour la broche 1 (broche maître) et les broches 2, 3 et 4 :

```
Code de programme
N10 LIMS=300 LIMS[2]=450 LIMS[3]=800
LIMS[4]=1500
...
```

Exemple 3 : affectation d'un axe Y pour un traitement d'usinage avec l'axe X

Code de programme	Commentaire
N10 G18 LIMS=3000 T1 D1	; Vitesse limitée à 3000 tours/min.
N20 G0 X100 Z200	
N30 Z100	
N40 G96 S20 M3	; Vitesse de coupe constante = 20 m/min, fonction de l'axe X.
N50 G0 X80	
N60 G1 F1.2 X34	; Surfaçage en X à 1,2 mm/tour.
N70 G0 G94 X100	
N80 Z80	
N100 T2 D1	
N110 G96 S40 SCC[Y]	; L'axe Y est affecté à G96 et G96 est actif (possible dans un bloc). Vitesse de coupe constante = 40 m/min, fonction de l'axe Y.
...	
N140 Y30	
N150 G01 F1.2 Y=27	; Plongée dans Y, avance F = 1,2 mm/tour.
N160 G97	; Désactivation de la vitesse de coupe constante.
N170 G0 Y100	

Informations complémentaires

Calcul de la vitesse de rotation de broche

La base pour le calcul de la vitesse de rotation de broche à partir de la vitesse de coupe programmée est la position ENS de l'axe transversal (rayon).

Remarque

Les frames entre SCP et ENS (par ex. les frames programmables tels que SCALE, TRANS ou ROT) sont pris en compte lors du calcul de la vitesse de rotation de broche et peuvent entraîner une modification de la vitesse de rotation (p. ex. lorsque le diamètre actif est modifié dans SCALE).

Limitation de la vitesse de rotation LIMS

Pour usiner une pièce présentant des différences de diamètre importantes, il est recommandé d'indiquer une limitation de vitesse de rotation de la broche avec LIMS (vitesse de rotation maximale de la broche). Ceci permet d'exclure les vitesses excessives pour les faibles diamètres. LIMS est uniquement effective si G96, G961 et G97 sont actifs. Pour G971, LIMS n'est pas

effective. Lors du chargement du bloc dans l'exécution, toutes les valeurs programmées dans les données de réglage sont reprises.

Remarque

Les limites de vitesse modifiées dans le programme pièce avec LIMS sont reprises dans les données de réglage et restent donc enregistrées même après la fin du programme.

Si les limites de vitesses modifiées avec LIMS ne doivent plus être valables après la fin du programme, la définition suivante doit être insérée dans le bloc GUD du constructeur de machines :

REDEF $SA_SPIND_MAX_VELO_LIMS PRLOC

Désactivation de la vitesse de coupe constante (G97/G971/G972/G973)

Après G97 (ou G971 ... G973), S... est de nouveau interprétée comme vitesse de rotation de broche en tours/min. Si aucune nouvelle vitesse de rotation de broche n'est indiquée, la dernière vitesse de rotation réglée par G96 (ou G961 ou G962) est conservée.

La fonction G96/G961 peut également être désactivée avec G94 ou G95. Dans ce cas, c'est la dernière vitesse de rotation S... programmée qui sera utilisée pour poursuivre l'usinage.

G97 peut être programmé sans G96 préalable. La fonction agit alors comme G95, vous pouvez par ailleurs programmer LIMS.

G973 permet de désactiver la vitesse de coupe constante sans activer une limitation de vitesse de rotation de la broche.

Remarque

L'axe transversal doit être défini dans les paramètres machine.

Déplacement à vitesse rapide G0

En cas de déplacement à vitesse rapide G0, il n'y a pas de modification de la vitesse de rotation.

Exception :

Si le contour est accosté à vitesse rapide et si le bloc CN suivant contient une instruction d'interpolation G1/G2/G3/..., la vitesse de rotation s'adapte à l'instruction d'interpolation suivante dès le bloc d'accostage G0.

Autre axe de référence pour G96/G961/G962

Lorsqu'une fonction G96/G961/G962 est active, SCC[<axe>] permet d'affecter un axe géométrique au choix comme axe de référence. Si l'axe de référence est modifié et donc la position de référence de la pointe d'outil (TCP - Tool Center Point) pour la vitesse de coupe constante, la vitesse de rotation de broche qui en résulte est accostée via la rampe de freinage ou d'accélération réglée.

Permutation de l'axe de canal affecté

La propriété Axe de référence pour G96/G961/G962 est toujours affectée à un axe géométrique. Lors de la permutation de l'axe de canal affecté, la propriété Axe de référence pour G96/G961/G962 reste dans l'ancien canal.

Une permutation d'axe géométrique n'influence pas l'affectation de l'axe géométrique à la vitesse de coupe constante. Si une permutation d'axe géométrique modifie la position de

référence TCP pour G96/G961/G962, la broche se déplace via la rampe avec la nouvelle vitesse de rotation.

Si aucun nouvel axe de canal n'est affecté par permutation d'axe géométrique (p. ex. GEOAX(0,X)), la vitesse de rotation de broche est gelée en fonction de G97.

Exemples de permutations d'axe géométrique avec affectations de l'axe de référence :

Code de programme	**Commentaire**
N05 G95 F0.1	
N10 GEOAX(1, X1)	; L'axe de canal X1 devient le premier axe géométrique.
N20 SCC[X]	; Le premier axe géométrique (X) devient l'axe de référence
	; pour G96/G961/G962.
N30 GEOAX(1, X2)	; L'axe de canal X2 devient le premier axe géométrique.
N40 G96 M3 S20	; L'axe de référence pour G96 est l'axe de canal X2.

Code de programme	**Commentaire**
N05 G95 F0.1	
N10 GEOAX(1, X1)	; L'axe de canal X1 devient le premier axe géométrique.
N20 SCC[X1]	; X1 et implicitement le premier axe géométrique (X) devient
	;l'axe de référence pour G96/G961/G962.
N30 GEOAX(1, X2)	; L'axe de canal X2 devient le premier axe géométrique.
N40 G96 M3 S20	; L'axe de référence pour G96 est X2 ou X, pas d'alarme.

Code de programme	**Commentaire**
N05 G95 F0.1	
N10 GEOAX(1, X2)	; L'axe de canal X2 devient le premier axe géométrique.
N20 SCC[X1]	; X1 n'est pas un axe géométrique, alarme.

Code de programme	**Commentaire**
N05 G0 Z50	
N10 X35 Y30	
N15 SCC[X]	; L'axe de référence pour G96/G961/G962 est X.
N20 G96 M3 S20	; Vitesse de coupe constante activée à 10 mm/min.
N25 G1 F1.5 X20	; Surfaçage en X à 1,5 mm/tour.
N30 G0 Z51	
N35 SCC[Y]	; L'axe de référence pour G96 est Y,
	; Réduction de la vitesse de rotation de la broche (Y30).
N40 G1 F1.2 Y25	; Surfaçage en Y à 1,2 mm/tour.

2.6.4 Activer/désactiver la vitesse circonférentielle de meule constante (GWPSON, GWPSOF)

Les procédures GWPSON(...) et GWPSOF(...) prédéfinies permettent d'activer ou de désactiver la vitesse circonférentielle de meule constante (VCM) pour les outils de rectification (type d'outil : 400 à 499).

Syntaxe

```
GWPSON(<TNr>)
S<n>=...
...
GWPSOF(<TNr>)
```

Signification

`GWPSON(...)` :	Activer la vitesse circonférentielle de meule constante
`GWPSOF(...)` :	Désactiver la vitesse circonférentielle de meule constante
`<TNr>` :	Numéro T **Remarque :** Requis uniquement lorsque la vitesse circonférentielle de meule constante doit être activée ou désactivée pour une meule non active à la place de l'outil actif mis en œuvre.
`S<n>=...` :	Vitesse circonférentielle de meule en m/s ou pied/s pour la broche <n>
`S0=...` ou `S...` :	Vitesse circonférentielle de meule pour la broche maître

Interrogation de l'état

Les variables système suivantes permettent d'interroger, dans le programme pièce, si la vitesse circonférentielle de meule constante est active pour une broche donnée :

$P_GWPS[<n>] ; où <n> = numéro de broche

Valeur	Signification
0 (= FALSE)	VCM est **désactivée**.
1 (= TRUE)	VCM est **activée**.

2.6.5 Limitation programmable de la vitesse de rotation de broche (G25, G26)

Les vitesses de rotation minimales et maximales de la broche, définies dans les paramètres et données de réglage machine, peuvent être modifiées par une instruction du programme pièce.

Des limitations des vitesses de rotation sont programmables pour toutes les broches du canal.

Syntaxe

```
G25 S... S1=... S2=...
```

```
G26 S... S1=... S2=...
```

Signification

`G25` :	Limitation **inférieure** de la vitesse de rotation de la broche
`G26` :	Limitation **supérieure** de la vitesse de rotation de la broche
`S... S1=... S2=...` :	Vitesse(s) de rotation minimale(s) ou maximale(s) de la broche **Remarque :** Au maximum trois limitations de vitesse de rotation de la broche peuvent être programmées par bloc. Plage de valeurs : 0,1 ... 9999 9999,9 tr/min

Remarque

Une limitation de vitesse de rotation de la broche programmée avec `G25` ou `G26` écrase les vitesses de rotation limites dans les données de réglage et, par conséquent, reste en mémoire également après la fin du programme.

Si les limites de vitesses modifiées avec `G25`/`G26` ne doivent plus être valables après la fin du programme, les définitions suivantes doivent être insérées dans le bloc GUD du constructeur de machines :

REDEF $SA_SPIND_MIN_VELO_G25 PRLOC

REDEF $SA_SPIND_MAX_VELO_G26 PRLOC

Exemple

Code de programme	Commentaire
`N10 G26 S1400 S2=350 S3=600`	`; Vitesses limites supérieures pour la broche maître, la broche 2 et la broche 3`

2.7 Régulation d'avance

2.7.1 Avance (G93, G94, G95, F, FGROUP, FL, FGREF)

Ces instructions permettent de régler dans le programme CN les vitesses d'avance pour tous les axes concernés par l'usinage.

Syntaxe

```
G93
G94
G95
F<valeur>
FGROUP(<axe_1>,<axe_2>,…)
FGREF[<axe rotatif>]=<rayon de référence>
FL[<axe>]=<valeur>
```

Signification

`G93` :	Type de l'avance tangentielle : avance en sens inverse du temps [tr/min]
`G94` :	Type de l'avance tangentielle : avance linéaire [mm/min], [inch/min] ou [degré/min]
`G95` :	Type de l'avance tangentielle : avance par tour [mm/tour] ou [inch/tour] L'avance par tour peut être dérivée au choix à partir d'une broche maître, d'une autre broche choisie ou d'un axe rotatif.
`F<valeur>`	Avance tangentielle pour tous les axes à interpolation ou pour ceux sélectionnés avec FGROUP.
`FGROUP` :	définition des axes à interpolation auxquels se rapporte l'avance tangentielle programmée sous F.
`FGREF` :	`FGREF` permet de programmer le rayon effectif (`<rayon de référence>`) pour chaque axe rotatif indiqué sous `FGROUP`
`FL` :	Vitesse limite pour axes synchrones/à interpolation L'unité réglée avec `G94` s'applique. Il est possible de programmer une valeur `FL` par axe (axe de canal, axe géométrique ou axe d'orientation).
`<axe>` :	nom d'un axe de canal, type : AXIS

Exemples

Exemple 1 : mode d'action de FGROUP

L'exemple suivant a pour objet d'expliciter l'effet de `FGROUP` sur le trajet et l'avance tangentielle. La variable `$AC_TIME` contient le temps écoulé depuis le début d'un bloc en secondes. Elle ne peut être utilisée que dans les actions synchrones.

Code de programme	**Commentaire**
`N100 G0 X0 A0`	
`N110 FGROUP(X,A)`	

Code de programme	Commentaire
N120 G91 G1 G710 F100	; Avance = 100 mm/min ou 100 degrés/min
N130 DO $R1=$AC_TIME	
N140 X10	; Avance = 100 mm/min, trajet = 10 mm, R1 = env. 6 s
N150 DO $R2=$AC_TIME	
N160 X10 A10	; Avance = 100 mm/min, trajet = 14,14 mm, R2 = env. 8 s
N170 DO $R3=$AC_TIME	
N180 A10	; Avance = 100 degrés/min, trajet = 10 degrés, R3 = env. 6 s
N190 DO $R4=$AC_TIME	
N200 X0.001 A10	; Avance = 100 mm/min, trajet = 10 mm, R4= env. 6 s
N210 G700 F100	; Avance = 2540 mm/min ou 100 degrés/min
N220 DO $R5=$AC_TIME	
N230 X10	; Avance = 2540 mm/min, trajet = 254 mm, R5= env. 6 s
N240 DO $R6=$AC_TIME	
N250 X10 A10	; Avance = 2540 mm/min, trajet = 254,2 mm, R6 = env. 6 s
N260 DO $R7=$AC_TIME	
N270 A10	; Avance = 100 degrés/min, trajet = 10 degrés, R7 = env. 6 s
N280 DO $R8=$AC_TIME	
N290 X0.001 A10	; Avance = 2540 mm/min, trajet = 10 mm, R8= env. 0,288 s
N300 FGREF[A]=360/(2*$PI)	; Réglage 1 degré = 1 inch à l'aide du rayon effectif
N310 DO $R9=$AC_TIME	
N320 X0.001 A10	; Avance = 2540 mm/min, trajet = 254 mm, R9 = env. 6 s
N330 M30	

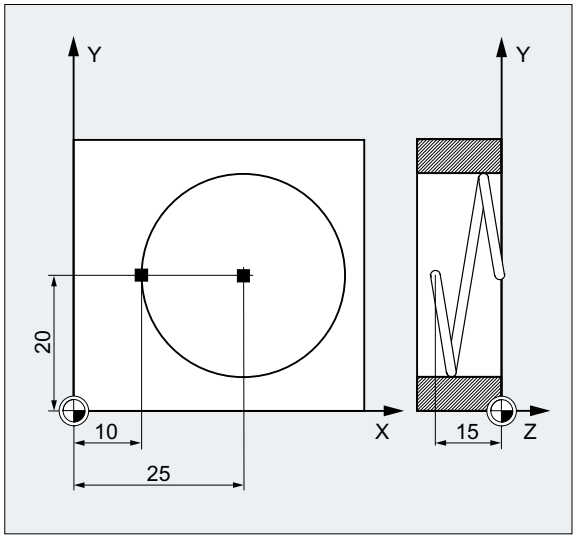
Exemple 2 : déplacement des axes synchrones avec vitesse limite FL

La vitesse tangentielle des axes à interpolation diminue quand l'axe synchrone Z atteint la vitesse limite.

Code de programme
N10 G0 X0 Y0
N20 FGROUP(X)
N30 G1 X1000 Y1000 G94 F1000 FL[Y]=500
N40 Z-50

Exemple 3 : Interpolation hélicoïdale

Les axes à interpolation X et Y se déplacent avec l'avance programmée, l'axe de pénétration Z est l'axe synchrone.

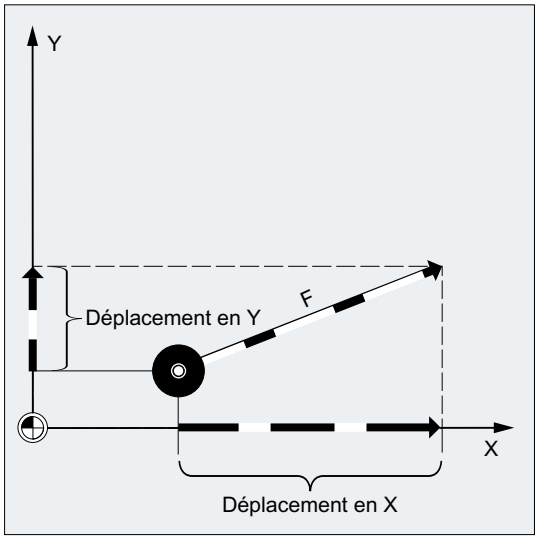


Code de programme	Commentaire
N10 G17 G94 G1 Z0 F500	; Approche de l'outil
N20 X10 Y20	; Accostage de la position de départ
N25 FGROUP(X,Y)	; Les axes X/Y sont des axes à interpolation, Z est un axe synchrone
N30 G2 X10 Y20 Z-15 I15 J0 F1000 FL[Z]=200	; Sur la trajectoire circulaire, l'avance est égale à 1000 mm/min, en direction Z, le déplacement est synchrone
...	
N100 FL[Z]=$MA_AX_VELO_LIMIT[0,Z]	; La lecture de la vitesse dans le PM désactive la vitesse limite qui est remplacée par la valeur lue
N110 M30	; Fin du programme.

plus d'informations

Vitesse d'avance pour axes à interpolation (F)

En règle générale, l'avance tangentielle est constituée des différentes composantes de vitesse de tous les axes géométriques impliqués dans le déplacement et se réfère au centre de la fraise ou à la pointe de l'outil de tournage.



La vitesse d'avance est indiquée sous l'adresse `F`. En fonction du préréglage dans les paramètres machine, les unités de mesure déterminées par les instructions G s'appliquent en mm ou inch.

Une seule valeur `F` est programmable par bloc CN. L'unité de la vitesse d'avance est définie par l'une des instructions G `G93`/`G94`/`G95`. L'avance `F` n'agit que sur les axes à interpolation et s'applique jusqu'à ce qu'une nouvelle valeur d'avance soit programmée. Des caractères de séparation sont admis après l'adresse `F`.

Exemples :

`F100` ou `F 100`

`F.5`

`F=2*FEED`

Mode d'avance (G93/G94/G95)

Les instructions G `G93`, `G94` et `G95` sont modales. En cas de permutation entre `G93`, `G94` et `G95`, il conviendra de programmer une nouvelle valeur pour l'avance tangentielle. L'avance peut également être indiquée en degrés/min pour l'usinage avec des axes rotatifs.

Avance en sens inverse du temps (G93)

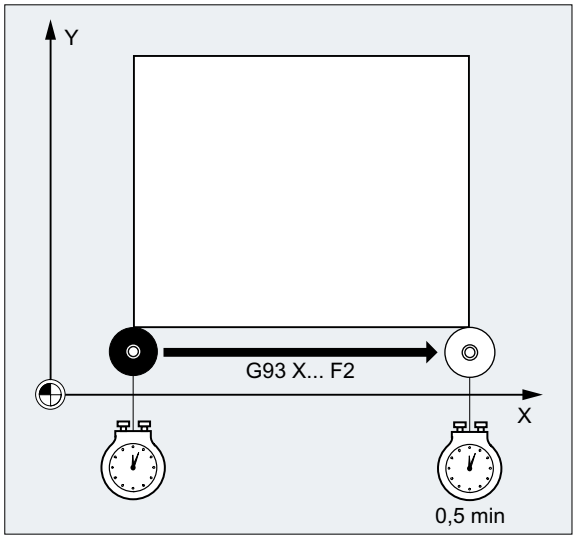
L'avance en sens inverse du temps indique la durée nécessaire à l'exécution d'un trajet programmé dans un bloc.

Unité : tr/min

Exemple :

```
N10 G93 G01 X100 F2
```

Signification : le trajet programmé est parcouru en 0,5 min.



Remarque

Si les trajets diffèrent fortement d'un bloc à l'autre, il est recommandé de définir une nouvelle valeur `F` dans chaque bloc pour `G93`. L'avance peut également être indiquée en degrés/min pour l'usinage avec des axes rotatifs.

Avance pour axes synchrones

L'avance programmée sous l'adresse `F` s'applique à tous les axes à interpolation programmés dans le bloc, mais pas aux axes synchrones. Les axes synchrones sont commandés de manière que leur déplacement prenne le même temps que les axes à interpolation, afin que tous les axes atteignent le point final en même temps.

Vitesse limite pour axes synchrones (FL)

L'instruction `FL` permet de programmer une vitesse limite pour les axes synchrones. Si aucune instruction `FL` n'est programmée, la vitesse en rapide s'applique. La désélection de `FL` s'effectue par affectation de PM (PM36200 $MA_AX_VELO_LIMIT).

Déplacement d'un axe à interpolation en tant qu'axe synchrone (FGROUP)

`FGROUP` permet de définir si un axe à interpolation doit être déplacé avec avance tangentielle ou comme axe synchrone. Dans le cas de l'interpolation hélicoïdale, il est possible de déterminer p. ex. que seuls deux axes géométriques X et Y doivent être déplacés avec l'avance programmée. L'axe de pénétration Z serait alors l'axe synchrone.

Exemple : `FGROUP(X,Y)`

Changer FGROUP

Une modification du paramétrage réalisé avec FGROUP est possible :

1. par une nouvelle programmation de FGROUP : p. ex. FGROUP(X,Y,Z)
2. par la programmation de FGROUP sans indication d'axe : FGROUP()
   Après FGROUP(), l'état de base réglé dans les paramètres machine s'applique. Les axes géométriques se déplacent à présent de nouveau dans le groupe des axes à interpolation.

Remarque

Les descripteurs d'axe pour FGROUP doivent être des noms d'axe de canal.

Unités de mesure pour l'avance F

Les instructions G G700 et G710 définissent non seulement les indications géométriques, mais également le système de mesure pour les avances F, c'est-à-dire :

- pour G700 : [inch/min]
- pour G710 : [mm/min]

Remarque

G70/G71 n'ont **aucun** effet sur les indications d'avance.

Unité de mesure pour axes synchrones avec vitesse limite FL

L'unité réglée pour F avec l'instruction G G700/G710 s'applique également à FL.

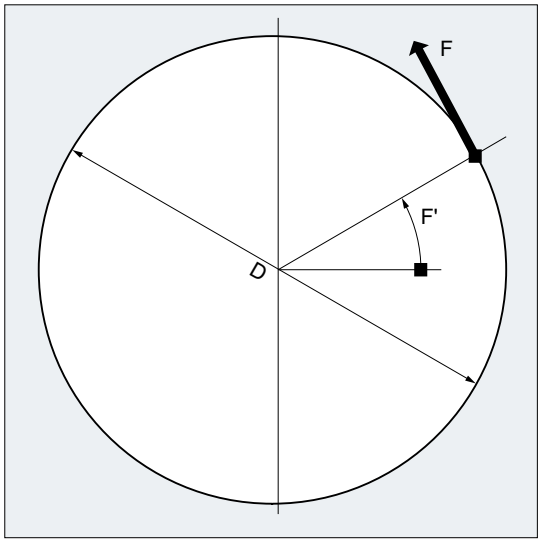
Unité des axes linéaires et des axes rotatifs

Pour les axes linéaires et les axes rotatifs combinés par FGROUP et parcourant ensemble une trajectoire, l'unité de l'avance est celle des axes linéaires. Selon le préréglage effectué avec G94/G95, en mm/min ou inch/min ou en mm/tour ou inch/tour.

La vitesse de l'axe rotatif en mm/min ou inch/min est calculée d'après la formule suivante :

F[mm/min] = F'[degrés/min] * π * D[mm] / 360[degrés]

où : F : Vitesse tangentielle
F' : Vitesse angulaire
π : Constante pi
D : Diamètre



Déplacement d'axes rotatifs avec une vitesse tangentielle F (FGREF)

Dans le cas des opérations d'usinage, pour lesquelles l'outil ou la pièce ou les deux sont déplacés par un axe rotatif, l'avance d'usinage doit pouvoir être programmée de la manière habituelle, c'est-à-dire sous la forme d'une avance tangentielle avec la valeur F. À cet effet, un rayon effectif (rayon de référence) doit être indiqué pour chaque axe rotatif concerné.

L'unité de ce rayon de référence dépend du réglage G70/G71/G700/G710.

Pour être pris en compte dans le calcul de l'avance tangentielle, tous les axes concernés doivent être indiqués dans l'instruction FGROUP.

Pour rester compatible avec le comportement obtenu sans programmation de FGREF, l'équivalence 1 degré = 1 mm est active après le lancement du système et après RESET. Cela correspond à un rayon de référence de FGREF = 360 mm / (2π) = 57,296 mm.

Remarque

Ce préréglage est indépendant du système de base actif (PM10240 $MN_SCALING_SYSTEM_IS_METRIC) et du réglage actuel G70/G71/G700/G710.

Particularités :

Code de programme

```
N100 FGROUP(X,Y,Z,A)
N110 G1 G91 A10 F100
N120 G1 G91 A10 X0.0001 F100,
```

Dans cette programmation, la valeur F programmée dans N110 est interprétée en tant qu'avance d'axe rotatif en degré/min, alors que l'avance dans N120 vaut soit 100 inch/min, soit 100 mm/min, en fonction du réglage G70/G71/G700/G710 actuel.

IMPORTANT
Différence d'avance L'équivalence FGREF est également active si seuls des axes rotatifs sont programmés dans le bloc. Dans ce cas, l'interprétation habituelle de la valeur F en degré/min n'est valable que si le rayon de référence correspond au préréglage de FGREF : • pour G71/G710 : FGREF[A]=57.296 • pour G70/G700 : FGREF[A]=57.296/25.4

Lecture du rayon de référence

La lecture de la valeur du rayon de référence pour un axe rotatif peut être réalisée au moyen de variables système :

- Dans les actions synchrones ou avec arrêt du prétraitement des blocs dans le programme pièce, au moyen de la variable système :

  $AA_FGREF[<Axe>] Valeur actuelle d'exécution

- Sans arrêt du prétraitement des blocs dans le programme pièce, au moyen de la variable système :

  $PA_FGREF[<Axe>] Valeur programmée

Si aucune valeur n'est programmée, la lecture dans les deux variables pour axes rotatifs indique le préréglage 360 mm / (2π) = 57,296 mm (ce qui correspond à 1 mm par degré).

Pour les axes linéaires, la lecture dans les deux variables indique toujours la valeur 1 mm.

Lecture d'axes à interpolation déterminant la vitesse

La lecture des axes concernés par l'interpolation de trajectoire peut être réalisée au moyen de variables système :

- Dans les actions synchrones ou avec arrêt du prétraitement des blocs dans le programme pièce, au moyen des variables système :

  $AA_FGROUP[<Axe>] Fournit la valeur "1", lorsque l'axe indiqué dans les paramètres de base ou par programmation de FGROUP exerce une influence sur la vitesse tangentielle dans le bloc principal actuel. Si ce n'est pas le cas, la variable fournit la valeur "0".

  $AC_FGROUP_MASK Fournit un code binaire des axes de canal programmés avec FGROUP et qui doivent contribuer à la vitesse tangentielle.

- Sans arrêt du prétraitement des blocs dans le programme pièce, au moyen des variables système :

$PA_FGROUP[<Axe>]	Fournit la valeur "1", lorsque l'axe indiqué dans les paramètres de base ou par programmation de `FGROUP` exerce une influence sur la vitesse tangentielle. Si ce n'est pas le cas, la variable fournit la valeur "0".
$P_FGROUP_MASK	Fournit un code binaire des axes de canal programmés avec `FGROUP` et qui doivent contribuer à la vitesse tangentielle.

Facteurs de référence de trajectoire pour les axes d'orientation avec FGREF

Sur les axes d'orientation, le mode d'action des facteurs `FGREF[]` varie selon que la modification de l'orientation de l'outil se fait par interpolation de l'axe rotatif ou par interpolation vectorielle.

Dans le cas d'une **interpolation de l'axe rotatif**, les facteurs `FGREF` respectifs des axes d'orientation sont pris en compte individuellement, comme dans le cas d'axes rotatifs, en tant que rayons de référence des courses axiales.

Dans le cas d'une **interpolation vectorielle**, un facteur effectif `FGREF` est pris en compte. Il est déterminé en tant que valeur géométrique moyenne à partir des différents facteurs `FGREF` :

FGREF[effectif] = racine n-ième de [(FGREF[A] * FGREF[B]...)]

où : A : descripteur du 1er axe d'orientation
B : descripteur du 2e axe d'orientation
C : descripteur du 3e axe d'orientation
n : nombre d'axes d'orientation

Exemple :

Pour une transformation standard à 5 axes, il y a deux axes d'orientation, de sorte que le facteur effectif se calcule sous forme de racine du produit des deux facteurs axiaux :

FGREF[effectif] = racine carrée de [(FGREF[A] * FGREF[B]...)]

Remarque

Avec le facteur effectif des axes d'orientation `FGREF`, il est ainsi possible de définir, sur l'outil, un point de référence auquel l'avance tangentielle programmée se rapporte.

2.7.2 Déplacement des axes de positionnement (POS, POSA, POSP, FA, WAITP, WAITMC)

Les axes de positionnement sont déplacés indépendamment des axes à interpolation avec une avance qui leur est propre. Les instructions d'interpolation sont sans effet. Les instructions POS/POSA/POSP déplacent les axes de positionnement et coordonnent simultanément les séquences de déplacement.

Des exemples typiques pour les axes de positionnement sont :

- Dispositifs de transfert de palettes
- Stations de mesure

WAITP permet de marquer la position dans le programme où la CN doit attendre jusqu'à ce qu'un axe programmé sous POSA dans un bloc CN antérieur atteigne son point final.

WAITMC permet de passer immédiatement au bloc CN suivant dès que la marque d'attente indiquée est atteinte.

Syntaxe

```
POS[<axe>]=<position>
POSA[<axe>]=<position>
POSP[<axe>]=(<position finale>,<longueur partielle>,<mode>)
FA[<axe>]=<valeur>
WAITP(<axe>) ; programmation dans un bloc CN spécifique !
WAITMC(<marque d'attente>)
```

Signification

POS / POSA :	Déplacement de l'axe de positionnement à la position indiquée POS et POSA ont la même fonctionnalité, mais diffèrent par leur comportement de changement de bloc : • Avec POS, le passage au bloc CN suivant a lieu seulement si la position à accoster est atteinte. • Avec POSA, le passage au bloc CN suivant a lieu même si la position à accoster n'est pas atteinte.		
	<axe> :	Nom de l'axe à déplacer (descripteur d'axe de canal ou d'axe géométrique)	
	<position> :	Position d'axe à accoster	
		Type :	REAL
POSP :	Déplacement de l'axe de positionnement en trajets partiels à la position finale indiquée		
	<position finale> :	Position de fin d'axe à accoster	
	<longueur partielle> :	Longueur d'un trajet partiel	
	<mode> :	Mode d'accostage	
		= 0 :	Pour les deux derniers trajets partiels, la distance restant à parcourir jusqu'à la position finale est divisée en deux trajets restants identiques (préréglage).
		= 1 :	La longueur partielle est adaptée de sorte que la somme de toutes les longueurs partielles calculées corresponde exactement au trajet jusqu'à la position finale.
	Remarque : POSP est spécialement mis en œuvre pour la programmation de mouvements d'oscillation (Page 984).		

FA :	Avance de l'axe de positionnement indiqué		
	<axe> :	Nom de l'axe à déplacer (descripteur d'axe de canal ou d'axe géométrique)	
	<valeur> :	Vitesse d'avance	
		Unité :	mm/min ou inch/min ou degré/min
	Remarque : Au maximum 5 valeurs FA peuvent être programmées dans un bloc CN.		
WAITP :	Attente de la fin de déplacement d'un axe de positionnement L'exécution des blocs suivants est en attente jusqu'à ce que l'axe de positionnement indiqué et programmé avec POSA dans un bloc CN précédent ait atteint sa position finale (avec arrêt précis fin).		
	<axe> :	Nom de l'axe (descripteur d'axe de canal ou d'axe géométrique) auquel l'instruction WAITP doit s'appliquer	
	Remarque : WAITP permet de libérer un axe comme axe d'oscillation ou comme axe de positionnement piloté (par l'AP) pour un déplacement.		
WAITMC :	Attente jusqu'à ce que la marque d'attente indiquée soit atteinte Lorsque la marque d'attente est atteinte, le passage au bloc CN suivant s'effectue immédiatement.		
	<marque d'attente> :	Numéro de la marque d'attente	

⚠ PRUDENCE

Déplacement avec POSA

En cas de lecture d'un bloc contenant une instruction qui déclenche implicitement un arrêt du prétraitement des blocs, ce bloc ne sera exécuté que lorsque tous les blocs prétraités auparavant et mémorisés auront été entièrement exécutés. Un arrêt précis (comme pour G9) a lieu dans le bloc précédent.

Exemples

Exemple 1 : déplacement avec POSA et accès aux paramètres d'état de la machine

Lors de l'accès à des données d'état de la machine ($A...), la commande déclenche un arrêt du prétraitement des blocs. Le traitement est arrêté, jusqu'à ce que tous les blocs prétraités et mémorisés auparavant aient été exécutés.

```
Code de programme                                  Commentaire
N40 POSA[X]=100
N50 IF $AA_IM[X]==R100 GOTOF MARKE1                ; Accès aux données d'état de la machine
N60 G0 Y100
N70 WAITP(X)
N80 MARKE1 :
N...
```

Exemple 2 : attente de la fin du déplacement avec WAITP

Dispositif de transfert de palettes

Axe U : Magasin de palettes
Transport de la palette porte-pièce dans la zone de travail

Axe V : Système de transfert sur une machine à mesurer où seront effectués des contrôles par sondage en cours d'usinage

Code de programme	Commentaire
N10 FA[U]=100 FA[V]=100	; Indications de l'avance spécifiques à chaque axe de positionnement U et V
N20 POSA[V]=90 POSA[U]=100 G0 X50 Y70	; Déplacement des axes de positionnement et des axes à interpolation
N50 WAITP(U)	; Le programme se poursuit seulement lorsque l'axe U a atteint la position programmée dans N20
…	

Informations complémentaires

Déplacement avec POSA

Ni le changement de bloc, ni l'exécution du programme ne sont influencés par `POSA`. Le déplacement vers le point final peut être exécuté parallèlement à l'exécution des blocs CN suivants.

Déplacement avec POS

Le changement de bloc se fait seulement lorsque tous les axes programmés sous `POS` ont atteint leur position finale.

Attendre la fin du déplacement avec WAITP

Après `WAITP`, l'axe est considéré comme n'étant plus sous le contrôle du programme CN et ce, jusqu'à ce qu'il soit reprogrammé. Cet axe peut alors être exploité par l'AP comme axe de positionnement ou par le programme CN/AP ou HMI comme axe d'oscillation.

Changement de bloc dans la rampe de freinage avec IPOBRKA et WAITMC

Le freinage d'un axe ne peut avoir lieu que si la marque d'attente n'a pas encore été atteinte ou si un autre critère de fin de bloc empêche le changement de bloc. Après `WAITMC`, l'axe se met immédiatement en mouvement si un autre critère de fin de bloc n'empêche le changement de bloc.

2.7.3 Broche en asservissement de position (SPCON, SPCOF)

Les instructions SPCON et SPCOF permettent d'activer et de désactiver explicitement le mode d'asservissement de position de la broche.

Remarque

L'activation du mode d'asservissement de position avec SPCON nécessite trois cycles du régulateur de position au maximum.

Syntaxe

```
SPCON
SPCON(<n>)
SPCON(<n>,<m>,...)
SPCOF
SPCOF(<n>)
SPCOF(<n>,<m>,...)
```

Signification

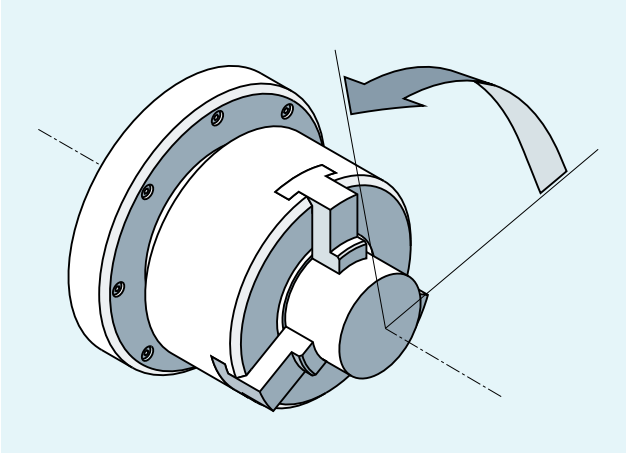
`SPCON` :	Activation du mode d'asservissement de position La broche indiquée est basculée de la régulation de vitesse de rotation à l'asservissement de position. `SPCON` est à effet modal et reste valide jusqu'à `SPCOF`.
`SPCOF` :	Désactivation du mode d'asservissement de position La broche indiquée est basculée de l'asservissement de position à la régulation de vitesse de rotation.
`<n>,<m>,...` :	Numéros des broches Sans indication de numéro de broche : Broche maître du canal

Remarque

Pour une broche synchrone avec couplage par valeur de consigne, la broche pilote ne doit pas être commutée en mode de régulation de vitesse avec SPCOF.

2.7.4 Positionner des broches (SPOS, SPOSA, M19, M70, WAITS)

SPOS, SPOSA ou M19 permettent de positionner des broches à des positions angulaires précises, p. ex. pour le changement d'outil.



SPOS, SPOSA et M19 provoquent une commutation temporaire en mode d'asservissement de position jusqu'au prochain M3/M4/M5/M41 ... M45.

Positionnement en mode axe

La broche peut également être déplacée en tant qu'axe à interpolation, axe synchrone ou axe de positionnement, sous son adresse définie dans un paramètre machine. En indiquant le descripteur de l'axe, la broche est en mode axe. M70 bascule directement la broche en mode axe.

Fin de positionnement

Le critère de fin de déplacement pour le positionnement de la broche est programmable avec FINEA, CORSEA, IPOENDA ou IPOBRKA.

Le changement de bloc s'effectue dès que les critères de fin de déplacement ainsi que le critère de changement de bloc pour l'interpolation de trajectoire sont remplis pour les broches et les axes à traiter dans le bloc.

Synchronisation

Pour synchroniser des déplacements de broche, WAITS permet d'attendre que la position de broche soit atteinte.

Conditions

La broche à positionner doit pouvoir fonctionner en mode d'asservissement de position.

Syntaxe

Positionnement de la broche :

`SPOS=<valeur>` / `SPOS[<n>]=<valeur>`

`SPOSA=<valeur>` / `SPOSA[<n>]=<valeur>`

`M19` / `M<n>=19`

Basculement de la broche en mode axe :

`M70` / `M<n>=70`

Définition du critère de fin de déplacement :

`FINEA` / `FINEA[S<n>]`

`COARSEA` / `COARSEA[S<n>]`

`IPOENDA` / `IPOENDA[S<n>]`

`IPOBRKA` / `IPOBRKA(<axe>[,<instant>])` ; programmation dans un bloc CN spécifique !

Synchronisation des déplacements de broche :

`WAITS` / `WAITS(<n>,<m>)` ; programmation dans un bloc CN spécifique !

Signification

<table>
<tr><td>SPOS / SPOSA :</td><td colspan="4">Positionnement de la broche à la position angulaire indiquée
SPOS et SPOSA ont la même fonctionnalité, mais diffèrent par leur mode de changement de bloc :
• Avec SPOS, le passage au bloc CN suivant a lieu seulement si la position est atteinte.
• Avec SPOSA, le passage au bloc CN suivant a lieu même si la position n'est pas atteinte.</td></tr>
<tr><td></td><td><n> :</td><td colspan="3">Numéro de la broche qui doit être positionnée.
Sans indication de numéro de broche ou avec le numéro de broche "0", SPOS ou SPOSA se rapportent à la broche maître.</td></tr>
<tr><td></td><td><valeur> :</td><td colspan="3">Position angulaire à laquelle la broche doit être positionnée.</td></tr>
<tr><td></td><td></td><td>Unité :</td><td colspan="2">degré</td></tr>
<tr><td></td><td></td><td>Type :</td><td colspan="2">REAL</td></tr>
<tr><td></td><td></td><td colspan="3">Les possibilités suivantes sont disponibles pour la programmation du mode d'accostage de position :</td></tr>
<tr><td></td><td></td><td>=AC(<valeur>) :</td><td colspan="2">Cotes absolues</td></tr>
<tr><td></td><td></td><td></td><td>Plage de valeurs :</td><td>0 ... 359,9999</td></tr>
<tr><td></td><td></td><td>=IC(<valeur>) :</td><td colspan="2">Cotes relatives</td></tr>
<tr><td></td><td></td><td></td><td>Plage de valeurs :</td><td>0 ... ±99 999,999</td></tr>
<tr><td></td><td></td><td>=DC(<valeur>) :</td><td colspan="2">Accostage direct de la valeur absolue</td></tr>
<tr><td></td><td></td><td>=ACN(<valeur>) :</td><td colspan="2">Introduction en cotes absolues, accostage en sens négatif</td></tr>
<tr><td></td><td></td><td>=ACP(<valeur>) :</td><td colspan="2">Introduction en cotes absolues, accostage en sens positif</td></tr>
<tr><td></td><td></td><td>=<valeur> :</td><td colspan="2">Identique à DC(<valeur>)</td></tr>
<tr><td>M<n>=19 :</td><td colspan="4">Positionnement de la broche maître (M19 ou M0=19) ou de la broche portant le numéro <n> (M<n>=19) à la position angulaire présélectionnée avec SD43240 $SA_M19_SPOS en utilisant le mode d'accostage de position présélectionné dans SD43250 $SA_M19_SPOSMODE
Le passage au bloc CN suivant s'effectue seulement lorsque la position est atteinte.</td></tr>
</table>

M<n>=70 :	Basculement de la broche maître (M70 ou M0=70) ou de la broche portant le numéro <n> (M<n>=70) en mode axe Aucune position définie n'est accostée. Le passage au bloc CN suivant s'effectue dès que le basculement est effectué.			
FINEA :	Fin du déplacement lorsque "Arrêt précis fin" est atteint			
COARSEA :	Fin du déplacement lorsque "Arrêt précis grossier" est atteint			
IPOENDA :	Fin de déplacement lorsque "Arrêt d'interpolateur" est atteint			
S<n> :	Broche pour laquelle le critère de fin de déplacement programmé doit être actif			
	<n> :	Numéro de broche		
	Sans indication de broche [S<n>] ou avec le numéro de broche "0", le critère de fin de déplacement programmé se rapporte à la broche maître.			
IPOBRKA :	Changement de bloc possible dans la rampe de freinage			
	<axe> :	Descripteur d'axe de canal		
	<instant> :	Instant du changement de bloc par rapport à la rampe de freinage		
		Unité :	Pourcent	
		Plage de valeurs :	100 (instant de mise en œuvre de la rampe de freinage) ... 0 (fin de la rampe de freinage)	
		Sans indication du paramètre <instant>, la valeur actuelle de la donnée de réglage est active : SD43600 $SA_IPOBRAKE_BLOCK_EXCHANGE **Remarque :** IBOBRKA avec l'instant "0" est identique à IPOENDA.		
WAITS :	Instruction de synchronisation de la ou des broches indiquées L'exécution des blocs suivants est en attente jusqu'à ce que la ou les broches indiquées et programmées avec SPOSA dans un bloc CN précédent aient atteint leur position finale (avec arrêt précis fin).			
	WAITS après M5 :	Attente jusqu'à l'arrêt de la ou des broches indiquées		
	WAITS après M3/M4 :	Attente jusqu'à ce que la ou les broches indiquées aient atteint leur vitesse de rotation de consigne		
	<n>,<m> :	Numéro de la broche à laquelle doit s'appliquer l'instruction de synchronisation Sans indication de numéro de broche ou avec le numéro de broche "0", WAITS se rapporte à la broche maître.		

Remarque

Dans un bloc CN, 3 indications de position de broche sont possibles.

Remarque

Dans le cas de cotes relatives IC(<valeur>), le positionnement de la broche est possible sur plusieurs tours.

Remarque

Si l'asservissement de position a été activé avec SPCON avant SPOS, celui-ci est conservé jusqu'à SPCOF.

Remarque

Grâce à la séquence de programmation, la commande détecte automatiquement le passage dans le mode axe. La programmation explicite de M70 dans le programme pièce ne s'avère donc plus nécessaire. La programmation de M70 peut toutefois être réalisée, p. ex. pour augmenter la lisibilité du programme pièce.

plus d'informations

Positionnement avec SPOSA

Ni le changement de bloc, ni l'exécution du programme ne sont influencés par SPOSA. Le positionnement de la broche peut être effectué parallèlement à l'exécution des blocs CN suivants. Le changement de bloc s'effectue quand toutes les fonctions programmées dans le bloc (à l'exception de la broche) ont atteint le critère de fin de bloc. Le positionnement de la broche peut s'effectuer sur plusieurs blocs (voir WAITS).

Remarque

Si la commande numérique lit, dans un des blocs suivants, une instruction qui génère implicitement un arrêt du prétraitement des blocs, le programme sera interrompu dans ce bloc jusqu'à ce que toutes les broches en cours de positionnement soient immobilisées.

Positionnement avec SPOS / M19

Le changement de bloc s'effectue seulement quand toutes les fonctions programmées dans le bloc ont atteint leur critère de fin de bloc (p. ex. toutes les fonctions auxiliaires ont été acquittées par l'AP, tous les axes ont atteint leur point final) et la broche a atteint la position programmée.

Vitesse des déplacements :

La vitesse et le comportement de temporisation pour le positionnement sont enregistrés dans les paramètres machine. La modification des valeurs configurées peut s'effectuer par programmation ou par actions synchrones, voir :

- Avance pour axes/broches et de positionnement (FA, FPR, FPRAON, FPRAOF) (Page 125)
- Correction programmable de l'accélération (ACC) (Page 130)

Indication des positions de la broche :

Puisque les instructions `G90`/`G91` sont inactives, les cotes correspondantes, p. ex. `AC`, `IC`, `DC`, `ACN`, `ACP` sont valables de manière explicite. Sans indications, le déplacement se fait automatiquement comme pour l'indication `DC`.

Synchronisation des déplacements de la broche avec WAITS

`WAITS` permet de marquer une position dans le programme où la CN attendra qu'une ou plusieurs broches programmées sous `SPOSA` dans un bloc CN antérieur aient atteint leur position.

Exemple :

Code de programme	Commentaire
`N10 SPOSA[2]=180 SPOSA[3]=0`	
`...`	
`N40 WAITS(2,3)`	`; On attend dans le bloc que les broches 2 et 3 aient atteint les positions indiquées dans le bloc N10`

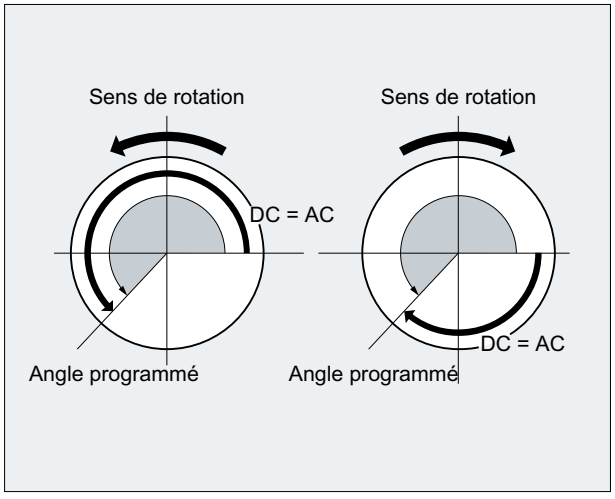
Après M5, WAITS permet d'attendre que la ou les broches soient à l'arrêt. Après M3/M4, WAITS permet d'attendre que la ou les broches aient atteint la vitesse ou le sens de rotation prédéfinis.

Remarque

Si la broche n'est pas encore synchronisée avec des marques de synchronisation, le sens de rotation positif est repris du paramètre machine (état à la livraison).

Positionner une broche déjà en rotation (M3/M4)

Lorsque M3 ou M4 sont activés, la broche s'immobilise à la valeur programmée.



Il n'y a pas de différence entre les indications DC et AC. Dans les deux cas, la rotation continue jusqu'à la position finale absolue dans le sens de rotation sélectionné par M3/M4. Pour ACN et ACP, un freinage peut avoir lieu pour respecter le sens d'accostage programmé. Pour l'indication IC, la rotation continue en fonction de la valeur indiquée, à partir de la position actuelle de la broche.

Positionner une broche à partir de l'arrêt (M5)

Le déplacement programmé est exécuté à partir de l'arrêt (M5), conformément aux indications.

2.7.5 Avance pour axes/broches et de positionnement (FA, FPR, FPRAON, FPRAOF)

Il existe en outre la possibilité de dériver l'avance par tour pour les axes à interpolation et axes synchrones ou pour les axes de positionnement et broches individuels à partir d'un autre axe rotatif ou d'une autre broche.

Le déplacement des axes de positionnement, comme p. ex. les systèmes de transport de pièces, les tourelles revolver ou les lunettes, s'effectue indépendamment des axes à interpolation et des axes synchrones. Par conséquent, on définit une avance spécifique à chaque axe de positionnement.

Une avance axiale spécifique peut également être programmée pour les broches.

Syntaxe

Avance pour l'axe de positionnement :
`FA[<axe>]=...`

Avance axiale pour la broche :
`FA[SPI(<n>)]=...`
`FA[S<n>]=...`

Dériver l'avance par tour pour les axes à interpolation ou axes synchrones :

`FPR(<axe rotatif>)`

`FPR(SPI(<n>))`

`FPR(S<n>)`

Dériver l'avance par tour pour les axes de positionnement ou broches :

`FPRAON(<axe>,<axe rotatif>)`

`FPRAON(<axe>,SPI(<n>))`

`FPRAON(<axe>,S<n>)`

`FPRAON(SPI(<n>),<axe rotatif>)`

`FPRAON(S<n>,<axe rotatif>)`

`FPRAON(SPI(<n>),SPI(<n>))`

`FPRAON(S<n>,S<n>)`

`FPRAOF(<axe>,SPI(<n>),...)`

`FPRAOF(<axe>,S<n>,...)`

Signification

`FA[...]=...` :	Avance pour l'axe ou la vitesse de positionnement (avance axiale) spécifiés pour la broche indiquée	
	Unité :	mm/min ou inch/min ou degré/min
	Plage de valeurs :	... 999 999,999 mm/min, degré/min ... 39 999,9999 inch/min
`FPR(...)` :	`FPR` marque l'axe rotatif (`<axe rotatif>`) ou la broche (`SPI(<n>)` / `S<n>`) à partir desquels l'avance par tour programmée sous `G95` doit être dérivée pour l'avance par tour des axes à interpolation et axes synchrones.	

`FPRAON(...)` :	Dériver l'avance par tour pour les axes de positionnement et les broches Le premier paramètre (`<axe>` / `SPI(<n>)` / `S<n>`) marque l'axe de positionnement ou la broche qui doivent se déplacer avec une avance par tour. Le deuxième paramètre (`<axe rotatif>` / `SPI(<n>)` / `S<n>`) marque l'axe rotatif ou la broche qui doivent être dérivés de l'avance par tour. **Remarque :** Le deuxième paramètre peut être omis, auquel cas l'avance sera dérivée de la broche maître.
`FPRAOF(...)` :	`FPRAOF` désélectionne l'avance par tour dérivée pour les axes ou broches indiqués.
`<axe>` :	Descripteur d'axe (axe de positionnement ou axe géométrique)
`SPI(<n>)` / `S<n>` :	Descripteur de broche `SPI(<n>)` et `S<n>` sont identiques du point de vue fonctionnel.
	`<n>` : Numéro de broche
	Remarque : `SPI` convertit le numéro de broche en descripteur d'axe. Le paramètre de transfert `(<n>)` doit contenir un numéro de broche valide.

Remarque

L'avance programmée FA[...] est modale.

Dans un bloc CN, vous pouvez programmer au maximum 5 avances pour les axes de positionnement / broches.

Remarque

L'avance déduite se calcule selon la formule suivante :

Avance dérivée = avance programmée * avance pilote en valeur absolue

Exemples

Exemple 1 : couplage de broches synchrones

Dans le cas d'un couplage de broches synchrones, la vitesse de positionnement de la broche asservie peut être programmée indépendamment de la broche pilote, p. ex. pour le positionnement.

```
Code de programme     Commentaire
...
FA[S2]=100            ; Vitesse de positionnement de la broche asservie (broche 2)
                      = 100 degrés/min
...
```

Exemple 2 : avance par tour dérivée pour les axes à interpolation

Les axes à interpolation X, Y doivent être déplacés avec une avance par tour qui est déduite de l'axe rotatif A.

Code de programme
...
N40 FPR(A)
N50 G95 X50 Y50 F500
...

Exemple 3 : dérivation de l'avance par tour pour la broche maître

Code de programme	Commentaire
N30 FPRAON(S1,S2)	; L'avance par tour pour la broche maître (S1) doit être dérivée de la broche 2
N40 SPOS=150	; Positionnement de la broche maître
N50 FPRAOF(S1)	; Désélection de l'avance par tour dérivée pour la broche maître

Exemple 4 : dérivation de l'avance par tour pour l'axe de positionnement

Code de programme	Commentaire
N30 FPRAON(X)	; L'avance par tour pour l'axe de positionnement X doit être dérivée de la broche maître
N40 POS[X]=50 FA[X]=500	; L'axe de positionnement se déplace à raison de 500 mm par tour de la broche maître
N50 FPRAOF(X)	

Informations complémentaires

FA[...]

Le mode d'avance est toujours G94. Si G70/G71 sont actifs, l'unité métrique/pouce dépend du préréglage effectué dans le paramètre machine. G700/G710 permettent de modifier l'unité dans le programme.

Remarque

Si FA n'est pas programmé, la valeur réglée dans le paramètre machine sera appliquée.

FPR(...)

En tant qu'extension de la fonction G95 (avance par tour rapportée à la broche maître), FPR permet également de dériver l'avance par tour d'une broche ou d'un axe rotatif quelconques. G95 FPR(...) s'applique aux axes à interpolation et aux axes synchrones.

Si l'axe rotatif / broche identifié(e) par FPR travaille avec asservissement de position, on a un couplage par la valeur de consigne, sinon c'est un couplage par la valeur réelle.

FPRAON(...)

FPRAON permet de dériver axialement l'avance par tour pour les axes de positionnement et les broches de l'avance momentanée d'un autre axe rotatif ou d'une autre broche.

FPRAOF(...)

FPRAOF permet de désactiver l'avance par tour pour un ou plusieurs axes ou broches à la fois.

2.7.6 Correction programmable de l'avance (OVR, OVRRAP, OVRA)

La vitesse des axes à interpolation et des axes de positionnement, ainsi que des broches peut être modifiée dans le programme CN.

Syntaxe

```
OVR=<valeur>
OVRRAP=<valeur>
OVRA[<axe>]=<valeur>
OVRA[SPI(<n>)]=<valeur>
OVRA[S<n>]=<valeur>
```

Signification

OVR :	Modification de l'avance tangentielle F	
OVRRAP :	Modification de l'avance pour la vitesse rapide	
OVRA :	Modification de l'avance de positionnement FA ou de la vitesse de rotation de broche S	
<axe> :	Descripteur d'axe (axe de positionnement ou axe géométrique)	
SPI(<n>) / S<n> :	Descripteur de broche SPI(<n>) et S<n> sont identiques du point de vue fonctionnel.	
	<n> :	Numéro de broche
	Remarque : SPI convertit le numéro de broche en descripteur d'axe. Le paramètre de transfert (<n>) doit contenir un numéro de broche valide.	
<valeur> :	Modification de l'avance en pourcentage La valeur se rapporte ou se superpose à la correction de l'avance réglée sur le tableau de commande machine.	
	Plage de valeurs :	0 ... 200 %, nombre entier
	Remarque : Les vitesses maximales réglées dans les paramètres machine ne sont pas dépassées lors de la correction de la trajectoire ou de la vitesse rapide.	

2.7.7 Correction programmable de l'accélération (ACC)

Dans les sections critiques d'un programme, il peut être nécessaire de limiter l'accélération en-deçà des valeurs maximales possibles, pour éviter des vibrations mécaniques par exemple.

La correction programmable de l'accélération permet de modifier l'accélération de chaque axe à interpolation ou broche au moyen d'une instruction dans le programme CN. La limitation agit dans tous les modes d'interpolation. Une accélération de 100 % signifie l'entrée en vigueur des valeurs définies dans les paramètres machine.

Syntaxe

```
ACC[<axe>]=<valeur>
ACC[SPI(<n>)]=<valeur>
ACC(S<n>)=<valeur>
```

Désactivation :
`ACC[...]=100`

Syntaxe

`ACC` :	Modification de l'accélération pour l'axe à interpolation indiqué ou modification de la vitesse de rotation pour la broche indiquée	
`<axe>` :	Nom d'axe de canal pour l'axe à interpolation	
`SPI(<n>)` / `S<n>` :	Descripteur de broche `SPI(<n>)` et `S<n>` sont identiques du point de vue fonctionnel.	
	`<n>` :	Numéro de broche
	Remarque : `SPI` convertit le numéro de broche en descripteur d'axe. Le paramètre de transfert `(<n>)` doit contenir un numéro de broche valide.	
`<valeur>` :	Modification de l'accélération en pourcentage La valeur se rapporte ou se superpose à la correction de l'avance réglée sur le tableau de commande machine.	
	Plage de valeurs :	1 ... 200 %, nombre entier

Remarque

En cas d'accélération plus importante, les valeurs admises par le constructeur de machine peuvent être dépassées.

Exemple

Code de programme	**Commentaire**
`N50 ACC[X]=80`	`; Le chariot de déplacement axial ne doit se déplacer qu'avec une accélération de 80 % en direction X`
`N60 ACC[SPI(1)]=50`	`; La broche 1 ne doit accélérer ou freiner qu'à 50 % de l'accélération possible`

Informations complémentaires

Correction de l'accélération programmée avec ACC

La correction de l'accélération programmée avec ACC[...] est toujours prise en compte comme dans la variable système $AA_ACC fournie à la sortie. La lecture dans le programme pièce et dans les actions synchrones intervient à des moments différents dans le traitement CN.

Dans le programme pièce

La valeur inscrite dans le programme pièce est seulement prise en compte dans la variable système $AA_ACC, telle qu'elle est écrite dans le programme pièce, si entre temps ACC n'a pas été modifié par une action synchrone.

Dans des actions synchrones

En conséquence : La valeur inscrite dans une action synchrone est seulement prise en compte dans la variable système $AA_ACC, telle qu'elle est écrite dans l'action synchrone, si entre temps ACC n'a pas été modifié par un programme pièce.

L'accélération prédéfinie peut également être modifiée par des actions synchrones.

Exemple :

Code de programme

```
...
N100 EVERY $A_IN[1] DO POS[X]=50 FA[X]=2000 ACC[X]=140
```

La valeur actuelle de l'accélération peut être interrogée avec la variable système $AA_ACC[<axe>]. Un paramètre machine permet de sélectionner la valeur qui doit s'appliquer après un RESET ou une fin de programme pièce, à savoir la dernière valeur ACC définie ou la valeur 100 %.

2.7.8 Avance avec correction par manivelle (FD, FDA)

Les instructions FD et FDA permettent de déplacer des axes à l'aide de manivelles pendant l'exécution du programme pièce. Les déplacements programmés des axes sont superposés aux impulsions de la manivelle, interprétées comme spécifications de trajet ou de vitesse.

Axes à interpolation
L'avance tangentielle programmée peut être superposée pour les axes à interpolation. C'est la manivelle du 1er axe géométrique du canal qui est exploitée. Les impulsions de la manivelle, interprétées pour chaque période d'appel de l'interpolateur en fonction du sens de rotation,

correspondent à la vitesse tangentielle à superposer. Les valeurs limites pouvant être atteintes par la vitesse tangentielle grâce à la correction par manivelle sont :

- Minimum : 0
- Maximum : valeurs limites des paramètres machine des axes à interpolation concernés par le déplacement

Remarque

Avance tangentielle

L'avance tangentielle F et l'avance par manivelle FD ne doivent pas être programmées simultanément dans un bloc CN.

Axe de positionnement
La superposition axiale du trajet ou de la vitesse est possible pour les axes de positionnement. L'exploitation portera sur la manivelle affectée à l'axe.

- Superposition de trajet
  Les impulsions de la manivelle, interprétées en fonction du sens de rotation, correspondent au trajet de l'axe à déplacer. Seules les impulsions de la manivelle dans le sens de la position programmée sont prises en compte.
- Superposition de vitesse
  Les impulsions de la manivelle, interprétées pour chaque période d'appel de l'interpolateur en fonction du sens de rotation, correspondent à la vitesse axiale à superposer. Les valeurs limites pouvant être atteintes par la vitesse tangentielle grâce à la correction par manivelle sont :
  - Minimum : 0
  - Maximum : valeurs limites des paramètres machine de l'axe de positionnement

Syntaxe

```
FD=<vitesse>
FDA[<axe>]=<vitesse>
```

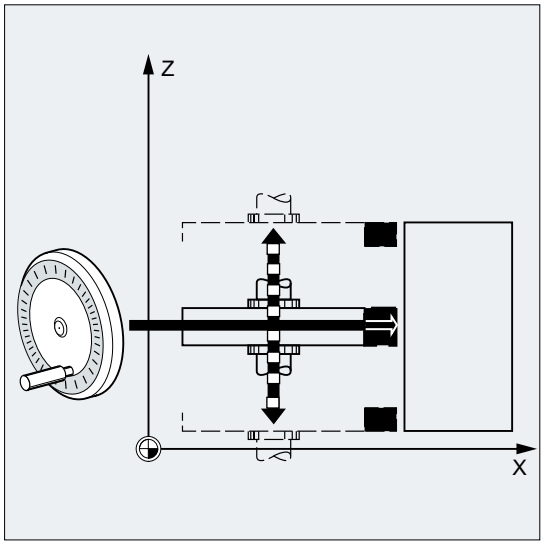
Signification

`FD=< vitesse >` :	Avance tangentielle et validation de la superposition de vitesse par manivelle. <vitesse> : • Valeur = 0 : non autorisée ! • Valeur ≠ 0 : vitesse tangentielle
`FDA[<axe>]=<vitesse>` :	Avance axiale <vitesse> : • Valeur = 0 : spécification de trajet par manivelle • Valeur ≠ 0 : vitesse axiale
`<axe>` :	descripteur de l'axe de positionnement

Remarque

FD et FDA sont non modaux.

Exemple



forçage de déplacement : La meule qui oscille dans la direction Z est amenée vers la pièce avec la manivelle, en direction X.

L'opérateur peut commander manuellement l'approche jusqu'à obtenir une projection d'étincelles homogène. L'activation de "Effacement de la distance restant à parcourir" bascule dans le bloc CN suivant et poursuit le traitement en mode AUTOMATIQUE.

Informations complémentaires

Déplacement d'axes à interpolation avec superposition de vitesse (FD=<vitesse>)
Les conditions suivantes doivent être remplies pour le bloc de programme pièce dans lequel la correction de vitesse tangentielle est programmée :

- instruction de déplacement G1, G2 ou G3 active
- arrêt précis G60 actif
- avance linéaire G94 active

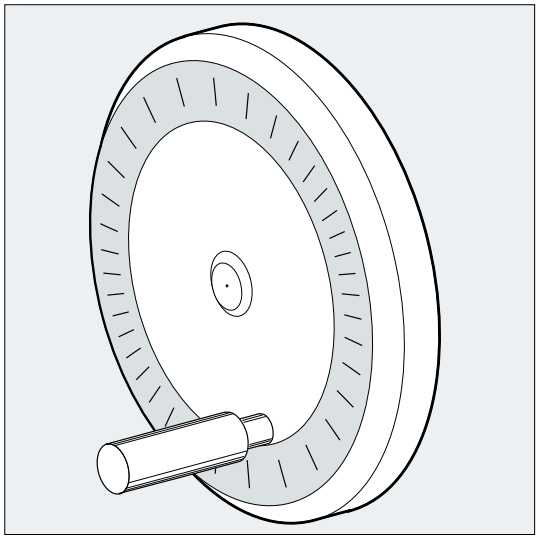
Correction de l'avance
La correction de l'avance a seulement un effet sur la vitesse tangentielle programmée, pas sur la composante de vitesse créée par la manivelle (exception : correction de l'avance par commutateur = 0).

Exemple :

Code de programme	Description
N10 X… Y… F500	; Avance tangentielle = 500 mm/min

Code de programme	Description
N20 X… Y… FD=700	; Avance tangentielle = 700 mm/min et correction de la vitesse par manivelle. ; Dans N20, l'accélération passe de 500 à 700 mm/min. Avec la manivelle, la vitesse tangentielle peut être modifiée en fonction du sens de rotation entre 0 et la valeur maximale (paramètres machine).

Déplacement d'axes de positionnement avec spécification du trajet (FDA[<axe>]=0)
Dans le bloc CN, dans lequel `FDA[<axe>]=0` est programmé, l'avance est réglée sur zéro de telle sorte que le programme ne provoque pas de déplacement. Le déplacement programmé vers la position de destination est alors exclusivement commandé par l'utilisateur au moyen de la manivelle.



Exemple :

Code de programme	Description
...	
N20 POS[V]=90 FDA[V]=0	; Position de destination = 90 mm, avance axiale = 0 mm/min et Superposition de trajet au moyen de la manivelle. ; Vitesse de l'axe V en début de bloc = 0 mm/min ; La spécification de la distance à parcourir et de la vitesse s'effectue par impulsions de la manivelle

Sens de déplacement, vitesse de déplacement :
Les axes parcourent la distance spécifiée au moyen des impulsions de la manivelle dans le sens horaire. En fonction du sens de rotation, le déplacement peut s'effectuer dans le sens horaire ou antihoraire. La vitesse de déplacement est d'autant plus élevée que la rotation de la manivelle est rapide.

Plage de déplacement :
La plage de déplacement est limitée par la position de départ et la position finale programmée.

Déplacement d'axes de positionnement avec superposition de vitesse (FDA[<axe>]=<vitesse>)

Dans le bloc CN, dans lequel FDA[...]=... est programmé, l'avance est accélérée ou retardée à partir de la dernière valeur FA programmée jusqu'à la valeur programmée sous FDA. A partir de l'avance FDA actuelle, le déplacement programmé peut être accéléré jusqu'à la position de destination ou retardé jusqu'à zéro par rotation de la manivelle. La vitesse maximale correspond aux valeurs définies dans les paramètres machine.

Exemple :

Code de programme	Description
N10 POS[V]=… FA[V]=100	; Avance axiale = 100 mm/min
N20 POS[V]=100 FAD[V]=200	; Position de destination axiale = 100, avance axiale = 200 mm/min
	; et superposition de vitesse par manivelle.
	; Dans N20, l'accélération passe de 100 à 200 mm/min. Grâce
	; La manivelle permet de modifier la vitesse entre 0 et la
	; valeur maximale (paramètres machine) en fonction du sens de rotation
	.

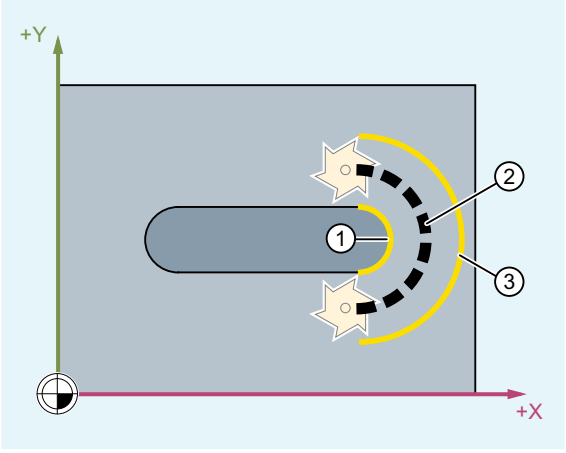
Plage de déplacement :
La plage de déplacement est limitée par la position de départ et la position finale programmée.

2.7.9 Optimisation de l'avance sur contours courbes (CFTCP, CFC, CFIN)

Lorsque la correction G41/G42 du rayon de fraise est activée, l'avance programmée s'applique d'abord à la trajectoire du centre de la fraise (voir chap. "Transformations des coordonnées (Frames) (Page 311)").

Dans le cas du fraisage d'un cercle (ainsi que pour l'interpolation d'un polynôme et d'une courbe de type spline), l'avance au niveau du bord de la fraise peut dans certains cas être modifiée de manière si importante que le résultat du traitement en pâtit.

Exemple : fraisage d'un rayon extérieur faible avec un outil de taille importante. Le trajet que doit parcourir l'extérieur de la fraise est beaucoup plus important que le trajet sur le contour.



On utilise de ce fait une avance très faible au niveau du contour. Afin d'éviter de tels effets, il est recommandé de régler l'avance en conséquence pour les contours courbés.

Syntaxe

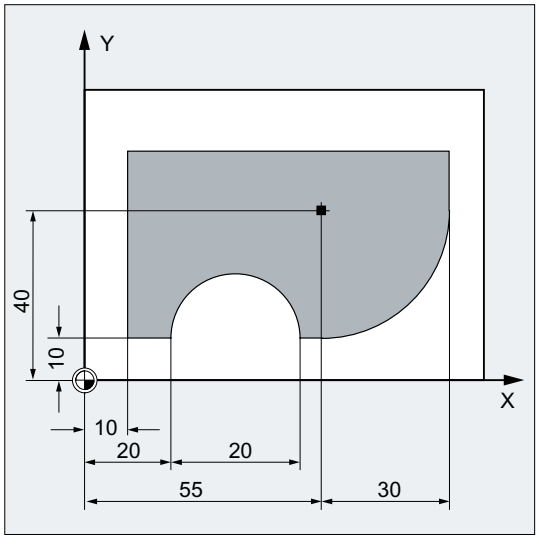
```
CFTCP
CFC
CFIN
```

Signification

CFTCP :	Avance constante au niveau de la trajectoire centrale de la fraise La commande maintient la vitesse d'avance constante, les corrections de l'avance sont désactivées.
CFC :	Avance constante au niveau du contour (arête tranchante de l'outil) Cette fonction est préréglée.
CFIN :	Avance constante au niveau de l'arête tranchante de l'outil uniquement sur les contours concaves, sinon sur la trajectoire centrale de la fraise La vitesse d'avance est réduite sur les courbures concaves.

Exemple

Dans cet exemple, on exécute d'abord le contour avec une avance à correction CFC. Pour la finition, le fond est traité en outre avec CFIN. Ceci permet d'éviter que le fond ne soit endommagé par une vitesse d'avance excessive au niveau des courbures convexes.

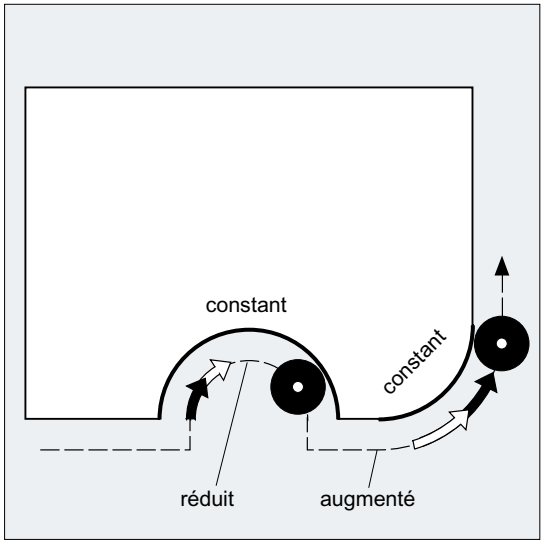


Code de programme	Commentaire
N10 G17 G54 G64 T1 M6	
N20 S3000 M3 CFC F500 G41	
N30 G0 X-10	
N40 Y0 Z-10	; Pénétration à première profondeur de passe

Code de programme	Commentaire
N50 CONTOUR1	; Appel du sous-programme
N40 CFIN Z-25	; Pénétration à seconde profondeur de passe
N50 CONTOUR1	; Appel du sous-programme
N60 Y120	
N70 X200 M30	

Informations complémentaires

Avance constante sur le contour avec CFC



La vitesse d'avance est réduite sur des courbures concaves et augmentée sur des courbures convexes. De ce fait, la vitesse reste constante au niveau de l'arête tranchante de l'outil et donc, au niveau du contour.

2.7.10 Plusieurs des valeurs d'avance dans un bloc (F, ST, SR, FMA, STA, SRA)

La fonction "Plusieurs valeurs d'avance dans un bloc" permet d'activer, de manière synchrone au déplacement, différentes valeurs d'avance d'un bloc CN, d'un arrêt temporisé ainsi que du retrait en fonction d'entrées externes TOR et/ou analogiques.

Syntaxe

Déplacement avec interpolation :
```
F=... F7=... F6=... F5=... F4=... F3=... F2=... ST=... SR=...
```

Déplacement axial :
```
FA[<Axe>]=... FMA[7,<Axe>]=... FMA[6,<Axe>]=... FMA[5,<Axe>]=...
FMA[4,<Axe>]=... FMA[3,<Axe>]=... FMA[2,<Axe>]=... STA[<Axe>]=...
SRA[<Axe>]=...
```

Signification

F=... :	L'avance tangentielle, valable tant qu'aucun signal d'entrée ne survient, est programmée sous l'adresse F.	
	Prise d'effet :	modale
F2=... à F7=... :	Outre l'avance tangentielle, il est possible de programmer jusqu'à 6 avances supplémentaires dans le bloc. L'extension numérique indique le numéro de bit de l'entrée dont la modification active l'avance.	
	Prise d'effet :	bloc par bloc (non modale)
ST=... :	Arrêt temporisé en s (pour les boucles technologiques : durée des passes à lécher)	
	Bit d'entrée :	1
	Prise d'effet :	bloc par bloc (non modale)
SR=... :	Distance de retrait L'unité pour la distance de retrait est l'unité momentanément valide (mm ou inch).	
	Bit d'entrée :	0
	Prise d'effet :	bloc par bloc (non modale)
FA[<Axe>]=... :	L'avance axiale, valable tant qu'aucun signal d'entrée ne survient, est programmée sous l'adresse FA.	
	Prise d'effet :	modale
FMA[2,<Axe>]=... à FMA[7,<Axe>]=... :	Outre l'avance axiale FA, FMA permet de programmer jusqu'à 6 avances supplémentaires par axe dans le bloc. Le premier paramètre indique le numéro de bit de l'entrée, le deuxième l'axe auquel l'avance doit s'appliquer.	
	Prise d'effet :	bloc par bloc (non modale)
STA[<Axe>]=... :	arrêt temporisé axial en s (pour les boucles technologiques : durée des passes à lécher)	
	Bit d'entrée :	1
	Prise d'effet :	bloc par bloc (non modale)
SRA[<Axe>]=... :	distance de retrait axiale	
	Bit d'entrée :	0
	Prise d'effet :	bloc par bloc (non modale)
<Axe> :	axe auquel l'avance doit s'appliquer	

Remarque

Priorité des signaux

L'interrogation des signaux se fait dans l'ordre croissant à partir du bit d'entrée 0 (E0). Ainsi, le mouvement de retrait possède la plus grande priorité et l'avance F7 de la plus faible. L'arrêt temporisé et le mouvement de retrait terminent les mouvement d'avance qui ont été activées par F2 à F7.

Le signal ayant la priorité la plus élevée détermine l'avance actuelle.

Remarque

Effacement de la distance restant à parcourir

Lorsque l'entrée bit 1 pour la temporisation ou bit 0 pour la distance de retrait est activée, la distance restant à parcourir par les axes à interpolation ou les axes individuels concernés est effacée et la temporisation ou le retrait est déclenché.

Remarque

Distance de retrait

L'unité pour la distance de retrait est l'unité momentanément valide (mm ou inch).

La course de retrait est toujours inverse à la direction du mouvement actuel. La valeur absolue du retrait est toujours programmée avec `SR`/`SRA`. Aucun signe n'est programmé.

Remarque

POS au lieu de POSA

Si des avances, une temporisation ou une distance de retrait activées par des actions externes sont programmés pour un axe, cet axe ne doit pas être programmé en tant qu'axe POSA (axe de positionnement au-delà de limites de bloc) dans ce bloc.

Remarque

Interrogation de l'état

L'état d'une entrée peut également être interrogé pour les instructions synchrones de différents axes.

Remarque

Anticipation (LookAhead)

L'anticipation (LookAhead) est également active si plusieurs avances sont programmées dans un bloc. L'avance actuelle peut ainsi être limitée par l'anticipation (LookAhead).

Exemples

Exemple 1 : déplacement avec interpolation

Code de programme	Commentaire
G1 X48 F1000 F7=200 F6=50 F5=25 F4=5 ST=1.5 SR=0.5	; Avance tangentielle = 1000 ; Valeurs supplémentaires d'avance tangentielle : 200 (bit d'entrée 7) ; 50 (bit d'entrée 6) ; 25 (bit d'entrée 5) ; 5 (bit d'entrée 4) ; Arrêt temporisé 1,5 s ; Retrait 0,5 mm

Exemple 2 : déplacement axial

Code de programme	Commentaire
POS[A]=300 FA[A]=800 FMA[7,A]=720 FMA[6,A]=640 FMA[5,A]=560 STA[A]=1.5 SRA[A]=0.5	; Avance pour l'axe A = 800 ; Valeurs d'avance supplémentaires pour l'axe A : 720 (bit d'entrée 7) ; 640 (bit d'entrée 6) ; 560 (bit d'entrée 5) ; Arrêt temporisé axial : 1,5 s ; Retrait axial : 0,5 mm

Exemple 3 : plusieurs cycles d'opération dans un même bloc

Code de programme	Commentaire
N20 T1 D1 F500 G0 X100	Position de départ
N25 G1 X105 F=20 F7=5 F3=2.5 F2=0.5 ST=1.5 SR=0.5	; Avance normale avec F, ; ébauchage avec F7, ; finition avec F3, ; superfinition avec F2, ; arrêt temporisé 1,5 s, ; distance de retrait 0,5 mm
...	

2.7.11 Avance à effet non modal (FB)

La fonction "Avance non modale" permet de présélectionner une avance distincte pour un bloc individuel. Après ce bloc, l'avance modale précédente est de nouveau active.

Syntaxe

```
FB=<valeur>
```

Signification

FB :	Avance pour le bloc actuel uniquement
<valeur> :	La valeur programmée doit être supérieure à zéro. L'interprétation s'effectue en fonction du mode d'avance actif : • G94 : avance en mm/min ou °/min • G95 : avance en mm/tr ou inch/tr • G96 : vitesse de coupe constante

Remarque

Si aucun déplacement n'est programmé dans le bloc (p. ex. bloc de calcul), FB est sans effet.

Si aucune avance explicite n'est programmée pour le chanfrein/arrondi, la valeur de FB s'applique également à un élément de contour chanfrein/arrondi de ce bloc.

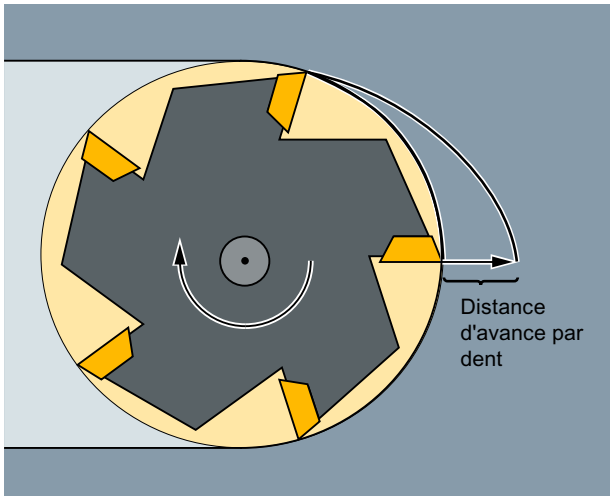
Aucune restriction ne s'applique aux interpolations d'avance FLIN, FCUB, ...

La programmation simultanée de FB et FD (déplacement par manivelle avec superposition d'avance) ou F (avance tangentielle modale) n'est **pas** possible.

Exemple

Code de programme	Commentaire
N10 G0 X0 Y0 G17 F100 G94	; Position de départ
N20 G1 X10	; Avance 100 mm/min
N30 X20 FB=80	; Avance 80 mm/min
N40 X30	; L'avance est de nouveau égale à 100 mm/min
...	

2.7.12 Avance par dent (G95 FZ)



Avec le paramètre d'outil $TC_DPNT (nombre de dents) du bloc de données de correction d'outil actif, la commande calcule l'avance par tour active pour chaque bloc de déplacement à partir de l'avance par dent programmée :

F = FZ * $TC_DPNT

où : F : Avance par tour en mm/tr ou inch/tr
FZ : Avance par dent mm/dent ou inch/dent
$TC_DPNT : Paramètres d'outil de la variable système : nombre de dents/tr

Il n'est pas tenu compte du type ($TC_DP1) de l'outil actif.

L'avance par dent programmée est indépendante d'un changement d'outil et de la sélection/désélection d'un bloc de données de correction de l'outil et est conservée de manière modale.

Une modification du paramètre d'outil $TC_DPNT pour le tranchant actif prend effet à la prochaine sélection de la correction de l'outil ou à la prochaine actualisation des paramètres de correction actifs.

Un changement d'outil et une sélection/désélection d'un bloc de données de correction d'outil entraînent un recalcul de l'avance par tour active.

Remarque

L'avance par dent concerne uniquement la trajectoire, une programmation spécifique à l'axe n'est pas possible.

Syntaxe

```
G95 FZ...
```

Signification

G95 :	Mode d'avance : avance par tour en mm/tr ou inch/tr (dépend de G700/G710) Pour G95, voir "Avance (G93, G94, G95, F, FGROUP, FL, FGREF) (Page 108)"	
FZ :	Vitesse d'avance par dent	
	Activation :	avec G95
	Prise d'effet :	modale
	Unité :	mm/dent ou inch/dent (dépend de G700/G710)

IMPORTANT
Changement d'outil / changement de la broche maître
L'utilisateur doit prendre en compte un changement d'outil ou de broche maître ultérieur en effectuant une programmation adéquate, p. ex. une nouvelle programmation de FZ.

IMPORTANT
Actions de l'outil non définies Tout comme la géométrie de trajectoire (droite, cercle...), les objets technologiques tels que le fraisage en avalant ou en opposition, le fraisage en bout ou circonférentiel, etc. ne sont pas non plus pris en compte automatiquement. Ces facteurs doivent de ce fait être pris en compte lors de la programmation de l'avance par dent.

Remarque

Basculement entre G95 F... et G95 FZ...

Un basculement entre `G95 F...` (avance par tour) et `G95 FZ...` (avance par dent) supprime respectivement la valeur d'avance non active.

Remarque

Dérivation de l'avance avec FPR

De manière analogique à l'avance par tour, `FPR` permet également de dériver l'avance par dent d'un axe rotatif ou d'une broche quelconques (voir "Avance pour axes/broches et de positionnement (FA, FPR, FPRAON, FPRAOF) (Page 125)").

Exemples

Exemple 1 : Fraise à 5 dents ($TC_DPNT = 5)

Code de programme	Commentaire
N10 G0 X100 Y50	
N20 G1 G95 FZ=0.02	; Avance par dent 0,02 mm/dent
N30 T3 D1	; Mise en place de l'outil et activation du bloc de données de correction d'outil
M40 M3 S200	; Vitesse de rotation de la broche 200 tr/min
N50 X20	; Fraiser avec :
	FZ = 0,02 mm/dent
	avance par tour active :
	F = 0,02 mm/dent * 5 dents/tr = 0,1 mm/tr
	ou
	F = 0,1 mm/tr * 200 tr/min = 20 mm/min
...	

Exemple 2 : basculement entre G95 F... et G95 FZ...

Code de programme	Commentaire
N10 G0 X100 Y50	
N20 G1 G95 F0.1	; Avance par tour 0,1 mm/tr
N30 T1 M6	
N35 M3 S100 D1	
N40 X20	

Code de programme	Commentaire
N50 G0 X100 M5	
N60 M6 T3 D1	; Mise en place de l'outil à 5 dents, p. ex. ($TC_DPNT = 5)
N70 X22 M3 S300	
N80 G1 X3 G95 FZ=0.02	; Changement de G95 F… à G95 FZ…, avance par dent de 0,02 mm/dent active
...	

Exemple 3 : dérivation de l'avance par dent à partir d'une broche (FBR)

Code de programme	Commentaire
…	
N41 FPR(S4)	; Outil sur la broche 4 (pas la broche maître)
N51 G95 X51 FZ=0.5	; Avance par dent de 0,5 mm/dent en fonction de la broche S4
...	

Exemple 4 : changement d'outil ultérieur

Code de programme	Commentaire
N10 G0 X50 Y5	
N20 G1 G95 FZ=0.03	; Avance par dent 0,03 mm/dent
N30 M6 T11 D1	; Mise en place l'outil à 7 dents, p. ex. ($TC_DPNT = 7)
N30 M3 S100	
N40 X30	; Avance par tour active 0,21 mm/tr
N50 G0 X100 M5	
N60 M6 T33 D1	; Mise en place de l'outil à 5 dents, p. ex. ($TC_DPNT = 5)
N70 X22 M3 S300	
N80 G1 X3	; Avance par dent modale de 0,03 mm/dent, avance par tour active 0,15 mm/tr
...	

Exemple 5 : changement de la broche maître

Code de programme	Commentaire
N10 SETMS(1)	; La broche 1 est la broche maître
N20 T3 D3 M6	; L'outil 3 est mis en place sur la broche 1
N30 S400 M3	; Vitesse de rotation S400 de la broche 1 (et ainsi de T3)
N40 G95 G1 FZ0.03	; Avance par dent 0,03 mm/dent
N50 X50	; Déplacement avec interpolation, l'avance active dépend de : - l'avance par dent FZ - la vitesse de rotation de la broche 1 - nombre de dents de l'outil actif T3
N60 G0 X60	
...	
N100 SETMS(2)	; La broche 2 devient la broche maître

Code de programme	Commentaire
N110 T1 D1 M6	; L'outil 1 est mis en place sur la broche 2
N120 S500 M3	; Vitesse de rotation S500 de la broche 2 (et ainsi de T1)
N130 G95 G1 FZ0.03 X20	; Déplacement avec interpolation, l'avance active dépend de :
	- l'avance par dent FZ
	- la vitesse de rotation de la broche 2
	- nombre de dents de l'outil actif T1

Remarque

Après le changement de la broche maître (N100), un outil (N110) entraîné par la broche 2 doit être mis en place.

Informations complémentaires

Changement entre G93, G94 et G95

FZ peut également être programmé lorsque G95 n'est pas actif, mais il n'a cependant aucun effet et la sélection de G95 le supprime, c'est-à-dire qu'un changement entre G93, G94 et G95 supprime également la valeur de FZ, de manière analogique à F.

Nouvelle sélection de G95

Une nouvelle sélection de G95 lorsque G95 est déjà actif n'a aucun effet (si aucun changement entre F et FZ n'est programmé).

Avance non modale (FB)

Une avance non modale FB... est interprétée comme avance par dent lorsque G95 FZ... est actif (modal).

Mécanisme SAVE

Dans les sous-programmes avec l'attribut SAVE, FZ est inscrit dans la valeur avant le démarrage du sous-programme, de manière analogique à F.

Plusieurs avances dans un bloc

La fonction "Plusieurs valeurs d'avance dans un bloc" n'est pas possible pour l'avance par dent.

Actions synchrones

La spécification de FZ n'est pas possible à partir d'actions synchrones.

Lecture de la vitesse d'avance par dent et du mode d'avance tangentielle

La lecture de la vitesse d'avance par dent et du mode d'avance tangentielle peut être réalisée au moyen de variables système :

- Avec arrêt d'avance dans le programme pièce, au moyen des variables système :

	$AC_FZ	Vitesse d'avance par dent qui était active lors du traitement du bloc principal actuel.	
	$AC_F_TYPE	Mode d'avance tangentielle qui était actif lors du traitement du bloc principal actuel.	
		Valeur :	**Signification :**
		0	mm/min
		1	mm/tr
		2	inch/min
		3	inch/tr
		11	mm/dent
		33	inch/dent

- Sans arrêt d'avance dans le programme pièce, au moyen des variables système :

	$P_FZ	Vitesse d'avance par dent programmée	
	$P_F_TYPE	Mode d'avance tangentielle programmée	
		Valeur :	**Signification :**
		0	mm/min
		1	mm/tr
		2	inch/min
		3	inch/tr
		11	mm/dent
		33	inch/dent

Remarque

Si `G95` n'est pas actif, les variables $P_FZ et $AC_FZ fournissent toujours la valeur zéro.

2.8 Paramètres géométriques

2.8.1 Décalage d'origine réglable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153)

Les instructions G54 à G57 et G505 à G599 permettent d'activer les décalages d'origine réglables via l'interface utilisateur pour le décalage du système de coordonnées pièce par rapport au système de coordonnées de base.

Syntaxe

Mise en marche de l'entraînement :

```
G54
...
G57
G505
...
G599
```

Désactivation ou inhibition :

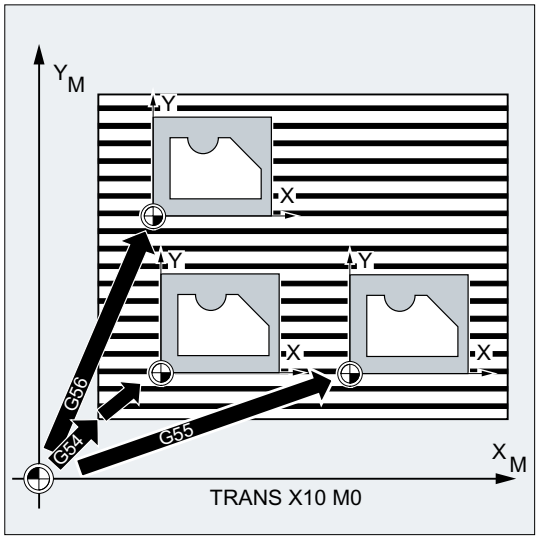
G500 / G53 / G153 / SUPA

Signification

G54 ... G57 :	Appel du 1er au 4e décalage d'origine réglable (DO)	
G505 ... G599 :	Appel du 5e au 99e DO réglable	
G500 :	Désactivation du DO réglable courant	
	G500=Frame neutre : (réglage par défaut ne contenant ni décalage, ni rotation, ni fonction miroir, ni changement d'échelle)	Désactivation du DO réglable jusqu'à l'appel suivant, activation du frame de base résultant ($P_ACTBFRAME).
	G500 différent de 0 :	Activation du premier décalage d'origine réglable ($P_UIFR[0]) et activation du frame de base résultant ($P_ACTBFRAME), ou activation d'un éventuel frame de base modifié.
G53 :	G53 inhibe en mode non modal les DO réglables et les DO programmables.	
G153 :	G153 agit comme G53 et désactive également le frame de base résultant.	
SUPA :	SUPA agit comme G153 et désactive en plus les déplacements suivants : • Décalages par manivelle (DRF) • Déplacements forcés • DO externe • Décalage PRESET	

Exemple

Il s'agit d'usiner successivement trois pièces qui sont disposées sur une palette suivant les valeurs de décalage d'origine définies avec G54 à G56. L'usinage successif est programmé dans le sous-programme L47.



Code de programme	Commentaire
N10 G0 G90 X10 Y10 F500 T1	; Accostage
N20 G54 S1000 M3	; Appel du premier DO, broche sens horaire
N30 L47	; Exécution du programme, ici sous-programme
N40 G55 G0 Z200	; Appel du deuxième DO, Z pour franchir obstacle
N50 L47	; Exécution du programme, ici sous-programme
N60 G56	; Appel du troisième DO
N70 L47	; Exécution du programme, ici sous-programme
N80 G53 X200 Y300 M30	; Suppression du décalage d'origine, fin du programme

2.8.2 Décalage d'origine réglable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153) : Informations complémentaires

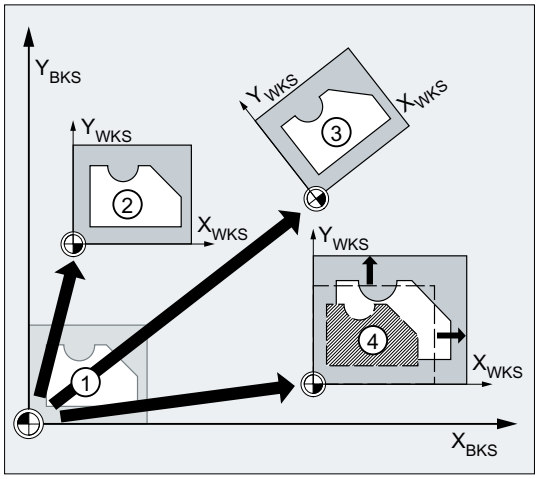
Informations complémentaires

Décalage d'origine réglable

Un décalage d'origine réglable est en principe un frame (Page 311) réglable. C'est pourquoi les composants ou valeurs de frame suivants sont également disponibles pour un décalage d'origine réglable :

- Décalage
- Rotation

- Homothétie
- Echelle



① Position initiale dans SCB
② Décalage
③ Décalage + rotation
④ Décalage + homothétie

L'indication de valeurs de frame pour les décalages d'origine réglables s'effectue via l'interface utilisateur :

SINUMERIK Operate : Groupe fonctionnel "Paramètres" > "Décalages d'origine" > "Détails"

Nombre paramétrable de frames réglables (G505 - G599)

Le nombre de décalages d'origine réglables propres à l'utilisateur (G505 - G599) peut être réglé en fonction du canal via :

PM28080 $MC_MM_NUM_USER_FRAMES = <nombre>

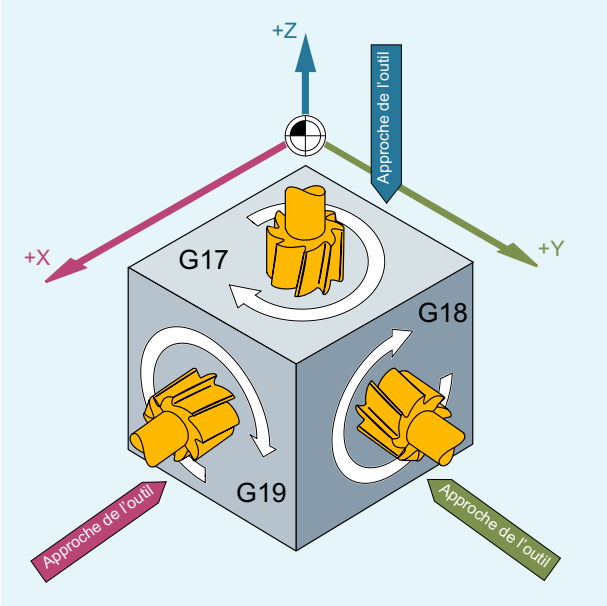
Voir aussi

Décalage d'origine programmable (G58, G59) (Page 320)

2.8.3 Sélection du plan de travail (G17/G18/G19)

Le fait de sélectionner le plan de travail, dans lequel le contour devra être usiné, détermine en même temps les fonctions suivantes :

- le plan pour la correction du rayon d'outil.
- l'axe de pénétration pour la correction de longueur d'outil en fonction du type d'outil.
- le plan pour l'interpolation circulaire.



Syntaxe

```
G17/G18/G19 ...
```

Signification

`G17` :	Plan de travail X/Y Axe de pénétration Z sélection du plan 1er - 2e axe géométrique
`G18` :	Plan de travail Z/X Axe de pénétration Y sélection du plan 3e - 1er axe géométrique
`G19` :	Plan de travail Y/Z Axe de pénétration X sélection du plan 2e - 3e axe géométrique

Remarque

Le réglage de base est G17 (plan X/Y) pour le fraisage et G18 (plan Z/X) pour le tournage.

Lors de l'appel de la correction de la trajectoire de l'outil G41/G42 (voir chapitre "Corrections de rayon d'outil (Page 256)"), le plan de travail doit être spécifié afin que la commande puisse effectuer la correction de la longueur et du rayon de l'outil.

Exemple

La marche à suivre "classique" pour le fraisage est :

1. Définir le plan de travail (`G17` réglage de base pour le fraisage)
2. Appel du type d'outil (`T`) et des valeurs de correction d'outil (`D`)

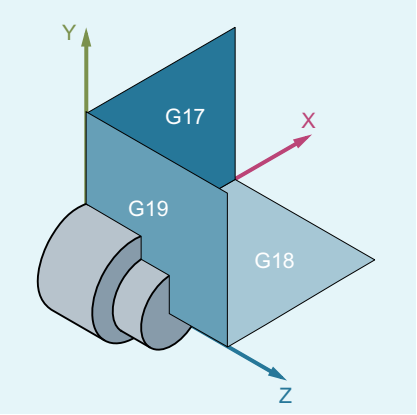
3. Activer la correction de trajectoire (G41)
4. Programmer les déplacements

Code de programme	**Commentaire**
N10 G17 T5 D8	; Appel du plan de travail X/Y, appel de l'outil. La correction de longueur d'outil se fait dans l'axe Z.
N20 G1 G41 X10 Y30 Z-5 F500	; La correction du rayon s'effectue dans le plan X/Y
N30 G2 X22.5 Y40 I50 J40	; La correction de l'interpolation circulaire/du rayon de l'outil s'effectue dans le plan X/Y

Informations complémentaires

Généralités

Il est recommandé de définir le plan de travail G17 à G19 au début du programme. Dans le réglage de base, le plan Z/X prédéfini pour le tournage est G18 :

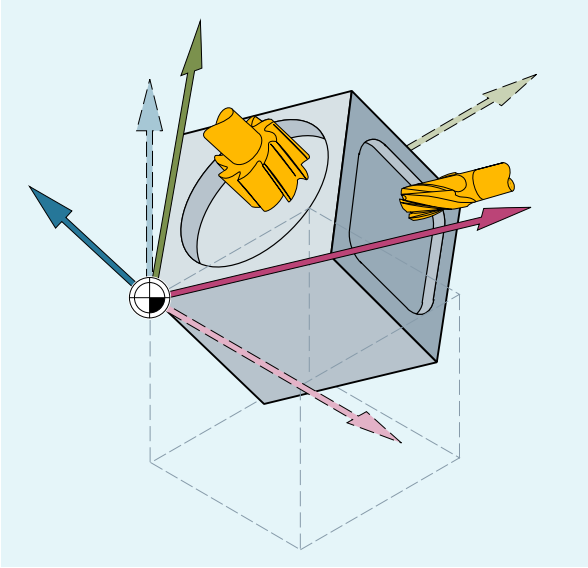


Pour le calcul du sens de rotation, la commande a besoin de l'indication du plan de travail. Des informations complémentaires à ce sujet figurent au chapitre "Interpolation circulaire (Page 193)".

Usinage sur des plans obliques

En faisant pivoter le système de coordonnées avec ROT (Page 322), les axes de coordonnées basculent dans le plan oblique. Les plans de travail pivotent en même temps.

Correction de longueur d'outil dans un plan oblique



D'une manière générale, la correction de longueur d'outil est toujours calculée par référence au plan de travail figé dans l'espace, non pivoté.

Remarque

Les fonctionnalités de la "correction de longueur d'outil pour outils orientables" permettent de calculer les composantes de la longueur d'outil adaptées aux plans de travail qui ont été pivotés.

Le choix du plan de correction s'effectue avec CUT2D, CUT2DF. Pour de plus amples informations à ce sujet et sur cette possibilité de calcul, voir le chapitre "Corrections de rayon d'outil (Page 256)".

Pour la définition dans l'espace du plan de travail, la commande numérique dispose de possibilités de transformation de coordonnées très confortables. Des informations complémentaires à ce sujet figurent au chapitre "Transformations des coordonnées (Frames) (Page 311)".

2.8.4 Cotes

La base de la plupart des programmes CN est un dessin de la pièce avec des cotes concrètes.

Ces cotes peuvent être indiquées de la manière suivante :

- Cotes absolues ou cotes relatives
- Millimètres ou pouces
- Rayon ou diamètre (pour le tournage)

Pour que les cotes d'un dessin puissent être appliquées directement (sans conversion) au programme CN, l'utilisateur dispose d'instructions de programmation spécifiques pour les différents types de cotes possibles.

2.8.4.1 Indication de cotes absolues (G90, AC)

En cotes absolues, les positions se rapportent toujours à l'origine du système de coordonnées valide à l'instant, autrement dit l'utilisateur programme la position absolue que l'outil doit accoster.

Indication de cotes absolues modale

L'indication de cotes absolues modale est activée par l'instruction `G90`. Elle est valable pour tous les axes qui sont programmés dans les blocs CN qui suivent.

Indication de cotes absolues non modale

Lorsque l'indication de cotes relatives (`G91`) est préréglée, l'instruction `AC` permet de régler l'indication de cotes absolues à effet non modal pour des axes individuels.

Remarque

L'indication de cotes absolues non modale (`AC`) est également possible pour les positionnements de broche (`SPOS`, `SPOSA`) et les paramètres d'interpolation (`I`, `J`, `K`).

Syntaxe

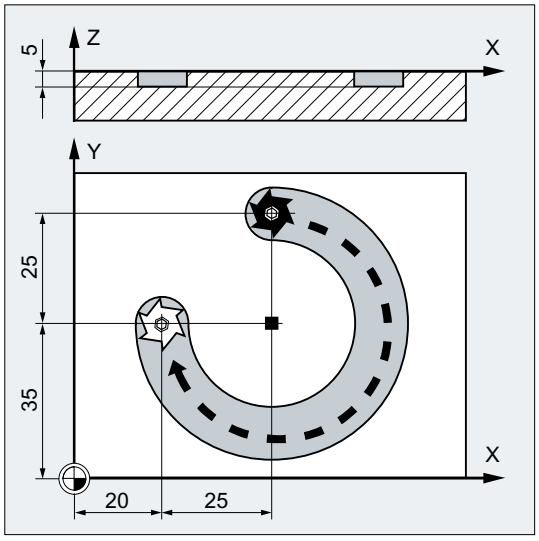
```
G90
<Axe>=AC(<valeur>)
```

Signification

`G90` :	Instruction d'activation de l'indication de cotes absolues modale
`AC` :	Instruction d'activation de l'indication de cotes absolues non modale
`<axe>` :	Descripteur de l'axe à déplacer
`<valeur>` :	Position de consigne de l'axe à déplacer en cotes absolues

Exemples

Exemple 1 : fraisage

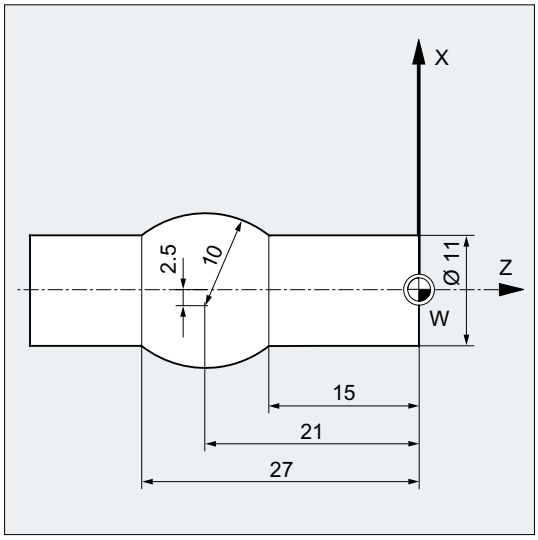


Code de programme	Commentaire
N10 G90 G0 X45 Y60 Z2 T1 S2000 M3	; Introduction de cotes absolues, accostage de la position XYZ en vitesse rapide, sélection d'outil, broche en marche avec sens de rotation horaire
N20 G1 Z-5 F500	; Interpolation linéaire, approche de l'outil
N30 G2 X20 Y35 I=AC(45) J=AC(35)	; Interpolation circulaire dans le sens horaire, point final et centre du cercle en cotes absolues
N40 G0 Z2	; Dégagement
N50 M30	; Fin de bloc

Remarque

Pour l'introduction des coordonnées I et J du centre du cercle, voir le chapitre "Interpolation circulaire".

Exemple 2 : tournage



Code de programme	**Commentaire**
`N5 T1 D1 S2000 M3`	`; Mise en place de l'outil T1, broche en marche avec sens de rotation horaire`
`N10 G0 G90 X11 Z1`	`; Introduction de cotes absolues, accostage de la position XZ en vitesse rapide`
`N20 G1 Z-15 F0.2`	`; Interpolation linéaire, approche de l'outil`
`N30 G3 X11 Z-27 I=AC(-5) K=AC(-21)`	`; Interpolation circulaire dans le sens antihoraire, point final et centre du cercle en cotes absolues`
`N40 G1 Z-40`	`; Dégagement`
`N50 M30`	`; Fin de bloc`

Remarque

Pour l'introduction des coordonnées I et J du centre du cercle, voir le chapitre "Interpolation circulaire".

Voir aussi

Indication de cotes absolues et de cotes relatives pour le tournage et le fraisage (G90/G91) (Page 159)

2.8.4.2 Indication de cotes relatives (G91, IC)

En cotes relatives, la position se rapporte au dernier point accosté, autrement dit la programmation en cotes relatives décrit de combien l'outil doit se déplacer.

Indication de cotes relatives modale

L'indication de cotes relatives modale est activée par l'instruction `G91`. Elle est valable pour tous les axes qui sont programmés dans les blocs CN qui suivent.

Indication de cotes relatives non modale

Lorsque l'indication de cotes absolues (G90) est préréglée, l'instruction IC permet de régler l'indication de cotes relatives à effet non modal pour des axes individuels.

Remarque

L'indication de cotes relatives non modale (IC) est également possible pour les positionnements de broche (SPOS, SPOSA) et les paramètres d'interpolation (I, J, K).

Syntaxe

```
G91
<Axe>=IC(<valeur>)
```

Signification

G91 :	Instruction d'activation de l'indication de cotes relatives modale
IC :	Instruction d'activation de l'indication de cotes relatives non modale
<axe> :	Descripteur de l'axe à déplacer
<valeur> :	Position de consigne de l'axe à déplacer en cotes relatives

Extension de G91

Pour certaines applications telles que l'effleurement, seule la course programmée est exécutée lorsque l'indication en cotes relatives est activée. Le déplacement correspondant au décalage d'origine activé ou à la correction de longueur d'outil n'est pas exécuté.

Ce comportement est réglable séparément pour le décalage d'origine activé et la correction de longueur d'outil au moyen des données de réglage suivantes :

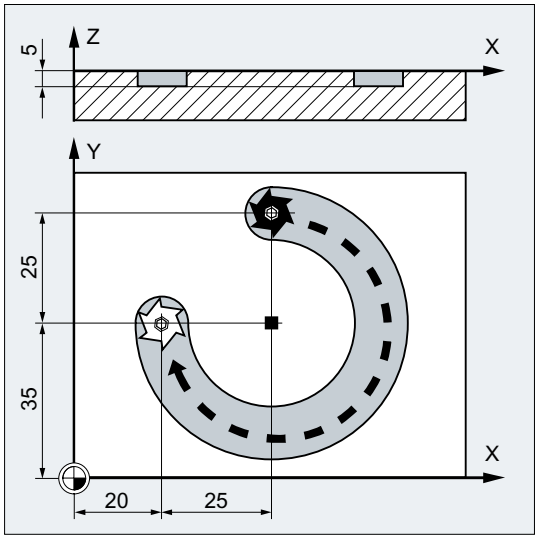
SD42440 $SC_FRAME_OFFSET_INCR_PROG (décalages d'origine dans des frames)

SD42442 $SC_TOOL_OFFSET_INCR_PROG (corrections de longueur d'outil)

Valeur	Signification
0	Lorsqu'un axe est programmé de manière relative (indication de cotes relatives), le déplacement correspondant au décalage d'origine activé ou à la correction de longueur d'outil n'est **pas** exécuté.
1	Lorsqu'un axe est programmé de manière relative (indication de cotes relatives), le déplacement correspondant au décalage d'origine activé ou à la correction de longueur d'outil est exécuté.

Exemples

Exemple 1 : fraisage

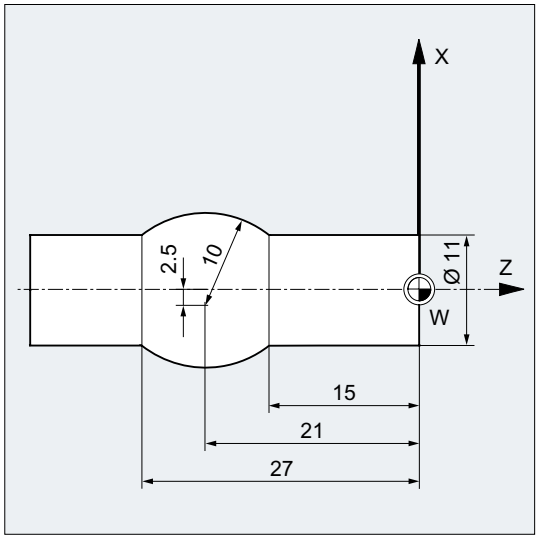


Code de programme	Commentaire
`N10 G90 G0 X45 Y60 Z2 T1 S2000 M3`	`; Introduction de cotes absolues, accostage de la position XYZ en vitesse rapide, sélection d'outil, broche en marche avec sens de rotation horaire`
`N20 G1 Z-5 F500`	`; Interpolation linéaire, approche de l'outil`
`N30 G2 X20 Y35 I0 J-25`	`; Interpolation circulaire dans le sens horaire, point final du cercle en cotes absolues, centre du cercle en cotes relatives`
`N40 G0 Z2`	`; Dégagement`
`N50 M30`	`; Fin de bloc`

Remarque

Pour l'introduction des coordonnées I et J du centre du cercle, voir le chapitre "Interpolation circulaire".

Exemple 2 : tournage



Code de programme	Commentaire
N5 T1 D1 S2000 M3	; Mise en place de l'outil T1, broche en marche avec sens de rotation horaire
N10 G0 G90 X11 Z1	; Indication de cotes absolues, accostage de la position XZ en vitesse rapide
N20 G1 Z-15 F0.2	; Interpolation linéaire, approche de l'outil
N30 G3 X11 Z-27 I-8 K-6	; Interpolation circulaire dans le sens antihoraire, point final du cercle en cotes absolues, centre du cercle en cotes relatives
N40 G1 Z-40	; Dégagement
N50 M30	; Fin de bloc

Remarque

Pour l'introduction des coordonnées I et J du centre du cercle, voir le chapitre "Interpolation circulaire".

Exemple 3 : indication de cotes relatives sans exécution du déplacement correspondant au décalage d'origine activé

Réglages :

- G54 contient un décalage de 25 sur l'axe X.
- SD42440 $SC_FRAME_OFFSET_INCR_PROG = 0

Code de programme	Commentaire
N10 G90 G0 G54 X100	
N20 G1 G91 X10	; L'indication de cotes relatives est activée, déplacement de 10 mm sur l'axe X (le déplacement correspondant au décalage d'origine n'est pas exécuté)

Code de programme	Commentaire
N30 G90 X50	; L'indication de cotes absolues est activée, accostage de la position X75 (le déplacement correspondant au décalage d'origine est exécuté)

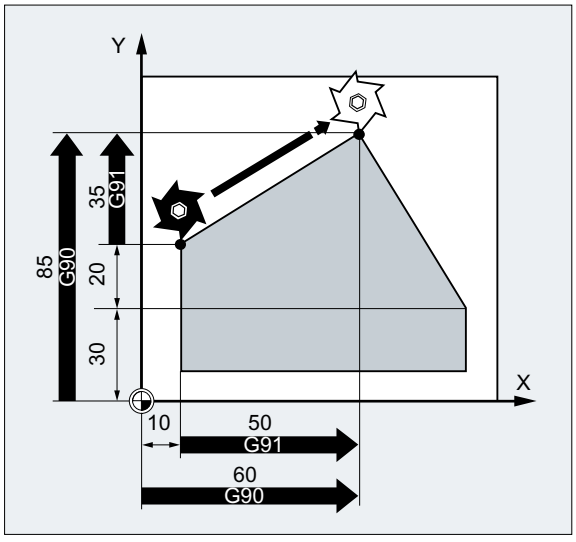
Voir aussi

Indication de cotes absolues et de cotes relatives pour le tournage et le fraisage (G90/G91) (Page 159)

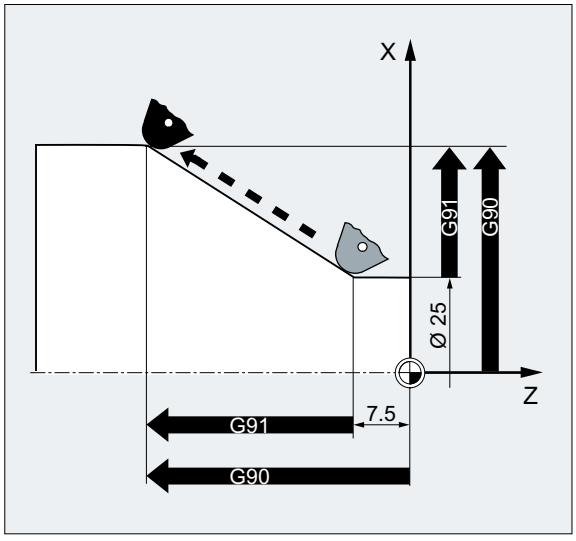
2.8.4.3 Indication de cotes absolues et de cotes relatives pour le tournage et le fraisage (G90/G91)

Les deux figures ci-dessous illustrent la programmation en cotes absolues (G90) et en cotes relatives (G91) à partir d'exemples de tournage et de fraisage.

Fraisage :



Tournage :



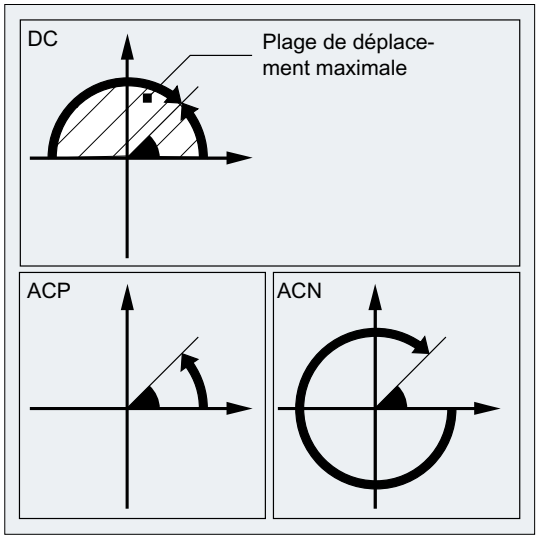
Remarque

Sur les tours classiques, on a l'habitude de considérer les déplacements relatifs dans l'axe transversal comme des valeurs de rayons, alors que les cotes absolues sont indiquées au diamètre. Ce réglage pour `G90` s'effectue avec les instructions `DIAMON`, `DIAMOF` et `DIAM90`.

2.8.4.4 Introduction en cotes absolues pour axes rotatifs (DC, ACP, ACN)

Pour le positionnement d'axes rotatifs en cotes absolues, les instructions DC, ACP et ACN non modales et indépendantes de G90/G91, sont disponibles.

DC, ACP et ACN se distinguent par la stratégie d'accostage utilisée :



Syntaxe

```
<Axe rotatif>=DC(<valeur>)
<Axe rotatif>=ACP(<valeur>)
<Axe rotatif>=ACN(<valeur>)
```

Signification

`<axe rotatif>` :	Descripteur de l'axe rotatif à déplacer (p. ex. A, B ou C)	
`DC` :	Instruction d'accostage **direct** de la position L'axe rotatif rejoint la position programmée par le chemin direct le plus court. Le déplacement maximum d'un axe rotatif est de 180°.	
`ACP` :	Instruction d'accostage de la position dans le sens **positif** L'axe rotatif rejoint la position programmée dans le sens de rotation positif de l'axe (sens antihoraire).	
`ACN` :	Instruction d'accostage de la position dans le sens **négatif** L'axe rotatif rejoint la position programmée dans le sens de rotation négatif de l'axe (sens horaire).	
`<valeur>` :	Position d'axe rotatif à accoster en cotes absolues	
	Plage de valeurs :	0 - 360 degrés

Remarque

Le sens de rotation positif (sens horaire ou antihoraire) est défini par un paramètre machine.

Remarque

Pour le positionnement avec indication de sens (ACP, ACN), la plage de déplacement entre 0° et 360° doit avoir été réglée dans les paramètres machine (comportement modulo). Pour déplacer des axes rotatifs modulo de plus de 360° dans un bloc, il convient de programmer G91 ou IC.

Remarque

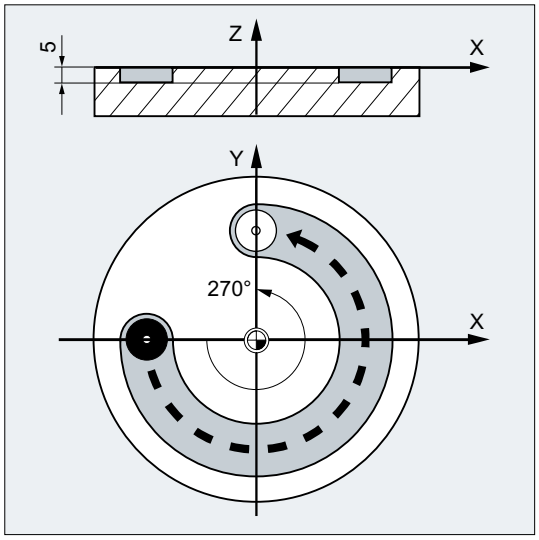
Les instructions DC, ACP et ACN peuvent également être utilisées pour le positionnement de broche (SPOS, SPOSA) depuis l'arrêt.

Exemple : SPOS=DC(45)

Exemple

Opération de fraisage sur table tournante

L'outil est immobile, le plateau tourne jusqu'à la position de 270° **dans le sens horaire**. Ceci engendre une rainure sur cercle.



Code de programme	Commentaire
N10 SPOS=0	; Broche en asservissement de position
N20 G90 G0 X-20 Y0 Z2 T1	; Indication de cotes absolues, approche de l'outil T1 en vitesse rapide
N30 G1 Z-5 F500	; Abaissement de l'outil en vitesse d'avance
N40 C=ACP(270)	; Le plateau tourne jusqu'à la position de 270 degrés dans le sens horaire (positif), l'outil fraise une rainure sur cercle
N50 G0 Z2 M30	; Relèvement, fin du programme

2.8.4.5 Système de mesure en pouces / métrique (G70/G71, G700/G710)

Les instructions du groupe G13 (système de mesure anglo-saxon/métrique) permettent de basculer à l'intérieur du programme pièce entre le système de mesure métrique et le système de mesure anglo-saxon.

Activation

Pour les instructions G700 et G710 soient disponibles, la fonctionnalité étendue du système de mesure doit être activée (PM10260 $MN_CONVERT_SCALING_SYSTEM = 1).

Syntaxe

```
G70
G71
G700
G710
```

Signification

G70 :	Activation du système de mesure anglo-saxon La lecture et l'écriture des **données géométriques de longueur** s'effectuent dans le **système de mesure anglo-saxon**. La lecture et l'écriture des **données technologiques de longueur** (par ex. avances, corrections d'outil, décalages d'origine réglables, paramètres machine et variables système) s'effectuent dans le **système de base paramétré**.	
	groupe G :	13
	Position de base :	Réglable par PM20150 $MC_GCODE_RESET_VALUES
	Prise d'effet :	modale
G71 :	Activation du système de mesure métrique La lecture et l'écriture des **données géométriques de longueur** s'effectuent dans le **système de mesure métrique**. La lecture et l'écriture des **données technologiques de longueur** (par ex. avances, corrections d'outil, décalages d'origine réglables, paramètres machine et variables système) s'effectuent dans le **système de base paramétré**.	
	groupe G :	13
	Position de base :	Réglable par PM20150 $MC_GCODE_RESET_VALUES
	Prise d'effet :	modale
G700 :	Activation du système de mesure anglo-saxon La lecture et l'écriture de l'ensemble des données géométriques et caractéristiques techniques de longueur s'effectuent dans le système de mesure anglo-saxon.	
	groupe G :	13
	Position de base :	Réglable par PM20150 $MC_GCODE_RESET_VALUES
	Prise d'effet :	modale
G710 :	Activation du système de mesure métrique La lecture et l'écriture de l'ensemble des données géométriques et caractéristiques techniques de longueur s'effectuent dans le système de mesure métrique.	
	groupe G :	13
	Position de base :	Réglable par PM20150 $MC_GCODE_RESET_VALUES
	Prise d'effet :	modale

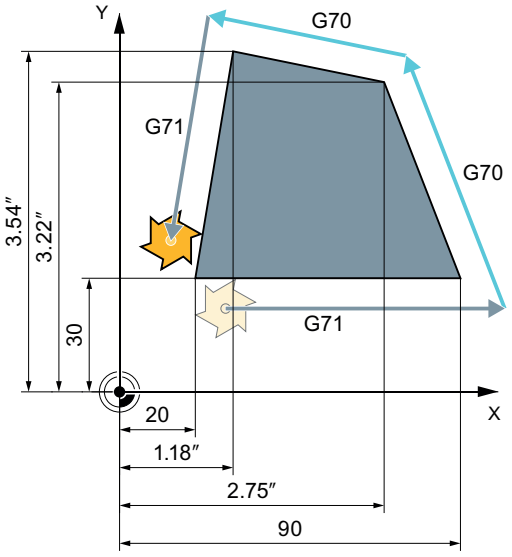
IMPORTANT

Données spécifiques à un axe des axes rotatifs

La lecture et l'écriture des données spécifiques à un axe des axes rotatifs s'effectuent toujours dans le système de base paramétré.

Exemple

Le système de base est métrique (PM10240 $MN_SCALING_SYSTEM_IS_METRIC = 1). Toutefois, le plan de la pièce contient des indications de cotes en inch. C'est pourquoi la commutation dans le système de mesure anglo-saxon s'opère à l'intérieur du programme pièce. Après l'introduction des cotes en inch, la bascule a lieu à nouveau dans le système métrique.



Code de programme	Commentaire
N10 G0 G90 X20 Y30 Z2 S2000 M3 T1	; X=20 mm, Y=30 mm, Z=2 mm, F=vitesse rapide mm/min
N20 G1 Z-5 F500	; Z=-5 mm, F=500 mm/min
N30 X90	; X=90 mm
N40 **G70** X2.75 Y3.22	; Système de mesure programmé : anglo-saxon
	; X=2,75 inch, Y=3,22 inch, F=500 mm/min
N50 X1.18 Y3.54	; X=1,18 inch, Y=3,54 inch, F=500 mm/min
N60 **G71** X20 Y30	; Système de mesure programmé : métrique
	; X=20 mm, Y=30 mm, F=500 mm/min
N70 G0 Z2	; Z=2 mm, F=vitesse rapide mm/min
N80 M30	; Fin du programme

informations complémentaires

Lecture et écriture de données pour G70/G71 et G700/G710

Zone de données	G70 / G71		G700 / G710	
	Lecture	Écriture	Lecture	Écriture
Affichage, décimales (SCP)	P	P	P	P
Affichage, décimales (SCM)	G	G	G	G
Avances	G	G	P	P
Indications de position X, Y, Z	P	P	P	P
Paramètres d'interpolation I, J, K	P	P	P	P

Zone de données	G70 / G71		G700 / G710	
	Lecture	**Écriture**	**Lecture**	**Écriture**
Rayon de cercle (CR)	P	P	P	P
Rayon polaire (RP)	P	P	P	P
Pas de vis	P	P	P	P
FRAME programmable	P	P	P	P
FRAMES réglables	G	G	P	P
Frames de base	G	G	P	P
Décalages d'origine externes	G	G	P	P
Décalage Preset axial	G	G	P	P
Limites de la zone de travail (G25/G26)	G	G	P	P
Zones de protection	P	P	P	P
Corrections d'outil	G	G	P	P
Paramètres machine relatifs à des longueurs	G	G	P	P
Données de réglage relatives à des longueurs	G	G	P	P
Variables système relatives à des longueurs	G	G	P	P
Variables utilisateur globales	G	G	G	G
Variables utilisateur locales	G	G	G	G
Variables utilisateur du programme	G	G	G	G
Paramètres R	G	G	G	G
Cycles Siemens	P	P	P	P
Évaluation des incréments JOG/manivelle	G	G	G	G
P : L'écriture / la lecture se font dans le système de mesure programmé G : L'écriture / la lecture se font dans le système de base programmé				

Actions synchrones

Remarque

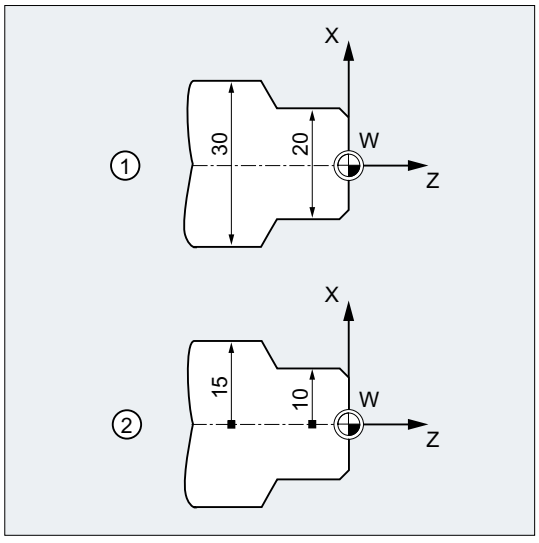
Lecture de données de positionnement dans les actions synchrones

En l'absence de programmation explicite du système de mesure dans l'action synchrone (partie condition et/ou partie action ou fonction technologique), les **données de positionnement relatives à une longueur** de l'action synchrone sont toujours lues dans le **système de base** paramétré.

Informations complémentaires : Description fonctionnelle Actions synchrones

2.8.4.6 Programmation diamétrale/radiale spécifique au canal (DIAMON, DIAM90, DIAMOF, DIAMCYCOF)

Dans le cas du tournage, les cotes **pour l'axe transversal** peuvent être indiquées de manière diamétrale (①) ou radiale (②) :



Afin que les cotes puissent être appliquées directement dans le programme CN, sans conversion à partir du schéma technique, la programmation diamétrale/radiale spécifique au canal peut être activée avec les instructions modales `DIAMON`, `DIAM90`, `DIAMOF` et `DIAMCYCOF`.

Remarque

La programmation diamétrale/radiale spécifique au canal se rapporte à l'axe géométrique défini en tant qu'axe transversal au moyen du paramètre machine PM20100 $MC_DIAMETER_AX_DEF (→ voir les indications du constructeur de machine).

Le paramètre machine PM20100 permet uniquement de définir un axe transversal par canal.

Syntaxe

```
DIAMON
DIAM90
DIAMOF
```

Signification

`DIAMON` :	Instruction d'activation de la programmation diamétrale spécifique au canal et **indépendante** L'effet de `DIAMON` est indépendant du mode programmé pour les cotes (cotes absolues `G90` ou cotes relatives `G91`) :	
	• pour G90 :	Cotes diamétrales
	• pour G91 :	Cotes diamétrales

DIAM90 :	Instruction d'activation de la programmation diamétrale spécifique au canal et **dépendante** L'effet de DIAM90 dépend du mode programmé pour les cotes :	
	• pour G90 :	Cotes diamétrales
	• pour G91 :	Indication de cotes au rayon
DIAMOF :	Instruction de désactivation de la programmation diamétrale spécifique au canal La désactivation de la programmation diamétrale entraîne l'activation de la programmation radiale spécifique au canal. L'effet de DIAMOF est indépendant du mode programmé pour les cotes :	
	• pour G90 :	Indication de cotes au rayon
	• pour G91 :	Indication de cotes au rayon
DIAMCYCOF :	Instruction de désactivation de la programmation diamétrale spécifique au canal pendant l'exécution du cycle Le mode radial des calculs est ainsi toujours possible dans le cycle. Pour l'affichage de position et l'affichage de base du bloc, la dernière instruction G active de ce groupe reste active.	

Remarque

Avec DIAMON ou DIAM90, les valeurs réelles de l'axe transversal sont toujours affichées comme diamètre. Cela est aussi valable pour la lecture des valeurs réelles dans le système de coordonnées pièce avec MEAS, MEAW, $P_EP[x] et $AA_IW[x].

Exemple

Code de programme	Commentaire
N10 G0 X0 Z0	; Accostage du point de départ
N20 DIAMOF	; Programmation diamétrale désactivée
N30 G1 X30 S2000 M03 F0.7	; Axe X = axe transversal, programmation radiale activée, accostage de la position radiale X30
N40 DIAMON	; La programmation diamétrale est activée pour l'axe transversal
N50 G1 X70 Z-20	; Accostage de la position diamétrale X70 et Z-20
N60 Z-30	
N70 DIAM90	; Programmation diamétrale pour la cote absolue et programmation radiale pour la cote relative
N80 G91 X10 Z-20	; Cote relative activée
N90 G90 X10	; Cote absolue activée
N100 M30	; Fin du programme.

Informations complémentaires

Valeurs de diamètre (DIAMON/DIAM90)

Les valeurs de diamètre sont valables pour les données suivantes :

- Affichage de la position réelle de l'axe transversal dans le système de coordonnées pièce
- Mode JOG : Incréments pour le mode manuel incrémental et le déplacement par manivelle
- Programmation de positions finales :
  Paramètres d'interpolation `I`, `J`, `K` avec `G2`/`G3`, si ceux-ci sont programmés de manière absolue avec `AC`.
  En cas de programmation incrémentale (`IC`) de `I`, `J`, `K`, le rayon est toujours pris en compte.
- Lecture de valeurs réelles dans le système de coordonnées pièce avec :
  `MEAS`, `MEAW`, `$P_EP[X]`, `$AA_IW[X]`

2.8.4.7 Programmation diamétrale/radiale spécifique à l'axe (DIAMONA, DIAM90A, DIAMOFA, DIACYCOFA, DIAMCHANA, DIAMCHAN, DAC, DIC, RAC, RIC)

En plus de la programmation diamétrale spécifique au canal, la programmation diamétrale spécifique à l'axe permet d'obtenir l'indication de cotes et l'affichage au diamètre avec un effet modal ou non modal pour un ou plusieurs axes.

Remarque

La programmation diamétrale spécifique à l'axe est uniquement possible pour les axes qui sont autorisés par le paramètre machine PM30460 $MA_BASE_FUNCTION_MASK comme autres axes transversaux pour la programmation diamétrale spécifique à l'axe (→ voir les indications du constructeur de machine).

Syntaxe

Programmation diamétrale spécifique à l'axe et modale pour plusieurs axes transversaux dans le canal :

```
DIAMONA[<axe>]
DIAM90A[<axe>]
DIAMOFA[<axe>]
DIACYCOFA[<axe>]
```

Application de la programmation diamétrale/radiale spécifique au canal :

```
DIAMCHANA[<axe>]
DIAMCHAN
```

Programmation diamétrale/radiale spécifique à l'axe et non modale :

```
<Axe>=DAC(<valeur>)
<Axe>=DIC(<valeur>)
<Axe>=RAC(<valeur>)
<Axe>=RIC(<valeur>)
```

Signification

Programmation diamétrale spécifique à l'axe et modale		
DIAMONA :	Instruction d'activation de la programmation diamétrale spécifique à l'axe et **indépendante** L'effet de DIAMONA est indépendant du mode programmé pour les cotes (G90/G91 ou AC/IC) :	
	• pour G90, AC :	Cotes diamétrales
	• pour G91, IC :	Cotes diamétrales
DIAM90A :	Instruction d'activation de la programmation diamétrale spécifique à l'axe et **dépendante** L'effet de DIAM90A dépend du mode programmé pour les cotes :	
	• pour G90, AC :	Cotes diamétrales
	• pour G91, IC :	Indication de cotes au rayon
DIAMOFA :	Instruction de désactivation de la programmation diamétrale spécifique à l'axe La désactivation de la programmation diamétrale entraîne l'activation de la programmation radiale spécifique à l'axe. L'effet de DIAMOFA est indépendant du mode programmé pour les cotes :	
	• pour G90, AC :	Indication de cotes au rayon
	• pour G91, IC :	Indication de cotes au rayon
DIACYCOFA :	Instruction de désactivation de la programmation diamétrale spécifique à l'axe pendant l'exécution du cycle Le mode radial des calculs est ainsi toujours possible dans le cycle. Pour l'affichage de position et l'affichage de base du bloc, la dernière instruction G active de ce groupe reste active.	
<axe> :	Descripteur de l'axe pour lequel la programmation diamétrale spécifique à l'axe doit être activée Les descripteurs d'axes autorisés sont : • Nom d'axe géométrique/d'axe de canal ou • Nom d'axe machine	
	Plage de valeurs :	L'axe indiqué doit être un axe connu dans le canal. Autres conditions : • L'axe doit être autorisé pour la programmation diamétrale spécifique à l'axe par le paramètre machine PM30460 $MA_BASE_FUNCTION_MASK. • Les axes rotatifs ne sont pas autorisés comme axes transversaux.
Application de la programmation diamétrale/radiale spécifique au canal		
DIAMCHANA :	L'instruction DIAMCHANA[<axe>] applique l'état du canal de la programmation diamétrale/radiale à l'**axe indiqué** qui est ainsi affecté à la programmation diamétrale/radiale spécifique au canal.	
DIAMCHAN :	L'instruction DIAMCHAN applique l'état du canal de la programmation diamétrale/radiale à **tous** les axes autorisés pour la programmation diamétrale spécifique à l'axe, ces axes étant ainsi affectés à la programmation diamétrale/radiale spécifique au canal.	

Programmation diamétrale/radiale spécifique à l'axe et non modale La programmation diamétrale/radiale spécifique à l'axe et non modale détermine le type de cotes comme valeur de diamètre ou de rayon dans le programme pièce, ainsi que les actions synchrones. L'état modal de la programmation diamétrale/radiale n'est pas modifié.	
DAC :	L'instruction DAC active le type de cotes non modal suivant pour l'axe indiqué : Diamètre en cotes absolues
DIC :	L'instruction DIC active le type de cotes non modal suivant pour l'axe indiqué : Diamètre en cotes relatives
RAC :	L'instruction RAC active le type de cotes non modal suivant pour l'axe indiqué : Rayon en cotes absolues
RIC :	L'instruction RIC active le type de cotes non modal suivant pour l'axe indiqué : Rayon en cotes relatives

Remarque

Avec DIAMONA[<axe>] ou DIAM90A[<axe>], les valeurs réelles de l'axe transversal sont toujours affichées comme diamètre. Cela est aussi valable pour la lecture des valeurs réelles dans le système de coordonnées pièce avec MEAS, MEAW, $P_EP[x] et $AA_IW[x].

Remarque

Lors de la permutation d'un axe transversal supplémentaire en raison d'une requête GET, l'état de la programmation diamétrale/radiale est repris dans l'autre canal avec RELEASE[<axe>].

Exemples

Exemple 1 : programmation diamétrale/radiale spécifique à l'axe et modale

X est l'axe transversal dans le canal, la programmation diamétrale spécifique à l'axe est autorisée pour Y.

Code de programme	Commentaire
N10 G0 X0 Z0 DIAMON	; Activation de la programmation diamétrale spécifique au canal pour X
N15 DIAMOF	; Désactivation de la programmation diamétrale spécifique au canal
N20 DIAMONA[Y]	; Activation de la programmation diamétrale spécifique à l'axe et modale pour Y
N25 X200 Y100	; Programmation radiale activée pour X
N30 DIAMCHANA[Y]	; Y prend en compte l'état de la programmation diamétrale/radiale spécifique au canal et lui est affecté.
N35 X50 Y100	; Programmation radiale activée pour X et Y
N40 DIAMON	; Activation de la programmation diamétrale spécifique au canal
N45 X50 Y100	; Programmation diamétrale activée pour X et Y

Exemple 2 : programmation diamétrale/radiale spécifique à l'axe et non modale

X est l'axe transversal dans le canal, la programmation diamétrale spécifique à l'axe est autorisée pour Y.

Code de programme	Commentaire
N10 DIAMON	; Activation de la programmation diamétrale spécifique au canal
N15 G0 G90 X20 Y40 DIAMONA[Y]	; Activation de la programmation diamétrale spécifique à l'axe et modale pour Y
N20 G01 X=RIC(5)	; Type de cotes X activé pour ce bloc : rayon en cotes relatives
N25 X=RAC(80)	; Type de cotes X activé pour ce bloc : rayon en cotes absolues
N30 WHEN $SAA_IM[Y]> 50 DO POS[X]=RIC(1)	; X est l'axe de commande Type de cotes X activé pour ce bloc : rayon en cotes relatives
N40 WHEN $SAA_IM[Y]>60 DO POS[X]=DAC(10)	; X est l'axe de commande Type de cotes X activé pour ce bloc : rayon en cotes absolues
N50 G4 F3	

Informations complémentaires

Valeurs de diamètre (DIAMONA/DIAM90A)

Les valeurs de diamètre sont valables pour les données suivantes :

- Affichage de la position réelle de l'axe transversal dans le système de coordonnées pièce
- Mode JOG : Incréments pour le mode manuel incrémental et le déplacement par manivelle
- Programmation de positions finales :
  Paramètres d'interpolation I, J, K avec G2/G3, si ceux-ci sont programmés de manière absolue avec AC.
  En cas de programmation incrémentale IC de I, J, K, le rayon est toujours pris en compte.
- Lecture de valeurs réelles dans le système de coordonnées pièce avec :
  MEAS, MEAW, $P_EP[X], $AA_IW[X]

Programmation diamétrale spécifique à l'axe et non modale (DAC, DIC, RAC, RIC)

Les instructions DAC, DIC, RAC, RIC sont autorisées pour toutes les instructions pour lesquelles la programmation diamétrale spécifique au canal est prise en compte :

- Position de l'axe : X..., POS, POSA
- Oscillation : OSP1, OSP2, OSS, OSE, POSP
- Paramètres d'interpolation : I, J, K
- Élément de contour : droite avec indication d'angle
- Retrait rapide : POLF[AX]

- Déplacement dans le sens de l'outil : MOVT
- Accostage et retrait en douceur :
  G140 à G143, G147, G148, G247, G248, G347, G348, G340, G341

2.8.5 Position de la pièce lors du tournage

Désignations des axes

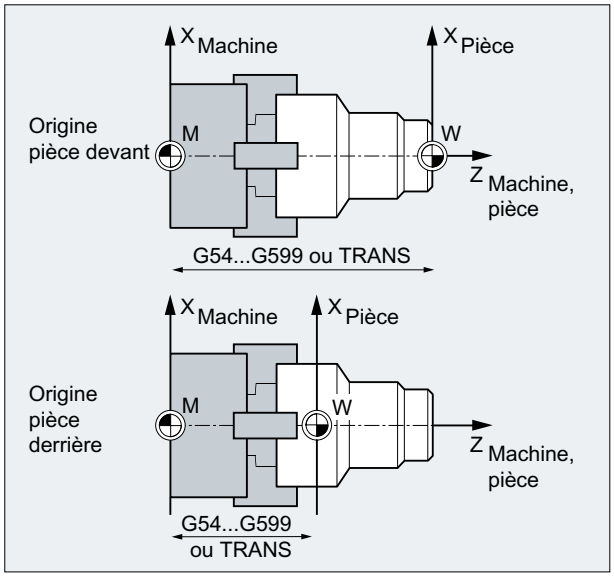
La dénomination habituelle des deux axes géométriques orthogonaux est la suivante :

Axe longitudinal	= axe Z (abscisse)
Axe transversal	= axe X (ordonnée)

Origine pièce

Alors que l'origine machine est prédéfinie et fixe, la position de l'origine pièce peut être choisie librement sur l'axe longitudinal. En général, on place l'origine pièce sur la partie frontale ou arrière de la pièce.

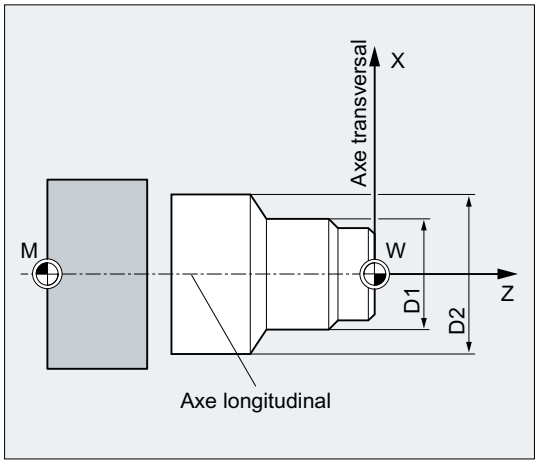
Les origines pièce et machine sont toutes deux placées sur le centre de rotation. De ce fait, le décalage réglable dans l'axe X est nul.



M	Origine machine
W	Origine pièce
Z	Axe longitudinal
X	Axe transversal
G54 à G599 ou TRANS	Appel pour la position de l'origine de la pièce

Axe transversal

Pour l'axe transversal, les cotes sont généralement indiquées au diamètre (cotes doubles par rapport aux autres axes).



L'axe géométrique utilisé comme axe transversal est à préciser dans les paramètres machine (→ constructeur de machine).

2.9 Instructions de déplacement

2.9.1 Informations générales pour les instructions de déplacement

Eléments de contour

Le contour de pièce programmé peut se composer des éléments de contour suivants :

- Droites
- Arcs de cercle
- Hélices (par superposition de droites et d'arcs de cercle)

Instructions de déplacement

Pour réaliser ces éléments de contour, on dispose de différentes instructions de déplacement :

- Déplacement à vitesse rapide (`G0`)
- Interpolation linéaire (`G1`)
- Interpolation circulaire dans le sens horaire (`G2`)
- Interpolation circulaire dans le sens antihoraire (`G3`)

Les instructions de déplacement sont des fonctions modales.

Positions de destination

Un bloc de déplacement contient les positions de destination des axes à déplacer (axes à interpolation, axes synchrones, axes de positionnement).

La programmation des positions de destination peut avoir lieu en coordonnées cartésiennes ou polaires.

Remarque

Une adresse d'axe ne peut être programmée qu'une seule fois dans un bloc.

Point de départ - Point de destination

Le déplacement se fait toujours de la dernière position accostée à la position de destination programmée. Cette position de destination devient alors, à son tour, la position de départ dans l'instruction de déplacement suivante.

Contour de pièce

IMPORTANT
Action de l'outil non définie Avant de lancer une opération, l'outil doit être positionné de façon à exclure tout endommagement aussi bien de l'outil que de la pièce.

Exécutés successivement, les blocs de déplacement créent le contour de la pièce :

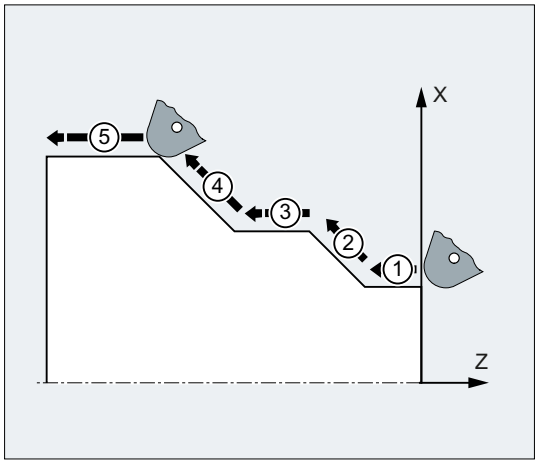


Figure 2-1 Blocs de déplacement pour le tournage

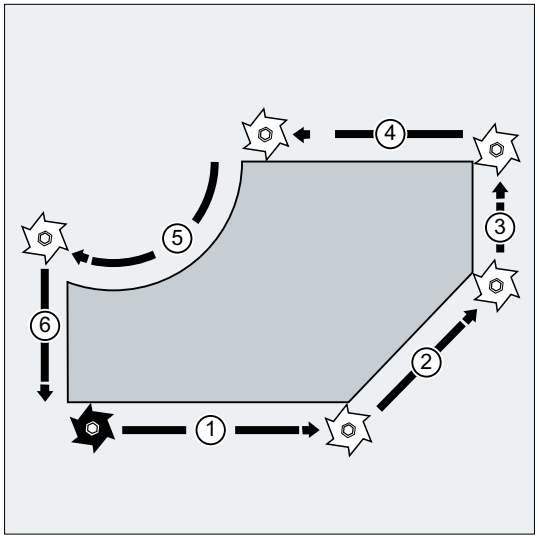


Figure 2-2 Blocs de déplacement pour le fraisage

2.9.2 Instructions de déplacement avec des coordonnées cartésiennes (G0, G1, G2, G3, X..., Y..., Z...)

La position indiquée dans le bloc CN en coordonnées cartésiennes peut être accostée par déplacement à vitesse rapide `G0`, interpolation linéaire `G1` ou interpolation circulaire `G2` /`G3`.

Syntaxe

```
G0 X... Y... Z...
G1 X... Y... Z...
G2 X... Y... Z... ...
G3 X... Y... Z... ...
```

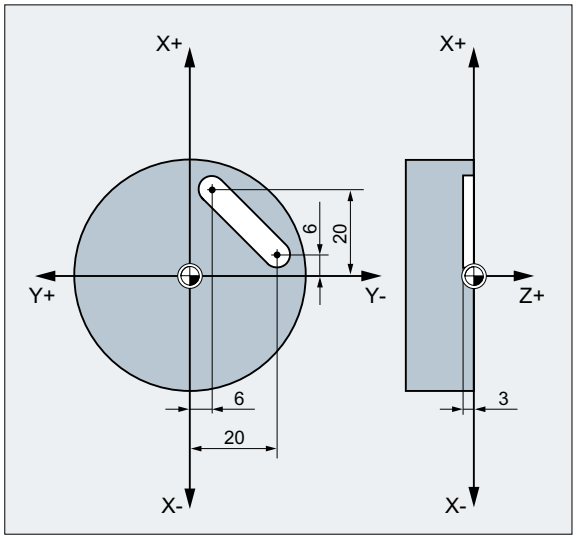
Signification

`G0` :	Instruction d'activation du déplacement à vitesse rapide
`G1` :	Instruction d'activation de l'interpolation linéaire
`G2` :	Instruction d'activation de l'interpolation circulaire dans le sens horaire
`G3` :	Instruction d'activation de l'interpolation circulaire dans le sens antihoraire
`X...` :	Coordonnée cartésienne de la position de destination en X
`Y...` :	Coordonnée cartésienne de la position de destination en Y
`Z...` :	Coordonnée cartésienne de la position de destination en Z

Remarque

Outres les coordonnées cartésiennes de la position de destination `X...`, `Y...`, `Z...`, d'autres indications sont nécessaires pour l'interpolation circulaire `G2` / `G3` (par exemple les coordonnées du centre du cercle; voir "Vue d'ensemble (Page 193)").

Exemple



Code de programme	Commentaire
N10 G17 S400 M3	; Sélection du plan de travail, broche sens horaire
N20 G0 X40 Y-6 Z2	; Accostage de la position de départ indiquée avec des coordonnées cartésiennes en vitesse rapide
N30 G1 Z-3 F40	; Activation de l'interpolation linéaire, approche de l'outil
N40 X12 Y-20	; Accostage de la position finale indiquée avec des coordonnées cartésiennes suivant une droite inclinée
N50 G0 Z100 M30	; Dégagement en vitesse rapide pour le changement d'outil

2.9.3 Instructions de déplacement avec des coordonnées polaires

2.9.3.1 Point de référence des coordonnées polaires (G110, G111, G112)

Le point qui sert d'origine pour la cotation s'appelle le pôle.

Le pôle peut être indiqué avec des coordonnées cartésiennes ou polaires.

Les instructions G110 à G112 permettent de définir un point de référence univoque pour les coordonnées polaires. L'indication de cotes absolues ou relatives n'a donc aucun effet.

Syntaxe

```
G110/G111/G112 X… Y… Z…
G110/G111/G112 AP=… RP=…
```

Signification

G110 ... :	Avec l'instruction G110, les coordonnées polaires qui suivent se rapportent **à la dernière position accostée**.	
G111 ... :	Avec l'instruction G111, les coordonnées polaires qui suivent se rapportent **à l'origine du système de coordonnées pièce courant**.	
G112 ... :	Avec l'instruction G112, les coordonnées polaires qui suivent se rapportent **au dernier pôle valide**.	
	Remarque : Les instructions G110...G112 sont à programmer dans un bloc CN spécifique.	
X… Y… Z… :	Indication du pôle en coordonnées cartésiennes	
AP=… RP=… :	Indication du pôle en coordonnées polaires	
	AP=… :	Angle polaire Angle situé entre le rayon polaire et l'axe horizontal du plan de travail (par exemple l'axe X pour G17). Le sens de rotation positif est le sens inverse des aiguilles d'une montre.
		Plage de valeurs : ± 0...360°
	RP=… :	Rayon polaire L'indication est **toujours en valeurs absolues positives**, en [mm] ou en [inch].

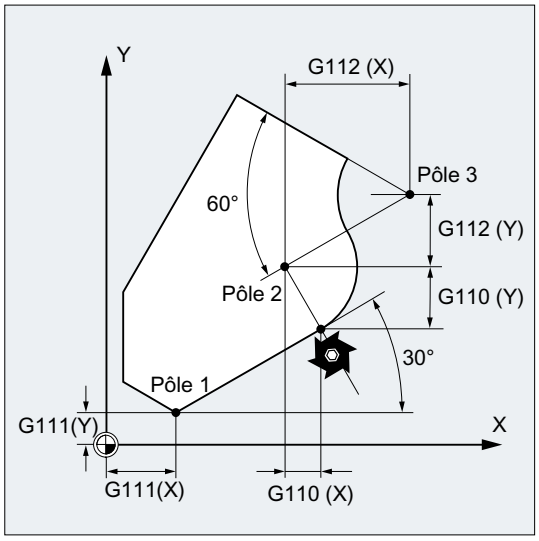
Remarque

Dans le programme CN, il set possible d'alterner d'un bloc à l'autre entre coordonnées polaires et coordonnées cartésiennes. L'utilisation de désignations cartésiennes (X..., Y..., Z...) permet de retourner directement au système cartésien. Le pôle défini est conservé jusqu'en fin de programme.

Remarque

Si aucun pôle n'a été défini, c'est l'origine du système de coordonnées pièce courant qui fait office de pôle.

Exemple

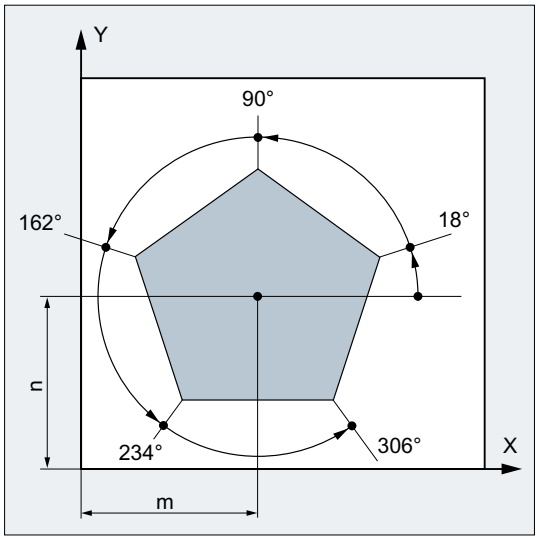


Les pôles 1 à 3 sont définis de la manière suivante :

- Pôle 1 avec `G111 X... Y...`
- Pôle 2 avec `G110 X... Y...`
- Pôle 3 avec `G112 X... Y...`

2.9.3.2 Instructions de déplacement avec des coordonnées polaires (G0, G1, G2, G3, AP, RP)

Les instructions de déplacement avec des coordonnées polaires sont utiles lorsque la cotation d'une pièce ou d'un élément de pièce part d'un point central et que les cotes sont indiquées par des angles et des rayons (par exemple dans le cas de réseaux de trous).



Syntaxe

```
G0/G1/G2/G3 AP=... RP=...
```

Signification

`G0` :	Instruction d'activation du déplacement à vitesse rapide	
`G1` :	Instruction d'activation de l'interpolation linéaire	
`G2` :	Instruction d'activation de l'interpolation circulaire dans le sens horaire	
`G3` :	Instruction d'activation de l'interpolation circulaire dans le sens antihoraire	
`AP` :	Angle polaire Angle situé entre le rayon polaire et l'axe horizontal du plan de travail (par exemple l'axe X pour `G17`). Le sens de rotation positif est le sens inverse des aiguilles d'une montre.	
	Plage de valeurs :	± 0...360°
	L'angle peut aussi bien être indiqué en coordonnées absolues que relatives.	
	`AP=AC(...)` :	introduction de cotes absolues
	`AP=IC(...)` :	Saisie de cotes relatives En cotes relatives, le dernier angle programmé sert de référence.
	L'angle polaire reste mémorisé jusqu'à ce qu'un nouveau pôle soit défini ou que le plan de travail soit modifié.	
`RP` :	Rayon polaire L'indication est **toujours en valeurs absolues positives**, en [mm] ou en [inch]. Le rayon polaire reste mémorisé jusqu'à son remplacement par l'introduction d'une nouvelle valeur.	

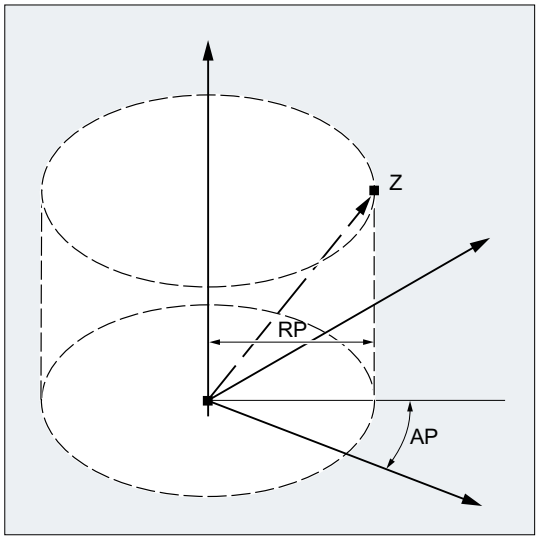
Remarque

Les coordonnées polaires se rapportent au pôle défini avec `G110` ... `G112` et sont valables dans le plan de travail sélectionné avec `G17` à `G19`.

Remarque

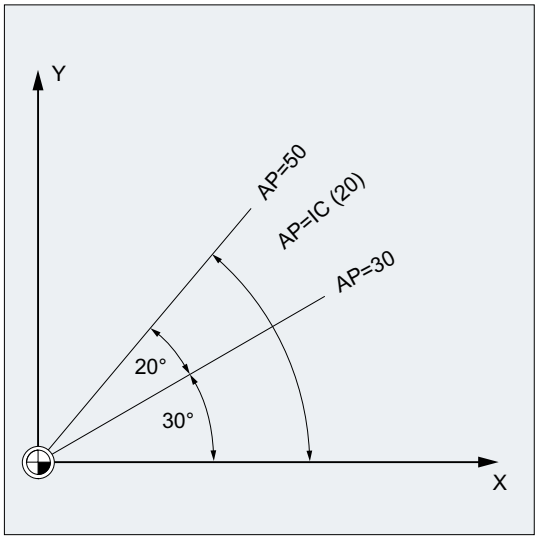
Le 3e axe géométrique, perpendiculaire au plan de travail, peut être défini en sus comme axe de coordonnées cartésiennes (voir figure suivante). Ainsi, il est possible de programmer des points dans l'espace en coordonnées cylindriques.

Exemple : `G17 G0 AP… RP… Z…`



Autres conditions à prendre en compte

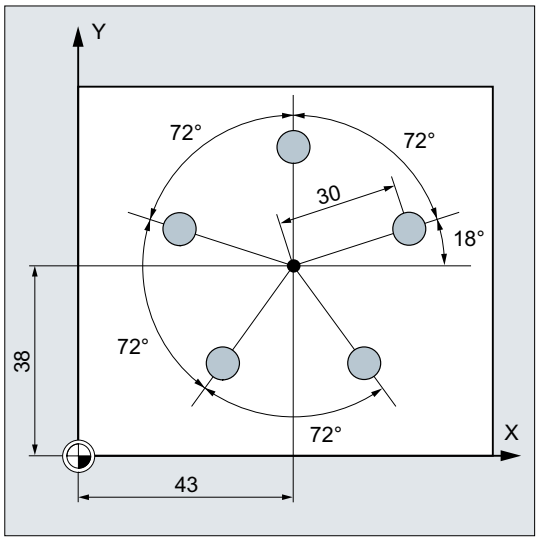
- Si des blocs CN contiennent des points de destination en coordonnées polaires, il n'est pas permis de programmer des coordonnées cartésiennes, telles que des paramètres d'interpolation, des adresses d'axes, etc. pour le plan de travail sélectionné.
- Si aucun pôle n'est défini avec `G110` ... `G112`, c'est l'origine du système de coordonnées pièce courant qui fait office de pôle.



- Rayon polaire RP = 0
  Le rayon polaire est calculé à partir de la distance entre le vecteur du point de départ dans le plan polaire et le vecteur polaire actif. Une fois calculé, le rayon polaire est enregistré de façon modale.
  Ceci est indépendant de la définition de pôle choisie (`G110` ... `G112`). Si les deux points sont programmés à l'identique, alors le rayon sera nul et l'alarme 14095 sera générée.
- Programmation de l'angle polaire AP uniquement
  Si le bloc actuel ne contient pas de rayon polaire RP, mais un angle polaire AP et s'il existe une différence entre la position actuelle et le pôle en coordonnées pièce, cette différence est utilisée comme rayon polaire et est enregistrée de façon modale. Si la différence est nulle, les coordonnées du pôle seront de nouveau prescrites et le rayon polaire modal restera nul.

Exemple

Réalisation d'un réseau de trous



les positions des trous sont fournies en coordonnées polaires.

Chaque trou est réalisé suivant la même procédure d'usinage :

Perçage d'avant-trous, perçage sur mesure, alésage à l'alésoir...

La gamme opératoire figure dans le sous-programme.

Code de programme	**Commentaire**
N10 G17 G54	; Plan de travail X/Y, origine pièce
N20 G111 X43 Y38	; Définition du pôle
N30 G0 RP=30 AP=18 Z5	; Accostage du point de départ, indication en coordonnées cylindriques
N40 L10	; Appel du sous-programme
N50 G91 AP=72	; Accostage de la position suivante en vitesse rapide, angle polaire en cotes relatives, le rayon polaire du bloc N30 reste mémorisé et peut être omis
N60 L10	; Appel du sous-programme
N70 AP=IC(72)	.
N80 L10	…
N90 AP=IC(72)	
N100 L10	…
N110 AP=IC(72)	
N120 L10	…
N130 G0 X300 Y200 Z100 M30	; Dégagement de l'outil, fin du programme

Voir aussi

Vue d'ensemble (Page 193)

2.9.4 Déplacements en rapide

2.9.4.1 Activation de la vitesse rapide (G0)

Le déplacement des axes d'interpolation en vitesse rapide est activé par l'instruction G G0.

Syntaxe

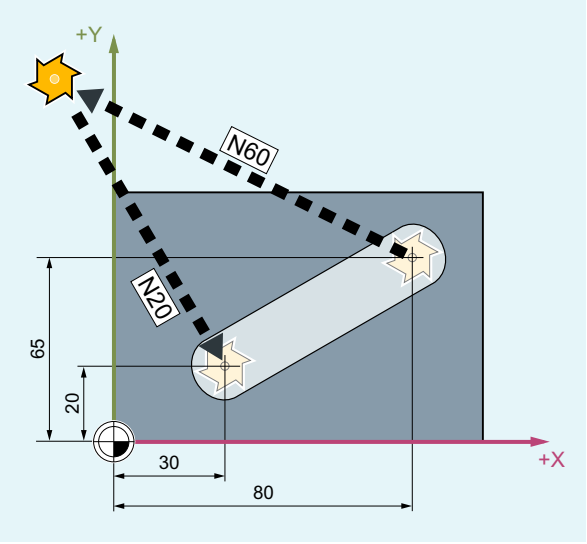
```
G0 X... Y... Z...
G0 RP=... AP=...
```

Signification

G0 :	Déplacement des axes à vitesse rapide	
	Prise d'effet :	modale
X... Y... Z... :	Indication du point final en coordonnées cartésiennes	
RP=... AP=... :	Indication du point final en coordonnées polaires	

Exemples

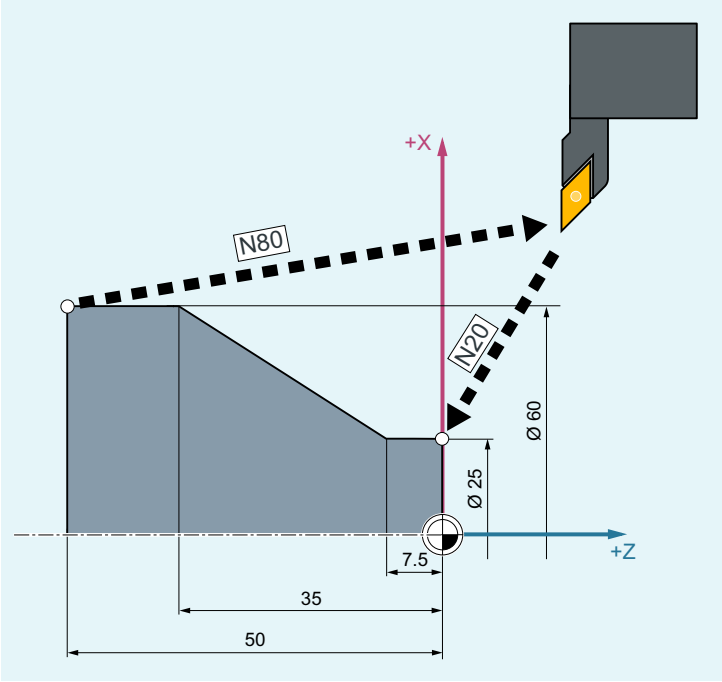
Exemple 1 : fraisage



Code de programme	Commentaire
N10 G90 S400 M3	; Introduction en cotes absolues, broche sens horaire
N20 G0 X30 Y20 Z2	; Accostage de la position de départ
N30 G1 Z-5 F1000	; Approche de l'outil
N40 X80 Y65	; Déplacement suivant une ligne droite
N50 G0 Z2	

Code de programme	Commentaire
N60 G0 X-20 Y100 Z100 M30	; Dégagement de l'outil, fin du programme

Exemple 2 : tournage



Code de programme	Commentaire
N10 G90 S400 M3	; Introduction en cotes absolues, broche sens horaire
N20 G0 X25 Z5	; Accostage de la position de départ
N30 G1 G94 Z0 F1000	; Approche de l'outil
N40 G95 Z-7.5 F0.2	
N50 X60 Z-35	; Déplacement suivant une ligne droite
N60 Z-50	
N70 G0 X62	
N80 G0 X80 Z20 M30	; Dégagement de l'outil, fin du programme

2.9.4.2 Activer/désactiver l'interpolation linéaire des déplacements de vitesse rapide (RTLION, RTLIOF)

Indépendamment des réglages par défaut (PM20730 $MC_G0_LINEAR_MODE), le comportement à l'interpolation lors des déplacements à vitesse rapide peut être réglé également dans le programme pièce à l'aide des instructions du groupe G 55.

Syntaxe

```
RTLIOF
```

```
...
RTLION
```

Signification

RTLIOF :	Instruction G pour désactiver l'interpolation linéaire ⇒ En vitesse rapide (G0), l'interpolation **non linéaire** est active. Tous les axes à interpolation atteignent leur point final indépendamment les uns des autres.	
	Prise d'effet :	modale
RTLION :	Instruction G pour activer l'interpolation linéaire ⇒ En vitesse rapide (G0), l'interpolation **linéaire** est active. Tous les axes à interpolation atteignent leur point final simultanément.	
	Prise d'effet :	modale

Remarque

Conditions pour RTLIOF

Pour forcer l'interpolation **non linéaire** avec RTLIOF, les conditions suivantes doivent être remplies :

- Aucune transformation (TRAORI, TRANSMIT, ...) active.
- G60 active (arrêt en fin de bloc).
- Aucun compresseur actif (COMPOF).
- Aucune correction du rayon d'outil active (G40).
- Aucune manivelle pour contour sélectionnée.
- Aucun grignotage actif.

Si l'une de ces conditions n'est pas remplie, l'interpolation linéaire est appliquée comme pour RTLION.

Exemple

Code de programme	**Commentaire**
	; Interpolation linéaire réglée par défaut :
	; MD20730 $MC_GO_LINEAR_MODE == TRUE
...	
N30 **RTLIOF**	; Désactiver l'interpolation linéaire.
N40 G0 X0 Y10	; Les blocs G0 sont exécutés avec interpolation non linéaire.
N50 G41 X20 Y20	; CRO active ⇒ Les blocs G0 sont exécutés avec interpolation linéaire.
N60 G40 X30 Y30	; CRO inactive ⇒ Les blocs G0 sont exécutés avec interpolation non linéaire.
N70 **RTLION**	; Activer l'interpolation linéaire.
...	

Informations complémentaires

Lecture du comportement à l'interpolation actuel

La variable système $AA_G0MODE permet de lire le comportement à l'interpolation actuel.

2.9.4.3 Adaptation des tolérances pour les déplacements en marche rapide (STOLF, CTOLG0, OTOLG0)

Les tolérances configurées via les paramètres machine (tolérances G0) peuvent être adaptées temporairement dans le programme pièce. Les réglages dans les paramètres machine ne sont pas modifiés. Les tolérances configurées sont à nouveau effectives après un reset de fin de programme ou de canal.

Conditions

Les tolérances G0 ne sont effectives que lorsque les conditions suivantes sont remplies :

- Une des fonctions suivantes est active :
  - Fonction compacteur COMP...
  - Fonction d'arrondissement G642 ou G645
  - Arrondissement d'orientation OST
  - Lissage d'orientation ORISON
  - Lissage pour orientation relative à la trajectoire ORIPATH
- Plusieurs blocs G0 (≥ 2) se suivent dans le programme pièce.
  Les tolérances G0 ne deviennent pas effectives pour un bloc G0 individuel étant donné que, lors de la transition d'un déplacement non G0 à un déplacement G0 (et inversement), c'est toujours la "**plus petite tolérance**" (tolérance d'usinage) qui agit !

Syntaxe

Adaptation de la tolérance G0 relative

```
STOLF=<valeur>
```

Adaptation des tolérances G0 absolues

```
CTOLG0=<valeur>
OTOLG0=<valeur>
```

Signification

STOLF :	Adresse pour la programmation d'un facteur de tolérance temporairement effectif pour les déplacements en marche rapide			
	<Value> :	Facteur de tolérance G0		
		Type :	REAL	
		Valeur :	≥ 0 :	Le facteur de tolérance G0 peut être aussi bien supérieur qu'inférieur à 1,0. Si le facteur est égal à 1,0 (valeur par défaut), les tolérances effectives pour les déplacements en marche rapide sont les mêmes que celles pour des déplacements non rapides. Normalement, le facteur de tolérance est réglé sur une valeur > 1,0. Le facteur de tolérance G0 programmé est effectif tant qu'il n'est pas écrasé par une nouvelle programmation STOLF ou déclenché par une programmation CTOLG0/OTOLG0, ou bien encore supprimé par une remise à zéro de fin de programme ou de canal.
			< 0 :	Suppression du facteur de tolérance programmé ⇒ Le facteur de tolérance prédéfini dans le paramètre machine redevient effectif.
CTOLG0 :	Adresse pour la programmation d'une tolérance de contour temporairement effective pour les déplacements en marche rapide			
	<Valeur> :	Valeur absolue pour la tolérance de contour		
		Type :	REAL	
		Valeur :	≥ 0 :	La valeur absolue programmée pour la tolérance de contour est effective tant qu'elle n'est pas écrasée par une nouvelle programmation CTOLG0 ou déclenchée par une programmation STOLF, ou bien encore supprimée par une remise à zéro de fin de programme ou de canal.
			< 0 :	Suppression de la valeur de tolérance programmée ⇒ La valeur de tolérance prédéfinie dans le paramètre machine redevient effective.
OTOLG0 :	Adresse pour la programmation d'une tolérance d'orientation temporairement effective pour les déplacements en marche rapide			
	<Valeur> :	Valeur absolue pour la tolérance d'orientation		
		Type :	REAL	
		Valeur :	≥ 0 :	La valeur absolue programmée pour la tolérance d'orientation est effective tant qu'elle n'est pas écrasée par une nouvelle programmation OTOLG0 ou déclenchée par une programmation STOLF, ou bien encore supprimée par une remise à zéro de fin de programme ou de canal.
			< 0 :	Suppression de la valeur de tolérance programmée ⇒ La valeur de tolérance prédéfinie dans le paramètre machine redevient effective.

Remarque

La dernière adresse programmée est toujours prioritaire, comme le montrent les exemples suivants :

- Dans le cadre de la programmation de CTOLG0 en cas de STOLF existant, la valeur de tolérance programmée avec CTOLG0 est utilisée pour l'arrondissement du contour.
- De la même manière, dans le cadre de la programmation de OTOLG0 en cas de STOLF existant, la valeur de tolérance programmée avec OTOLG0 est utilisée pour l'arrondissement de l'orientation.
- Après la reprogrammation de STOLF, le facteur de tolérance est de nouveau utilisé pour la tolérance de contour et d'orientation.

Exemples

Exemple 1 : Adaptation de la tolérance G0 relative

Code de programme	Commentaire
COMPCAD G645 G1 F10000	; Fonction compacteur COMPCAD
X... Y... Z...	; Les paramètres machine et données de réglage prennent effet à ce niveau.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
G0 X... Y... Z...	; Le paramètre machine $MC_G0_TOLERANCE_FACTOR (p. ex. =3), donc une tolérance d'arrondissement, prend effet à ce niveau pour une valeur de :
	$MC_G0_TOLERANCE_FACTOR * $MA_COMPRESS_POS_TOL
CTOL=0.02	
STOLF=4	
G1 X... Y... Z...	; À partir d'ici, une tolérance de contour de 0,02 mm prend effet.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
X... Y... Z...	; Un facteur de tolérance G0 de 4, donc une tolérance de contour de 0,08 mm, devient effectif à partir de ce niveau.
...	

Exemple 2 : Adaptation des tolérances G0 absolues

Les tolérances G0 absolues suivantes doivent être prédéfinies dans les paramètres machine :

- Tolérance de contour G0 : 0,1
- Tolérance d'orientation G0 : 1,0

Ces tolérances doivent être adaptées temporairement dans le programme pièce :

Code de programme	**Commentaire**
COMPCAD G645 G1 F10000	; Fonction compacteur COMPCAD
X... Y... Z...	; À partir d'ici, les tolérances d'usinage de pièce configurées sont effectives.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
G0 X... Y... Z...	; Ici, les tolérances G0 absolues configurées sont effectives.
CTOLG0=0.2 OTOLG0=2.0	; Programmation des tolérances G0 absolues.
G1 X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
X... Y... Z...	; À partir d'ici, les tolérances G0 programmées sont effectives.
...	

Informations complémentaires

Lecture du facteur de tolérance G0

Le facteur de tolérance actuellement effectif pour les déplacements en marche rapide peut être lu avec les variables système :

- Dans les actions synchrones ou avec arrêt du prétraitement des blocs dans le programme pièce, au moyen de la variable système :

$AC_STOLF	Facteur de tolérance G0 actif Facteur de tolérance G0 qui était effectif lors du traitement du bloc principal actuel.

- Sans arrêt du prétraitement des blocs dans le programme pièce, au moyen de la variable système :

$P_STOLF	Facteur de tolérance G0 programmé

Si aucune valeur n'est programmée avec STOLF dans le programme pièce actif, ces deux variables système fournissent la valeur configurée dans le paramètre machine.

Si aucune avance en rapide (G0) n'est active dans un bloc, ces variables système fournissent toujours la valeur 1.

Lecture des tolérances G0 absolues

Les tolérances absolues actuellement effectives pour les déplacements en marche rapide peuvent être lues avec des variables système :

- Dans les actions synchrones ou avec arrêt du prétraitement des blocs dans le programme pièce, au moyen des variables système :

$AC_CTOL_G0_ABS	Tolérance de contour active pour les déplacements G0 Tolérance de contour G0 qui était effective lors de la préparation du bloc principal actuel.
$AC_OTOL_ G0_ABS	Tolérance d'orientation active pour les déplacements G0 Tolérance d'orientation G0 qui était effective lors de la préparation du bloc principal actuel.

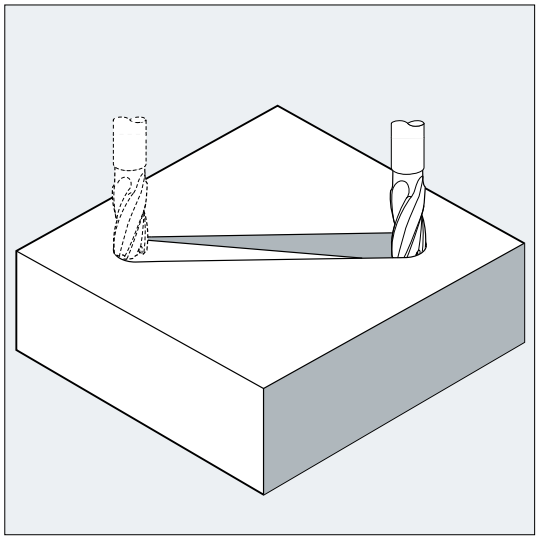
- Sans arrêt du prétraitement des blocs dans le programme pièce, au moyen des variables système :

$P_CTOL_ G0_ABS	Tolérance de contour programmée pour les déplacements G0
$P_OTOL_ G0_ABS	Tolérance d'orientation programmée pour les déplacements G0

Si aucune tolérance G0 absolue n'est programmée avec CTOLG0 et OTOLG0 dans le programme pièce actif, ces variables système fournissent les valeurs configurées dans les paramètres machine.

2.9.5 Interpolation linéaire (G1)

Avec `G1`, l'outil se déplace parallèlement aux axes, sur des droites inclinées dans le plan ou dans l'espace. L'interpolation linéaire permet de réaliser des surfaces 3 D, des rainures etc.



Syntaxe

```
G1 X... Y... Z ... F...
```

```
G1 AP=... RP=... F...
```

Signification

`G1` :	Interpolation linéaire (avec avance)
`X... Y... Z...`:	Point final en coordonnées cartésiennes
`AP=...` :	Point final en coordonnées polaires, ici angle polaire
`RP=...` :	Point final en coordonnées polaires, ici rayon polaire
`F...` :	Vitesse d'avance en mm/min L'outil se déplace avec l'avance F le long d'une droite, du point de départ courant jusqu'au point de destination programmé. Le point de destination est indiqué sous forme de coordonnées cartésiennes ou de coordonnées polaires. La pièce est usinée suivant cette trajectoire. Exemple : `G1 G94 X100 Y20 Z30 A40 F100` Le point final est accosté en X, Y, Z avec une avance de 100 mm/min ; l'axe rotatif A est déplacé en tant qu'axe synchrone de façon à ce que les quatre déplacements se terminent au même instant.

Remarque

`G1` est une fonction modale.

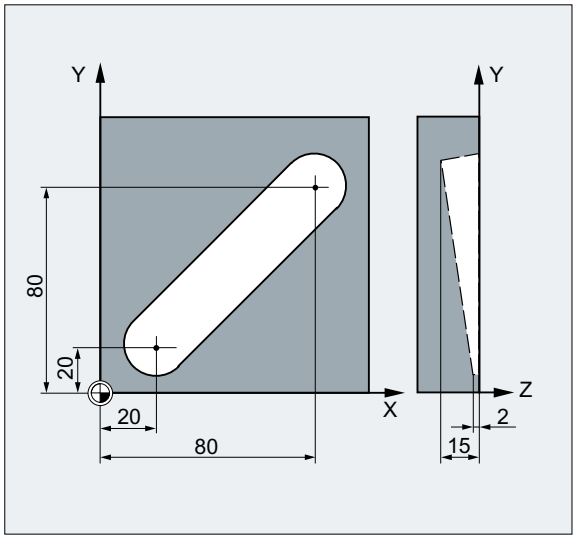
Pour l'usinage, il faut indiquer également la vitesse de rotation de la broche `S` et le sens de rotation `M3`/`M4`.

Avec `FGROUP`, vous pouvez définir des groupes d'axes pour lesquels s'applique l'avance tangentielle `F`. Vous trouverez plus d'informations à ce sujet au chapitre "Mode de déplacement".

Exemples

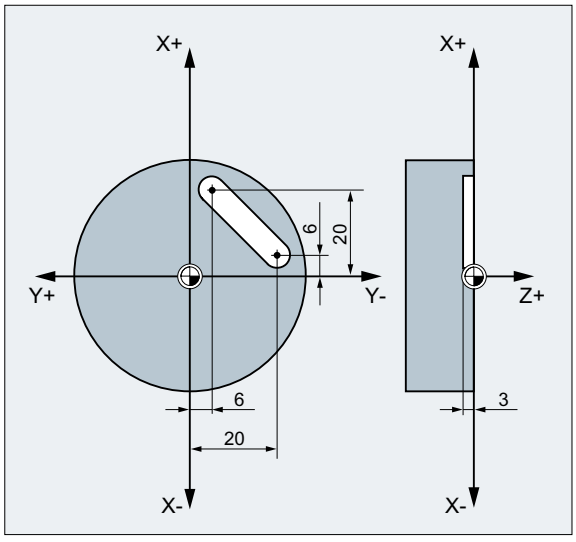
Exemple 1 : usinage d'une rainure (fraisage)

L'outil se déplace du point de départ vers le point final en direction X/Y. La pénétration en Z s'effectue simultanément.



Code de programme	Commentaire
N10 G17 S400 M3	; Sélection du plan de travail, broche sens horaire
N20 G0 X20 Y20 Z2	; Accostage de la position de départ
N30 G1 Z-2 F40	; Approche de l'outil
N40 X80 Y80 Z-15	; Déplacement suivant une droite oblique
N50 G0 Z100 M30	; Dégagement pour le changement d'outil

Exemple 2 : usinage d'une rainure (tournage)



Code de programme	Commentaire
N10 G17 S400 M3	; Sélection du plan de travail, broche sens horaire
N20 G0 X40 Y-6 Z2	; Accostage de la position de départ
N30 G1 Z-3 F40	; Approche de l'outil

Code de programme	Commentaire
N40 X12 Y-20	; Déplacement suivant une droite oblique
N50 G0 Z100 M30	; Dégagement pour le changement d'outil

2.9.6 Interpolation circulaire

2.9.6.1 Vue d'ensemble

L'interpolation circulaire permet de réaliser des cercles complets ou des arcs de cercle.

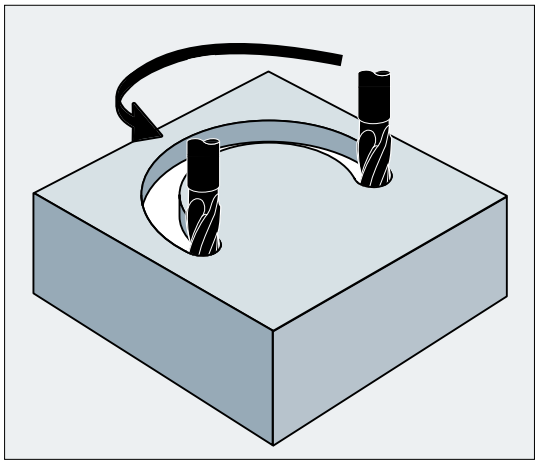


Figure 2-3 Exemple d'application : Fraisage d'une rainure circulaire

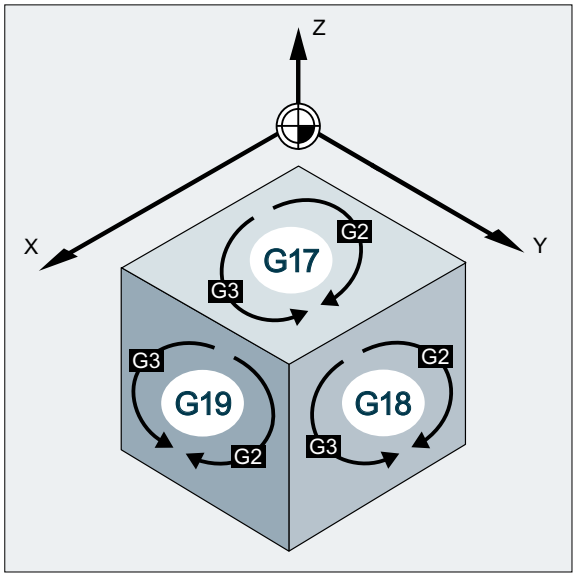
Variante de programmation :

La commande propose plusieurs possibilités pour programmer les déplacements circulaires. L'utilisateur peut ainsi transposer directement presque chaque type de cotation de dessin :

- Interpolation circulaire avec centre de cercle et point final (G2/G3, X... Y... Z..., I... J... K...) (Page 194)
- Interpolation circulaire avec rayon et point final (G2/G3, X... Y... Z..., CR) (Page 196)
- Interpolation circulaire avec angle au centre et point final / centre de cercle (G2/G3, X... Y... Z... / I... J... K..., AR) (Page 198)
- Interpolation circulaire avec coordonnées polaires (G2/G3, AP, RP) (Page 200)
- Interpolation circulaire avec point intermédiaire et point final (CIP, X... Y... Z..., I1... J1... K1...) (Page 202)
- Interpolation circulaire avec transition tangentielle (CT, X... Y... Z...) (Page 204)

Le plan pour l'Interpolation circulaire

Pour déterminer le sens du déplacement circulaire, dans le sens horaire avec G2 ou dans le sens antihoraire avec G3, la commande numérique doit disposer de l'indication du plan de travail (Page 149).



Exception :

Il est aussi possible de générer des cercles en dehors du plan de travail sélectionné (sauf si indication d'un angle au centre). Dans ce cas, ce sont les adresses d'axes indiquées par le programmateur comme point de fin de l'arc, qui déterminent le plan de travail.

2.9.6.2 Interpolation circulaire avec centre de cercle et point final (G2/G3, X... Y... Z..., I... J... K...)

Variante d'interpolation circulaire utilisant le **centre** et le **point final** d'un élément de contour circulaire pour l'interpolation.

Si vous programmez le cercle sans préciser de point final, vous générez un cercle complet.

Syntaxe

```
G2/G3 X... Y... Z... I... J... K...
G2/G3 X... Y... Z... I=AC(...) J=AC(...) K=(AC...)
```

Signification

G2 :	Interpolation circulaire sens horaire	
	Prise d'effet :	modale
G3 :	Interpolation circulaire en sens antihoraire	
	Prise d'effet :	modale

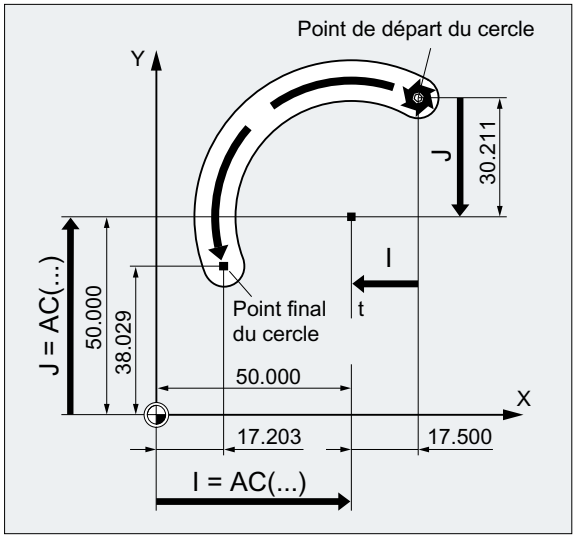
`X... Y... Z...:`	Point final du cercle en coordonnées cartésiennes En fonction du réglage actuel des cotes `G90`/`G91` ou `...=AC(...)` / `...=IC(...)`, les coordonnées du point final du cercle sont interprétées en cotation absolue ou en cotation relative.
`I... J... K...:`	Paramètres d'interpolation pour spécifier les coordonnées cartésiennes du centre du cercle dans les axes X, Y, Z Les coordonnées du centre du cercle sont spécifiées par défaut en cotes relatives, rapportées au point de départ du cercle. Si les coordonnées du centre du cercle sont spécifiées en cotation absolue par rapport à l'origine pièce, les paramètres d'interpolation I, J, K doivent être programmés comme suit : `I=AC(...) J=AC(...) K=AC(...)` **Remarque** Un paramètre d'interpolation de valeur "0" peut être omis, mais le second paramètre devra être impérativement indiqué.

Remarque

Le réglage par défaut `G90`/`G91`, cote absolue ou cote relative, est valable uniquement pour le point final du cercle.

Exemples

Exemple 1 : fraisage



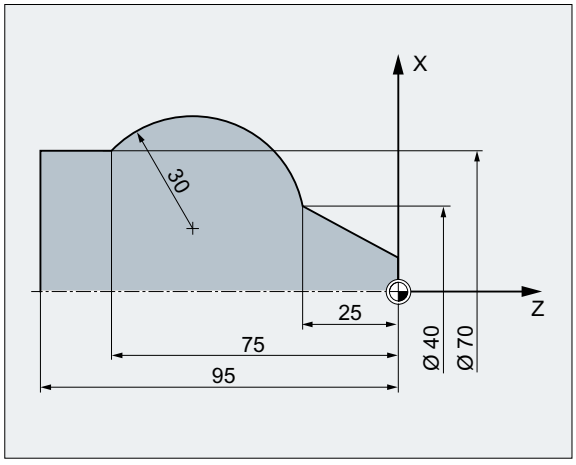
Introduction du centre du cercle en cotes relatives

```
N10 G0 X67.5 Y80.211
N20 G3 X17.203 Y38.029 I-17.5 J-30.211 F500
```

Introduction du centre du cercle en cotes absolues

```
N10 G0 X67.5 Y80.211
N20 G3 X17.203 Y38.029 I=AC(50) J=AC(50)
```

Exemple 2 : tournage



Introduction du centre du cercle en cotes relatives

```
N120 G0 X12 Z0
N125 G1 X40 Z-25 F0.2
N130 G3 X70 Z-75 I-3.335 K-29.25
N135 G1 Z-95
```

Introduction du centre du cercle en cotes absolues

```
N120 G0 X12 Z0
N125 G1 X40 Z-25 F0.2
N130 G3 X70 Z-75 I=AC(33.33) K=AC(-54.25)
N135 G1 Z-95
```

2.9.6.3 Interpolation circulaire avec rayon et point final (G2/G3, X... Y... Z..., CR)

Variante d'interpolation circulaire utilisant le **rayon** et le **point final** d'un élément de contour circulaire pour l'interpolation.

Remarque

Des cercles complets (angle de déplacement 360°) **ne peuvent pas** être programmés avec cette variante.

Syntaxe

```
G2/G3 X… Y… Z… CR=±...
```

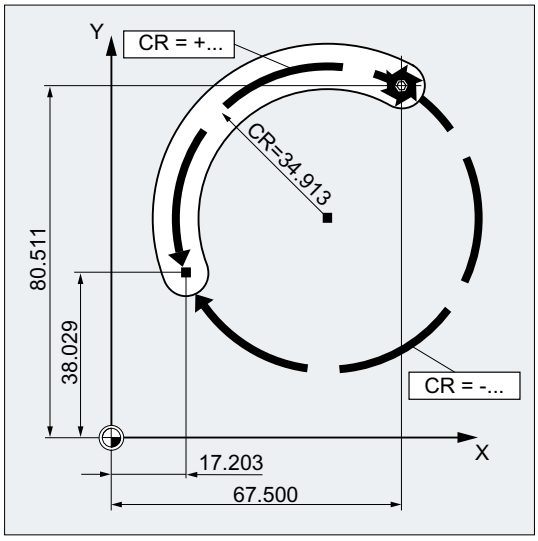
Signification

G2 :	Interpolation circulaire sens horaire	
	Prise d'effet :	modale
G3 :	Interpolation circulaire en sens antihoraire	
	Prise d'effet :	modale

`X... Y... Z...` :	Point final du cercle en coordonnées cartésiennes En fonction du réglage actuel des cotes `G90`/`G91` ou `...=AC(...)` / `...=IC(...)`, les coordonnées du point final sont interprétées en cotation absolue ou en cotation relative.	
`CR=±...` :	Rayon du cercle Le signe précise si l'angle de déplacement doit être supérieur ou inférieur à 180°. Le signe positif peut être omis.	
	`CR=+...` :	Angle de déplacement ≤ 180°
	`CR=-...` :	Angle de déplacement > 180°
	Remarque Aucune limitation pratique ne s'applique à la valeur maximale du rayon de cercle programmable.	

Exemples

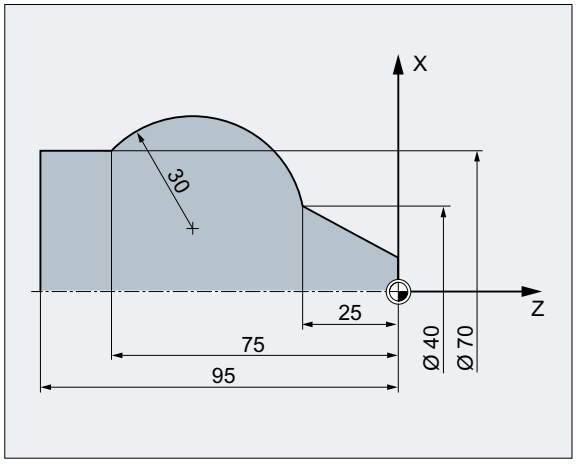
Exemple 1 : fraisage



Code de programme

```
N10 G0 X67.5 Y80.511
N20 G3 X17.203 Y38.029 CR=34.913 F500
...
```

Exemple 2 : tournage



Code de programme

```
...
N125 G1 X40 Z-25 F0.2
N130 G3 X70 Z-75 CR=30
N135 G1 Z-95
...
```

2.9.6.4 Interpolation circulaire avec angle au centre et point final / centre de cercle (G2/G3, X... Y... Z... / I... J... K..., AR)

Variante d'interpolation circulaire utilisant l'**angle au centre** et le **centre** ou le **point final** d'un élément de contour circulaire pour l'interpolation.

Remarque

Des cercles complets (angle de déplacement 360°) **ne peuvent pas** être programmés avec cette variante.

Syntaxe

```
G2/G3 X… Y… Z… AR=...
G2/G3 I… J… K… AR=...
```

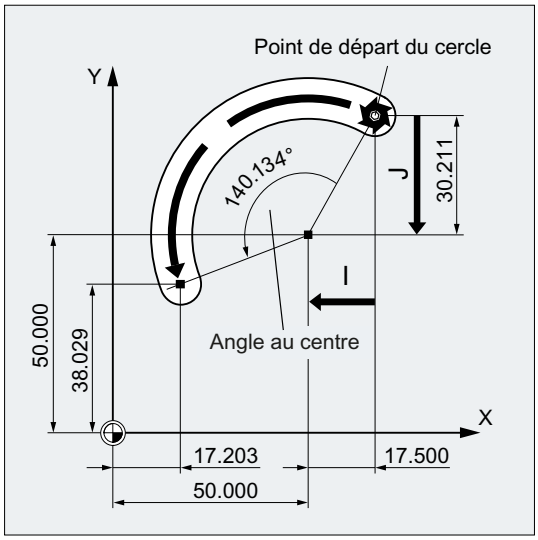
Signification

G2 :	Interpolation circulaire sens horaire	
	Prise d'effet :	modale
G3 :	Interpolation circulaire en sens antihoraire	
	Prise d'effet :	modale

`X... Y... Z...` :	Point final du cercle en coordonnées cartésiennes En fonction du réglage actuel des cotes `G90`/`G91` ou `...=AC(...)` / `...=IC(...)`, les coordonnées du point final du cercle sont interprétées en cotation absolue ou en cotation relative.	
`I... J... K...` :	Paramètres d'interpolation pour spécifier les coordonnées cartésiennes du centre du cercle dans les axes X, Y, Z Les coordonnées du centre du cercle sont spécifiées par défaut en cotes relatives, rapportées au point de départ du cercle. Si les coordonnées du centre du cercle sont spécifiées en cotation absolue par rapport à l'origine pièce, les paramètres d'interpolation I, J, K doivent être programmés comme suit : `I=AC(...) J=AC(...) K=AC(...)` **Remarque** Un paramètre d'interpolation de valeur "0" peut être omis, mais le second paramètre devra être impérativement indiqué.	
`AR=...` :	Angle au centre	
	Plage de valeurs :	0° ... 360°

Exemples

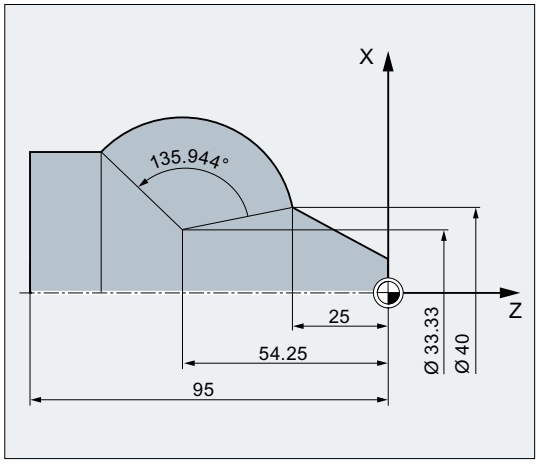
Exemple 1 : fraisage



Code de programme

```
N10 G0 X67.5 Y80.211
N20 G3 X17.203 Y38.029 AR=140.134 F500
N20 G3 I-17.5 J-30.211 AR=140.134 F500
```

Exemple 2 : tournage



Code de programme

```
N125 G1 X40 Z-25 F0.2
N130 G3 X70 Z-75 AR=135.944
N130 G3 I-3.335 K-29.25 AR=135.944
N130 G3 I=AC(33.33) K=AC(-54.25) AR=135.944
N135 G1 Z-95
```

2.9.6.5 Interpolation circulaire avec coordonnées polaires (G2/G3, AP, RP)

Variante d'interpolation circulaire utilisant le **le point final du cercle en coordonnées polaires** pour l'interpolation.

Dans ce cas, on applique la convention suivante :

- Le pôle se situe au centre du cercle.
- Le rayon polaire correspond au rayon du cercle.

Syntaxe

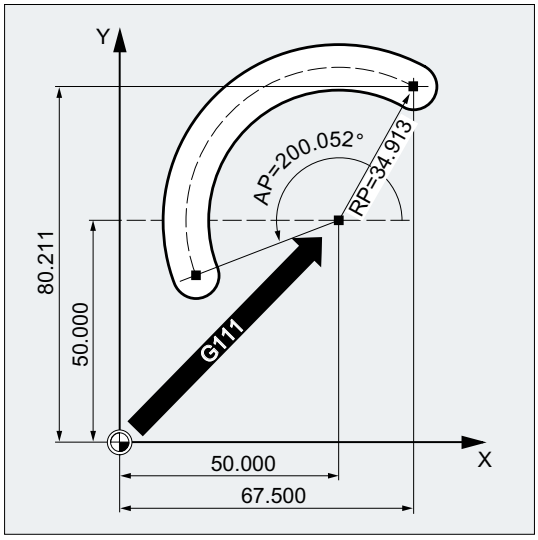
```
G2/G3 AP=... RP=...
```

Signification

G2 :	Interpolation circulaire sens horaire	
	Prise d'effet :	modale
G3 :	Interpolation circulaire en sens antihoraire	
	Prise d'effet :	modale
AP=... RP=... :	Point final du cercle en coordonnées polaires	
	AP=... :	Angle polaire
	RP=... :	Rayon polaire (≙ rayon du cercle)

Exemples

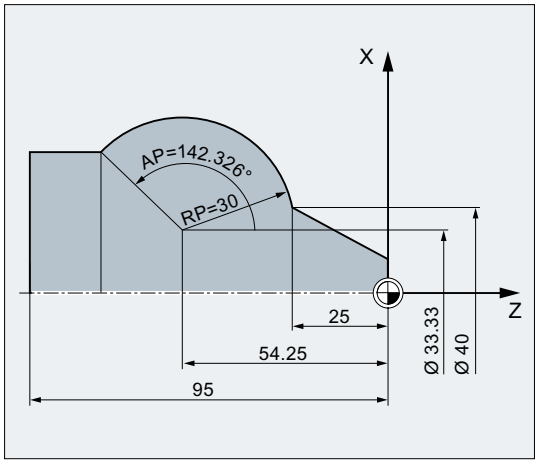
Exemple 1 : fraisage



Code de programme

```
N10 G0 X67.5 Y80.211
N20 G111 X50 Y50
N30 G3 RP=34.913 AP=200.052 F500
```

Exemple 2 : tournage



Code de programme

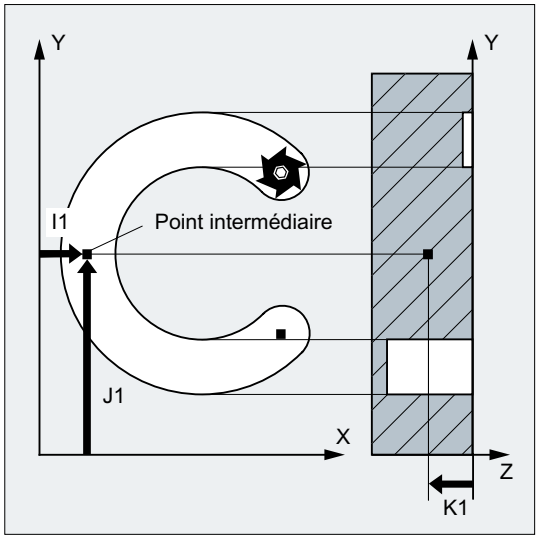
```
N125 G1 X40 Z-25 F0.2
N130 G111 X33.33 Z-54.25
N135 G3 RP=30 AP=142.326
N140 G1 Z-95
```

2.9.6.6 Interpolation circulaire avec point intermédiaire et point final (CIP, X... Y... Z..., I1... J1... K1...)

La variante d'interpolation circulaire programmée avec l'instruction G `CIP` permet l'interpolation d'arcs de cercle inclinés dans l'espace.

Le déplacement circulaire est décrit par le **point intermédiaire** et le **point final** du contour circulaire.

Le sens de déplacement est défini par l'ordre point de départ → point intermédiaire → point final.



Syntaxe

```
CIP X... Y... Z... I1=AC(...) J1=AC(...) K1=(AC...)
```

Signification

`CIP` :	Interpolation circulaire avec point intermédiaire	
	Prise d'effet :	modale
`X... Y... Z...` :	Point final du cercle en coordonnées cartésiennes En fonction du réglage actuel des cotes `G90`/`G91` ou `...=AC(...)` / `...=IC(...)`, les coordonnées du point final du cercle sont interprétées en cotation absolue ou en cotation relative.	
`I1... J1... K1...` :	Paramètres d'interpolation pour spécifier les coordonnées du point intermédiaire dans les axes X, Y, Z En fonction du réglage actuel des cotes `G90`/`G91` ou `...=AC(...)` / `...=IC(...)`, les coordonnées du point intermédiarie du cercle sont interprétées en cotation absolue ou en cotation relative. **Remarque** Un paramètre d'interpolation de valeur "0" peut être omis, mais le second paramètre devra être impérativement indiqué.	

Remarque

Le réglage par défaut `G90`/`G91` (cote absolue ou cote relative) est valable pour le point intermédiaire et le point final du cercle.

Lorsque la cotation relative est active `G91` ou `...=IC(...)`, le point intermédiaire et le point final ont comme référence le point de départ du cercle.

Remarque

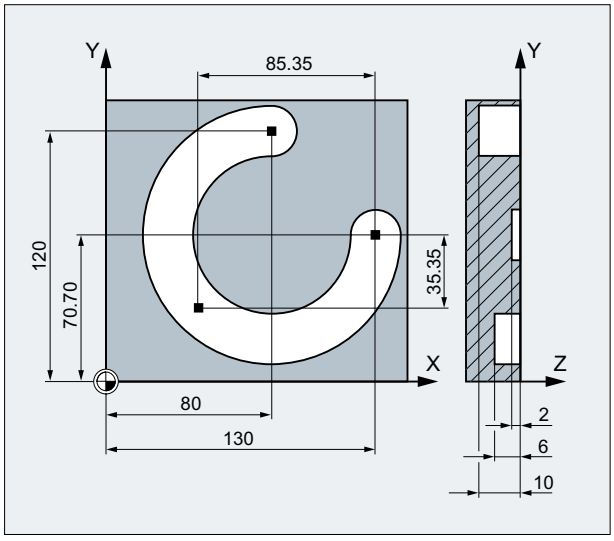
Technologie Tournage

La programmation du diamètre du paramètre d'interpolation de l'axe transversal n'est pas prise en charge par `CIP` lors de la programmation d'un cercle. Le paramètre d'interpolation de l'axe transversal doit, par conséquent, être programmé dans le **rayon**.

Exemples

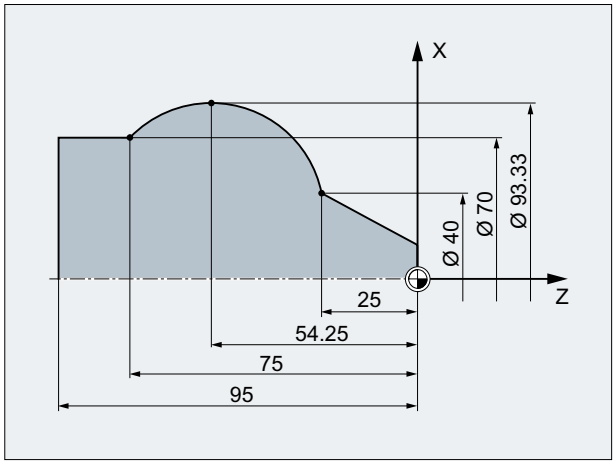
Exemple 1 : fraisage

Pour générer une rainure sur cercle inclinée dans l'espace, on décrit un cercle en définissant un point intermédiaire avec 3 paramètres d'interpolation et un point final avec ses 3 coordonnées.



Code de programme	Commentaire
`N10 G0 G90 X130 Y70.70 S800 M3`	`; Accostage du point de départ`
`N20 G17 G1 Z-2 F100`	`; Approche de l'outil`
`N30 CIP X80 Y120 Z-10 I1=IC(-85.35) J1=IC(-35.35) K1=-6`	`; Point final du cercle et point intermédiaire`
	`; Coordonnées dans les 3 axes géométriques`
`N40 M30`	`; Fin du programme`

Exemple 2 : tournage



Code de programme	Commentaire
...	
N125 G1 G90 X40 Z-25 F0.2	
N130 CIP X70 Z-75 **I1=IC(26.665)** K1=IC(-29.25)	; Paramètre d'interpolation I1 de l'axe transversal doit être programmé dans le rayon
; ou	
; N130 CIP X70 Z-75 **I1=46.665** K1=-54.25	
N135 G1 Z-95	

2.9.6.7 Interpolation circulaire avec transition tangentielle (CT, X... Y... Z...)

La variante d'interpolation circulaire programmée avec l'instruction G `CIP` permet l'interpolation d'arcs de cercle raccordés tangentiellement à l'élément de contour programmé précédemment.

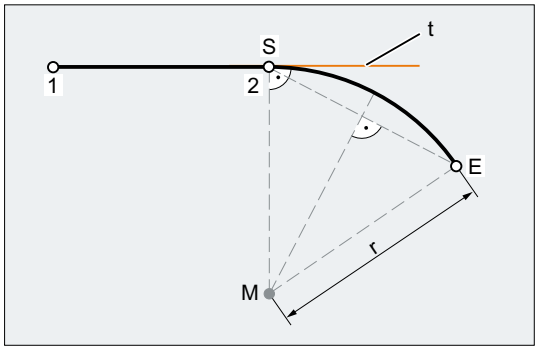
Dans ce cas, le cercle est défini par le **point de départ**, le **point final** et la **direction tangentielle au point de départ**.

Remarque

Direction tangentielle au point de départ

La direction tangentielle au point de départ d'un bloc CT est déterminée par la tangente au point de fin du contour programmé dans le bloc précédent contenant un déplacement.

Entre ce bloc et le bloc courant, on peut avoir un nombre quelconque d'autres blocs ne contenant cependant aucune information de déplacement.



S Point de départ
E Point final
M Centre du cercle
r Rayon du cercle
t Tangente au point final du contour programmé dans le bloc précédent contenant un déplacement.

Figure 2-4 Trajectoire circulaire S-E raccordée tangentiellement à la droite 1-2

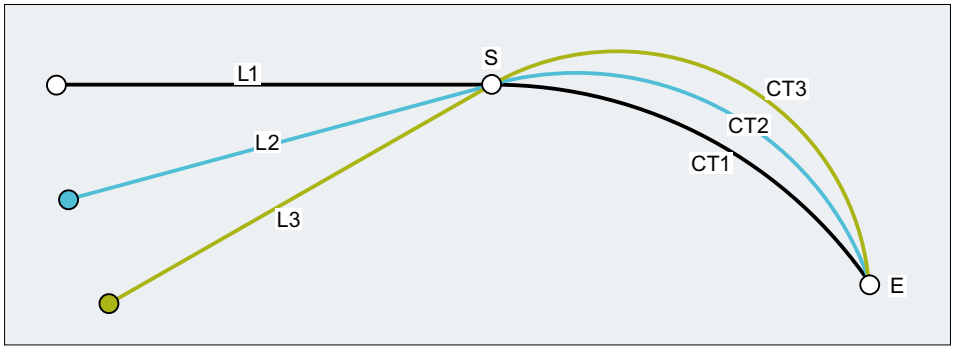


Figure 2-5 Les trajectoires circulaires raccordées tangentiellement dépendent de l'élément de contour précédent.

Syntaxe

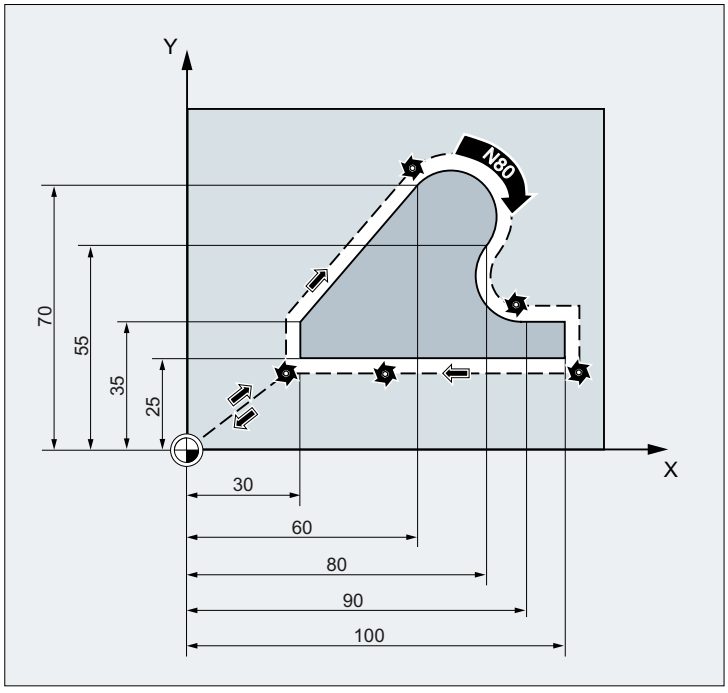
```
CT X... Y... Z...
```

Signification

`CT` :	Interpolation circulaire avec transition tangentielle	
	Prise d'effet :	modale
`X... Y... Z...` :	Point final du cercle en coordonnées cartésiennes En fonction du réglage actuel des cotes `G90`/`G91` ou `...=AC(...)` / `...=IC(...)`, les coordonnées du point final du cercle sont interprétées en cotation absolue ou en cotation relative.	

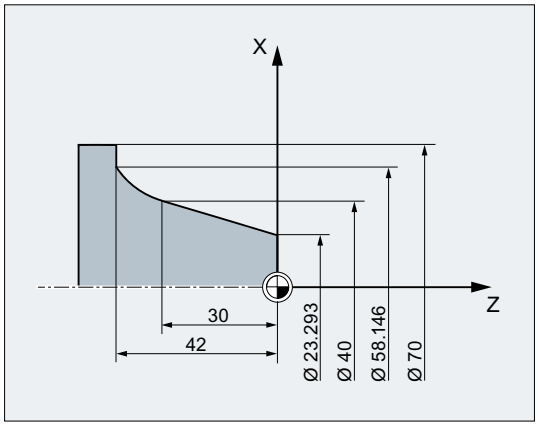
Exemples

Exemple 1 : fraisage



Code de programme	Commentaire
N10 G0 Z100	
N20 G17 T1 M6	
N30 G0 X0 Y0 Z2 M3 S300 D1	
N40 Z-5 F1000	; Prise de passe de l'outil
N50 G41 X30 Y25 G1 F1000	; Activation de la correction du rayon d'outil
N60 Y35	; Fraisage du contour
N70 X60 Y70	
N80 CT X80 Y55	**; Programmation d'un cercle avec transition tangentielle**
N90 X90 Y35	
N100 G1 X100	
N110 Y25	
N120 X30	
N130 G0 G40 X0 Y0	; Désactivation de la correction du rayon d'outil
N140 Z100	; Dégagement de l'outil
N140 M30	

Exemple 2 : tournage




```
Code de programme              Commentaire
...
N110 G1 X23.293 Z0 F10
N115 X40 Z-30 F0.2
N120 CT X58.146 Z-42           ; Programmation d'un cercle avec transition tangentielle
N125 G1 X70
...
```

Informations complémentaires

Splines

Dans le cas des courbes de type spline, la direction tangentielle est déterminée par la droite qui relie les deux derniers points. En général, sur une courbe de type spline A ou C quand ENAT ou EAUTO est activé, cette direction tangentielle n'est pas identique à la direction au point final de la courbe spline.

La transition entre des courbes de type spline B est toujours tangentielle, sachant que la direction tangentielle est définie comme pour les courbes spline A ou C avec ETAN activé.

Changement de frame

Lorsqu'un changement de frame est effectué entre le bloc définissant la tangente et le bloc CT, la tangente est assujettie à ce changement.

Cas limite

Lorsque le prolongement de la tangente au point de départ passe par le point final, ce n'est pas un cercle qui est généré, mais une droite (cas limite d'un cercle avec un rayon infini). Dans ce cas particulier, il ne faut pas programmer TURN ou il faut lui donner une valeur nulle TURN=0.

Remarque

Quand on s'approche de ce cas limite, les cercles sont générés avec un rayon arbitraire et quand TURN a été programmé avec une valeur différente de 0, l'usinage est généralement interrompu par une alarme causée par la violation des limites du logiciel.

Position du plan du cercle

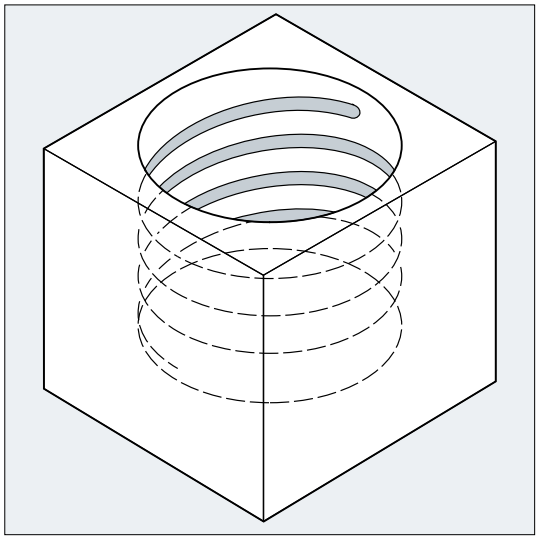
La position du plan du cercle dépend du plan activé (G17-G19).

Lorsque la tangente du bloc précédent n'est pas située dans le plan activé, c'est sa projection dans le plan activé qui est utilisée.

Si le point de départ et le point final n'ont pas la même composante de position perpendiculaire au plan activé, le programme ne génère pas un cercle, mais une hélice.

2.9.7 Interpolation hélicoïdale (G2/G3, TURN)

L'interpolation hélicoïdale permet par exemple de réaliser des filetages ou des rainures de graissage.



Dans une interpolation hélicoïdale, on superpose deux mouvements exécutés en parallèle :

- un déplacement circulaire dans un plan auquel
- se superpose un déplacement linéaire qui lui est perpendiculaire.

Syntaxe

```
G2/G3 X… Y… Z… I… J… K… TURN=
G2/G3 X… Y… Z… I… J… K… TURN=
G2/G3 AR=… I… J… K… TURN=
G2/G3 AR=… X… Y… Z… TURN=
G2/G3 AP… RP=… TURN=
```

Signification

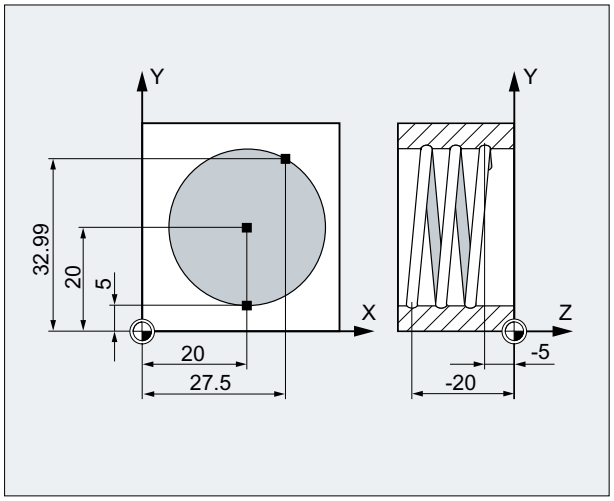
`G2` :	Déplacement sur une trajectoire circulaire dans le sens horaire
`G3` :	Déplacement sur une trajectoire circulaire dans le sens antihoraire
`X Y Z` :	Point final en coordonnées cartésiennes

I J K :	Centre du cercle en coordonnées cartésiennes
AR :	Angle au centre
TURN= :	Nombre de tours supplémentaires entre 0 et 999
AP= :	Angle polaire
RP= :	Rayon polaire

Remarque

G2 et G3 sont des fonctions modales.

Exemple



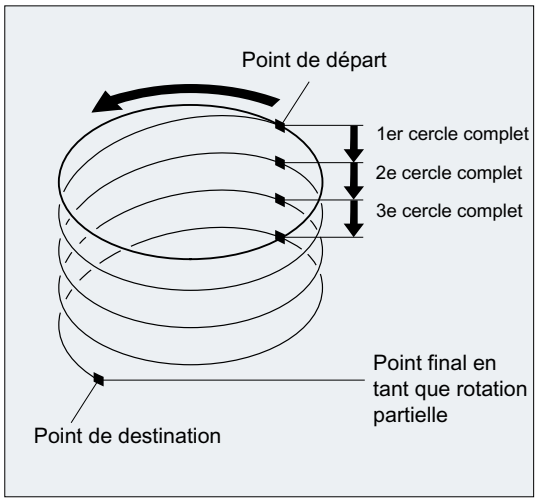
Code de programme	Commentaire
N10 G17 G0 X27.5 Y32.99 Z3	; Accostage de la position de départ
N20 G1 Z-5 F50	; Approche de l'outil
N30 G3 X20 Y5 Z-20 I=AC(20) J=AC(20) TURN=2	; Hélice dont les données sont les suivantes : à partir de la position de départ, exécuter 2 cercles complets, puis accoster le point final.
N40 M30	; Fin du programme

Informations complémentaires

Séquence de déplacements

1. Accostage du point de départ
2. Avec TURN=, exécution des cercles complets programmés.
3. Accostage du point final du cercle, par ex. rotation partielle.
4. Exécution des pts 2 et 3 en parallèle au déplacement linéaire

Le nombre de tours complets et le point final programmé - combinés au déplacement linéaire - donnent le pas avec lequel l'hélice doit être usinée.



Programmation du point final d'une interpolation hélicoïdale

Pour une explication détaillée des paramètres d'interpolation, reportez-vous à l'interpolation circulaire.

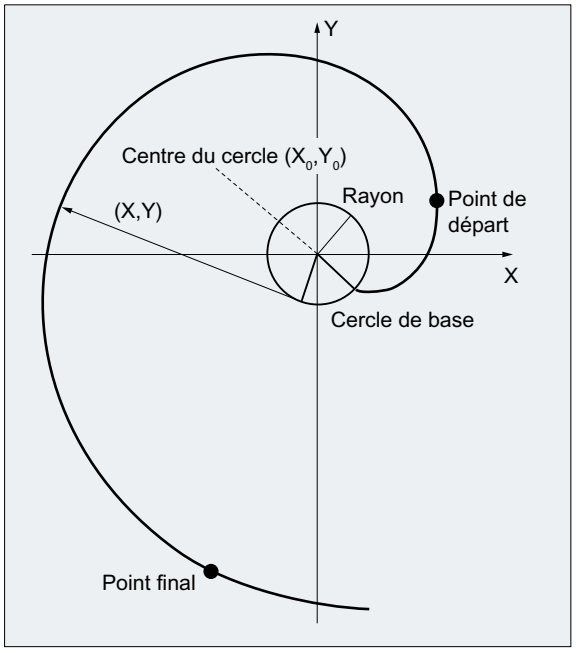
Avance programmée

Avec une interpolation hélicoïdale, il convient de programmer une correction de l'avance (`CFC`). Avec `FGROUP`, vous pouvez définir les axes qu'il s'agit de déplacer avec une avance programmée. Pour plus d'informations voir chapitre Mode de déplacement.

2.9.8 Interpolation de développante (INVCW, INVCCW)

La développante du cercle est une courbe décrite par l'extrémité d'un fil maintenu tendu et enroulé autour du cercle.

Cette forme d'interpolation permet de produire une trajectoire le long d'une développante. Elle est effectuée dans le plan dans lequel le cercle de base a été défini et va du point de départ au point final programmés.



Le point final peut être programmé de deux manières :

1. Programmation directe en coordonnées cartésiennes
2. Programmation indirecte par indication de l'angle au centre (voir également à ce sujet la programmation de l'angle au centre dans la programmation circulaire).

Si le point de départ et le point final ne sont pas situés dans le plan du cercle de base, il se forme une superposition qui génère une courbe dans l'espace de manière analogue à l'interpolation hélicoïdale des cercles.

En précisant des trajets perpendiculaires au plan actif (de façon comparable à l'interpolation hélicoïdale dans le cas des cercles), il est possible de réaliser un déplacement sur une développante dans l'espace.

Syntaxe

```
INVCW X... Y... Z... I... J... K... CR=...
INVCCW X... Y... Z... I... J... K... CR=...
INVCW I... J... K... CR=... AR=...
INVCCW I... J... K... CR=... AR=...
```

Signification

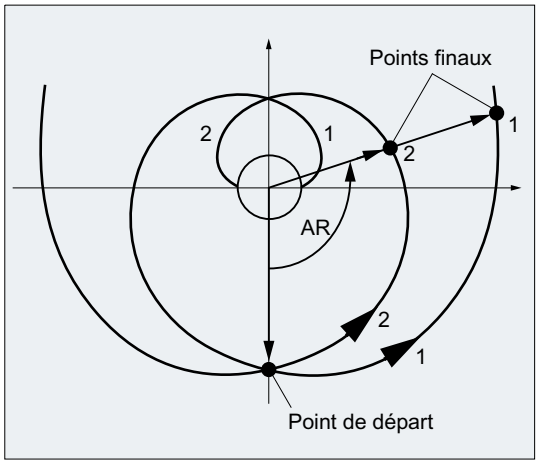
`INVCW` :	Instruction de déplacement sur une développante dans le sens horaire
`INVCCW` :	Instruction de déplacement sur une développante dans le sens antihoraire
`X... Y... Z...` :	Programmation directe du point final en coordonnées cartésiennes

`I... J... K...` :	Paramètres d'interpolation pour la définition du centre du cercle de base en coordonnées cartésiennes **Remarque :** Les coordonnées indiquées se rapportent au point de départ de la développante.	
`CR=...` :	Rayon du cercle de base	
`AR=...` :	Programmation indirecte du point final par indication de l'angle au centre (angle de rotation) L'origine de l'angle au centre est la droite qui relie le centre du cercle au point de départ.	
	`AR` > 0 :	Sur la développante, la trajectoire **s'éloigne du cercle de base**.
	`AR` < 0 :	Sur la développante, la trajectoire **se dirige vers cercle de base**. Dans le cas de AR < 0, l'angle de rotation maximal est limité du fait que le point final doit toujours se trouver en dehors du cercle de base.

Programmation indirecte du point final par indication de l'angle au centre

IMPORTANT
Angle au centre non défini La programmation indirecte du point final par indication de l'angle au centre AR doit tenir compte du signe de l'angle, car une inversion de signe conduirait à une autre développante et donc à une autre trajectoire.

L'exemple suivant illustre ce cas :

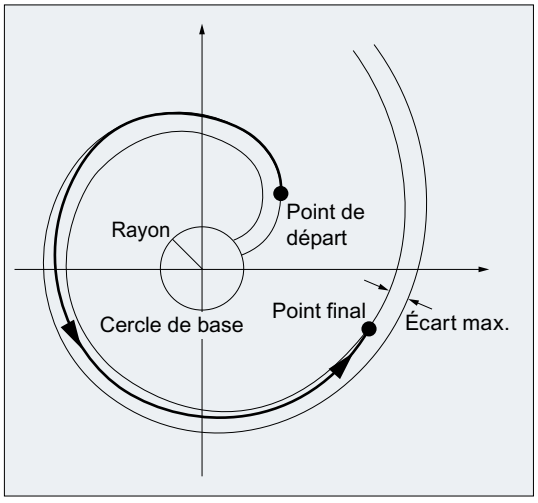


Pour les développantes 1 et 2, le rayon et le centre indiqués pour le cercle de base ainsi que le point de départ et le sens de rotation (`INVCW` / `INVCCW`) concordent. La seule différence est le signe de l'angle au centre :

- Lorsque AR > 0, la trajectoire suit la développante 1 jusqu'au point final 1.
- Lorsque AR < 0, la trajectoire suit la développante 2 jusqu'au point final 2.

Autres conditions à prendre en compte

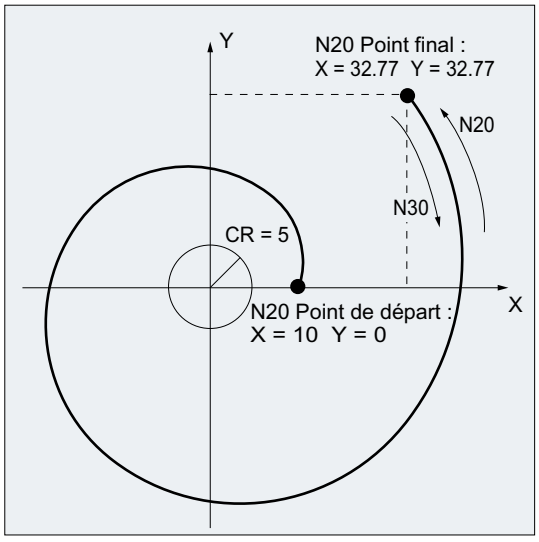
- Le point de départ et le point final doivent tous deux figurer en dehors du plan dans lequel se trouve le cercle de base de la développante (cercle avec rayon CR autour du centre défini par I, J, K). Si cette condition n'est pas réalisée, une alarme est générée et le programme est interrompu.
- Les deux modes de programmation du point final (programmation directe en coordonnées cartésiennes et programmation indirecte par indication de l'angle au centre) s'excluent mutuellement. Un bloc ne peut donc contenir que l'un des deux modes de programmation.
- Lorsque le point final programmé n'est pas situé exactement sur la développante définie par le point de départ et le cercle de base, une interpolation est effectuée entre les deux développantes définies par le point de départ et le cercle de base (voir la figure ci-dessous).



L'écart maximal du point final est défini par un paramètre machine (→ constructeur de machines). Lorsque l'écart, en direction radiale, du point final programmé est supérieur à la valeur fixée dans le paramètre machine MD, une alarme est générée et le programme est interrompu.

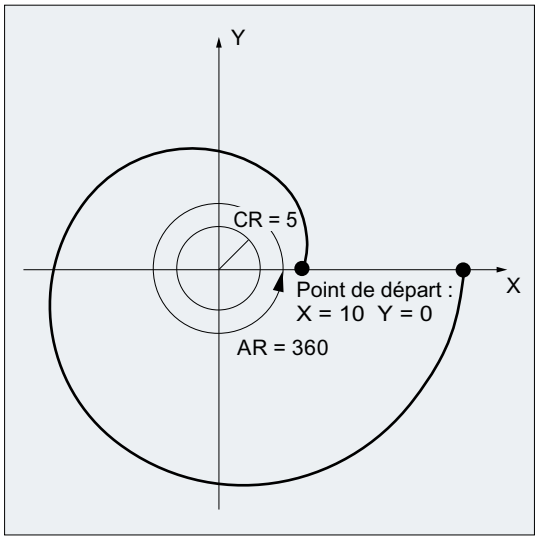
Exemples

Exemple 1 : développante dans le sens antihoraire du point de départ au point final programmé et retour en tant que développante dans le sens horaire



Code de programme	Commentaire
N10 G1 X10 Y0 F5000	; Accostage de la position de départ
N15 G17	; Sélection du plan X/Y en tant que plan de travail
N20 INVCCW X32.77 Y32.77 CR=5 I-10 J0	; Développante dans le sens antihoraire, point final en coordonnées cartésiennes
N30 INVCW X10 Y0 CR=5 I-32.77 J-32.77	; Développante dans le sens horaire, point de départ identique au point final de N20, nouveau point final identique au point de départ de N20, le nouveau centre du cercle se rapporte au nouveau point de départ et est identique à l'ancien centre du cercle.
...	

Exemple 2 : développante dans le sens antihoraire avec programmation indirecte du point final par indication de l'angle au centre



Code de programme	Commentaire
`N10 G1 X10 Y0 F5000`	`; Accostage de la position de départ`
`N15 G17`	`; Sélection du plan X/Y en tant que plan de travail`
`N20 INVCCW CR=5 I-10 J0 AR=360`	`; Développante s'éloignant du cercle de base (indication d'un angle positif) dans le sens antihoraire avec une rotation complète (360 degrés)`
`...`	

2.9.9 Eléments de contour

2.9.9.1 Programmation d'éléments de contour

Fonction

La programmation d'éléments de contour sert à programmer rapidement des contours simples.

Il est possible de programmer des éléments de contour avec 1, 2, 3 points et plus, avec des chanfreins ou des arrondis comme éléments de transition, au moyen de la définition de coordonnées cartésiennes et/ou d'angles (ANG ou ANG1 et ANG2).

Dans les blocs qui décrivent les éléments de contour, on peut utiliser d'autres adresses CN telles que des lettres d'adresse d'autres axes (axes individuels ou axes perpendiculaires au plan d'usinage), des indications de fonction auxiliaire, des instructions G, des vitesses, etc.

Remarque

Calculateur de contours

La programmation d'éléments de contour est également réalisable simplement au moyen du calculateur de contours. Il s'agit d'un outil de l'interface utilisateur pour la programmation et la représentation graphique de contours de pièce simples ou complexes. Les contours programmés avec le calculateur de contours sont enregistrés dans le programme pièce.

Informations complémentaires : Manuel d'utilisation

Paramétrage

Les descripteurs d'angle, de rayon et de chanfrein sont définis par des paramètres machine.

MD10652 $MN_CONTOUR_DEF_ANGLE_NAME (nom de l'angle pour les éléments de contour)

MD10654 $MN_RADIUS_NAME (nom du rayon pour les éléments de contour)

MD10656 $MN_CHAMFER_NAME (nom du chanfrein pour les éléments de contour)

Remarque

Voir les indications du constructeur de machines.

2.9.9.2 Eléments de contour : une droite

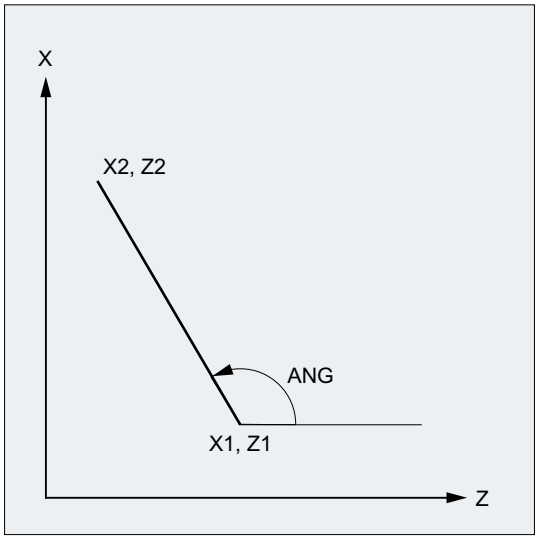
Remarque

Dans la description ci-dessous on suppose que :

- G18 est activé (⇒ le plan de travail actif est le plan Z/X).
  (La programmation d'éléments de contour est cependant possible également avec G17 ou G19 sans restriction.)
- Les descripteurs suivants sont définis pour l'angle, le rayon et le chanfrein :
  – ANG (angle)
  – RND (rayon)
  – CHR (chanfrein)

Le point final de la droite est défini par les indications suivantes :

- Angle ANG
- **Une** coordonnée cartésienne du point final (X2 ou Z2)



ANG : Angle de la droite
X1, Z1 : Coordonnées du point de départ
X2, Z2 : Coordonnées du point final de la droite

Syntaxe

```
X... ANG=...
Z... ANG=...
```

Signification

`X...` :	Coordonnée X du point final
`Z...` :	Coordonnée Z du point final
`ANG` :	Descripteur pour la programmation de l'angle La valeur indiquée (angle) se rapporte à l'abscisse du plan de travail actif (axe Z dans le cas de `G18`).

Exemple

Code de programme	**Commentaire**
`N10 X5 Z70 F1000 G18`	`; Accostage de la position de départ`
`N20 X88.8 ANG=110`	`; Droite avec indication d'angle`
`N30 ...`	

Ou :

Code de programme	**Commentaire**
`N10 X5 Z70 F1000 G18`	`; Accostage de la position de départ`
`N20 Z39.5 ANG=110`	`; Droite avec indication d'angle`
`N30 ...`	

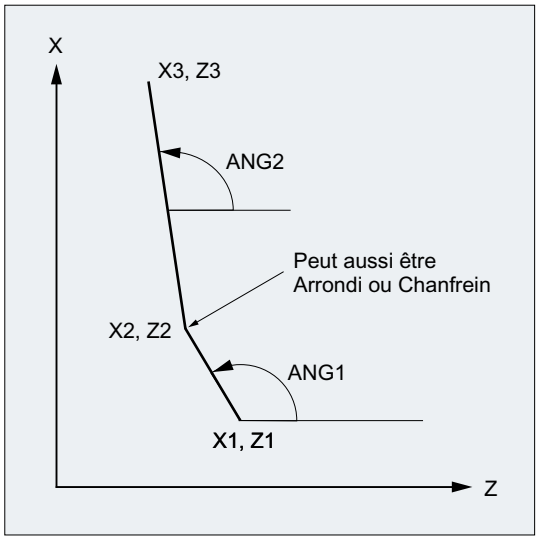
2.9.9.3 Eléments de contour : deux droites

Remarque

Dans la description ci-dessous on suppose que :

- G18 est activé (⇒ le plan de travail actif est le plan Z/X).
  (La programmation d'éléments de contour est cependant possible également avec G17 ou G19 sans restriction.)
- Les descripteurs suivants sont définis pour l'angle, le rayon et le chanfrein :
  – ANG (angle)
  – RND (rayon)
  – CHR (chanfrein)

Le point final de la première droite peut être programmé par indication des coordonnées cartésiennes ou par indication des angles des deux droites. Le point final de la deuxième droite doit toujours être programmé en coordonnées cartésiennes. Le point d'intersection des deux droites peut être réalisé sous forme d'angle, d'arrondi/congé ou de chanfrein.



ANG1 :	Angle de la première droite
ANG2 :	Angle de la seconde droite
X1, Z1 :	Coordonnées du point de départ de la première droite
X2, Z2 :	Coordonnées du point final de la première droite et du point de départ de la deuxième droite
X3, Z3 :	Coordonnées du point final de la deuxième droite

Syntaxe

Programmation du point final de la première droite par indication des angles

- Angle de transition entre les droites :

```
ANG=…
X… Z… ANG=…
```

- Arrondi/congé de transition entre les droites :

```
ANG=… RND=...
X… Z… ANG=…
```

- Chanfrein de transition entre les droites :

```
ANG=… CHR=...
X… Z… ANG=…
```

Programmation du point final de la première droite par indication des coordonnées

- Angle de transition entre les droites :

```
X… Z…
X… Z…
```

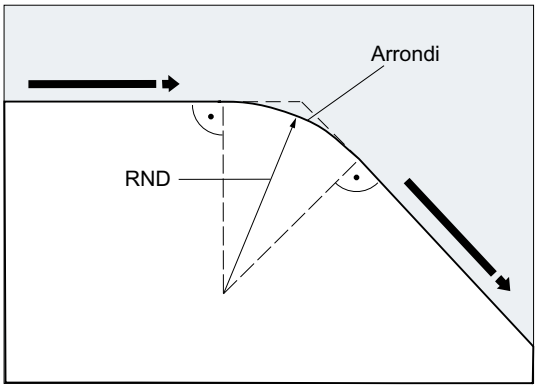
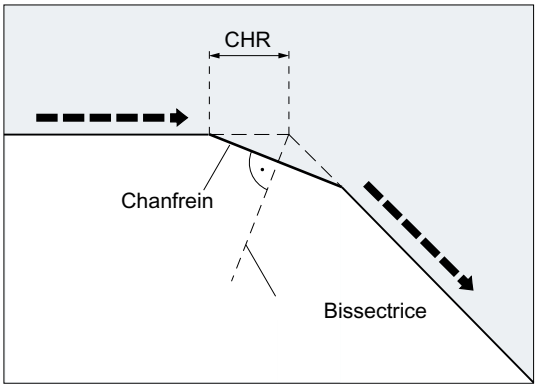
- Arrondi/congé de transition entre les droites :

```
X… Z… RND=...
X… Z…
```

- Chanfrein de transition entre les droites :

```
X… Z… CHR=...
X… Z…
```

Signification

`ANG=...` :	Descripteur pour la programmation de l'angle La valeur indiquée (angle) se rapporte à l'abscisse du plan de travail actif (axe Z dans le cas de `G18`).
`RND=...` :	Descripteur pour la programmation d'un arrondi/congé La valeur indiquée correspond au rayon de l'arrondi / du congé : 
`CHR=...` :	Descripteur pour la programmation d'un chanfrein La valeur indiquée correspond à la largeur du chanfrein dans le sens de déplacement : 
`X...` :	Coordonnées X
`Z...` :	Coordonnées Z

Remarque

Pour plus d'informations sur la programmation d'un chanfrein ou d'un arrondi/congé, voir "Chanfrein, arrondissement (CHF, CHR, RND, RNDM, FRC, FRCM) (Page 250)".

Exemple

```
Code de programme            Commentaire
N10 X10 Z80 F1000 G18        ; Accostage de la position de départ
N20 ANG=148.65 CHR=5.5       ; Droite avec indication d'angle et de chanfrein
N30 X85 Z40 ANG=100          ; Droite avec indication d'angle et de point final
N40 ...
```

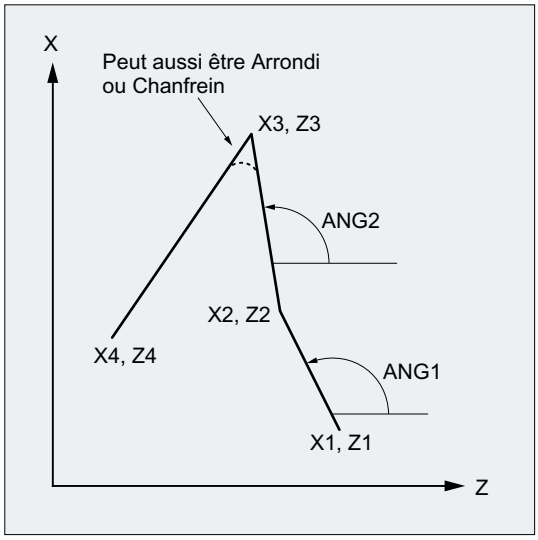
2.9.9.4 Eléments de contour : trois droites

Remarque

Dans la description ci-dessous on suppose que :

- G18 est activé (⇒ le plan de travail actif est le plan Z/X).
  (La programmation d'éléments de contour est cependant possible également avec G17 ou G19 sans restriction.)
- Les descripteurs suivants sont définis pour l'angle, le rayon et le chanfrein :
  - ANG (angle)
  - RND (rayon)
  - CHR (chanfrein)

Le point final de la première droite peut être programmé par indication des coordonnées cartésiennes ou par indication des angles des deux droites. Le point final de la deuxième et de la troisième droite doit toujours être programmé en coordonnées cartésiennes. Le point d'intersection des droites peut être réalisé sous forme d'angle, d'arrondi/congé ou de chanfrein.



ANG1 :	Angle de la première droite
ANG2 :	Angle de la seconde droite
X1, Z1 :	Coordonnées du point de départ de la première droite
X2, Z2 :	Coordonnées du point final de la première droite et du point de départ de la deuxième droite
X3, Z3 :	Coordonnées du point final de la deuxième droite et du point de départ de la troisième droite
X4, Z4 :	Coordonnées du point final de la troisième droite

Remarque

La programmation décrite ici pour un élément de contour à 3 points peut être poursuivie pour plus de trois points.

Syntaxe

Programmation du point final de la première droite par indication des angles

- Angle de transition entre les droites :

```
ANG=…
X… Z… ANG=…
X… Z…
```

- Arrondi/congé de transition entre les droites :

```
ANG=… RND=...
X… Z… ANG=… RND=...
X… Z…
```

- Chanfrein de transition entre les droites :

```
ANG=… CHR=...
X… Z… ANG=… CHR=...
X… Z…
```

Programmation du point final de la première droite par indication des coordonnées

- Angle de transition entre les droites :

```
X… Z…
X… Z…
X… Z…
```

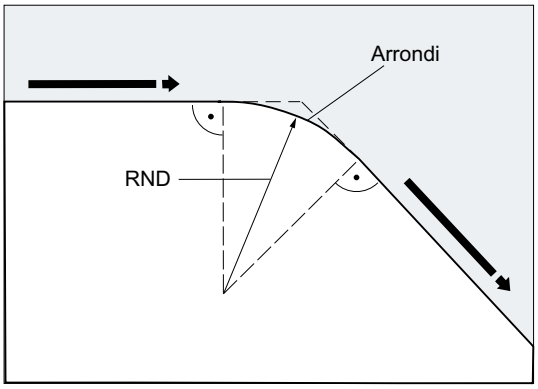
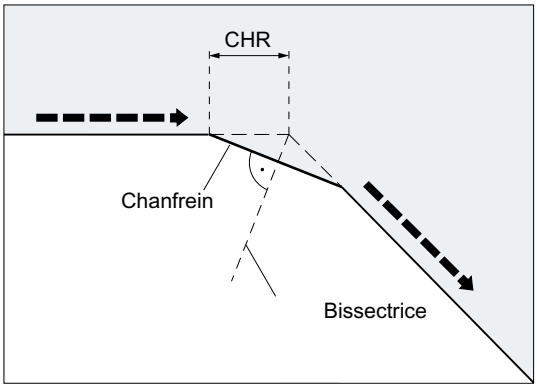
- Arrondi/congé de transition entre les droites :

```
X… Z… RND=...
X… Z… RND=...
X… Z…
```

- Chanfrein de transition entre les droites :

```
X… Z… CHR=...
X… Z… CHR=...
X… Z…
```

Signification

`ANG=...` :	Descripteur pour la programmation de l'angle La valeur indiquée (angle) se rapporte à l'abscisse du plan de travail actif (axe Z dans le cas de `G18`).
`RND=...` :	Descripteur pour la programmation d'un arrondi/congé La valeur indiquée correspond au rayon de l'arrondi / du congé : 
`CHR=...` :	Descripteur pour la programmation d'un chanfrein La valeur indiquée correspond à la largeur du chanfrein dans le sens de déplacement : 
`X...` :	Coordonnées X
`Z...` :	Coordonnées Z

Remarque

Pour plus d'informations sur la programmation d'un chanfrein ou d'un arrondi/congé, voir "Chanfrein, arrondissement (CHF, CHR, RND, RNDM, FRC, FRCM) (Page 250)".

Exemple

Code de programme	Commentaire
N10 X10 Z100 F1000 G18	; Accostage de la position de départ
N20 ANG=140 CHR=7.5	; Droite avec indication d'angle et de chanfrein
N30 X80 Z70 ANG=95.824 RND=10	; Droite vers point intermédiaire avec indication d'angle et d'arrondi
N40 X70 Z50	; Droite vers point final

2.9.9.5 Eléments de contour : Programmation du point final avec un angle

Fonction

Si le caractère d'adresse A apparaît dans un bloc CN, il est possible de ne programmer aucun axe ou de programmer un ou deux axes supplémentaires du plan actif.

Nombre d'axes programmés

- Si **aucun axe** du plan actif n'est programmé, il s'agit soit du premier soit du second bloc d'un élément de contour qui se compose de deux blocs.
  S'il s'agit du deuxième bloc de cet élément de contour, ceci signifie que le point de départ et le point final du plan actif sont identiques. Dans tous les cas, l'élément de contour ne peut se composer que d'un mouvement perpendiculaire au plan actif.
- Si **un seul axe** du plan actif est programmé, il s'agit soit d'une droite unique, dont le point final est déterminé de façon univoque à partir de l'angle et des coordonnées cartésiennes programmées ; soit du deuxième bloc d'un élément de contour se composant de deux blocs. Dans le deuxième cas, la coordonnée manquante est considérée comme égale à la dernière position (modale) atteinte.
- Si **deux axes** du plan actif sont programmés, il s'agit du deuxième bloc d'un élément de contour qui se compose de deux blocs. Si le bloc actuel n'était précédé d'aucun bloc avec programmation d'angle (sans axe programmé du plan actif), un tel bloc n'est pas admis.

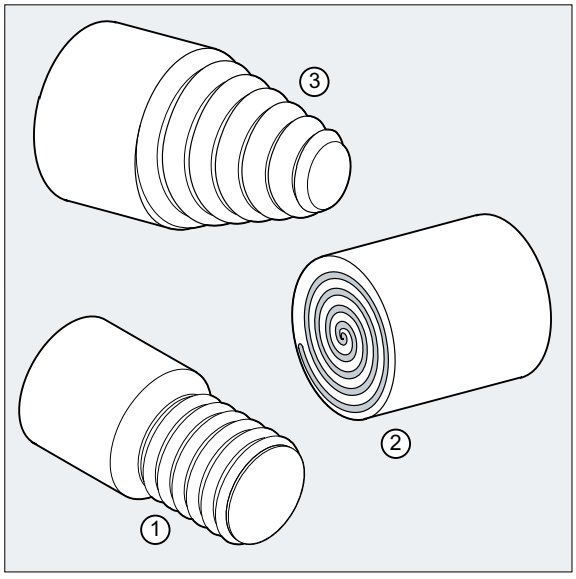
L'angle A doit uniquement être programmé pour les interpolations linéaires ou interpolations de type spline.

2.9.10 Filetage

2.9.10.1 Filetage avec pas constant (G33, SF)

G33 permet de réaliser des filetages avec un pas constant :

- Filetage sur corps cylindrique ①
- Filetage plan ②
- Filetage sur corps conique ③

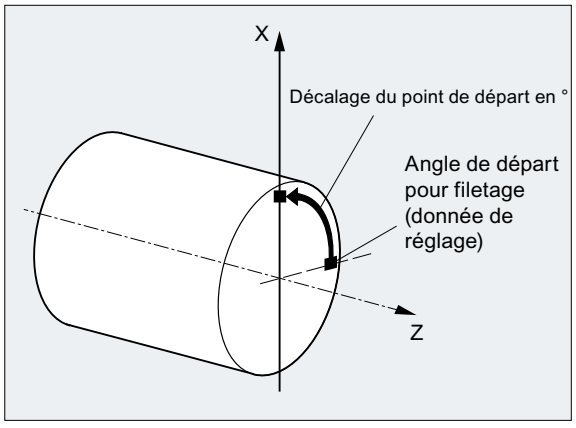


Remarque

Pour le filetage avec G33, la condition technique est une broche en asservissement de vitesse avec système de mesure de déplacement.

Filetages multifilets

Les filetages multifilets (filetages à coupes décalées) peuvent être réalisés par indication d'un décalage du point d'attaque. La programmation a lieu dans le bloc G33, sous l'adresse SF.

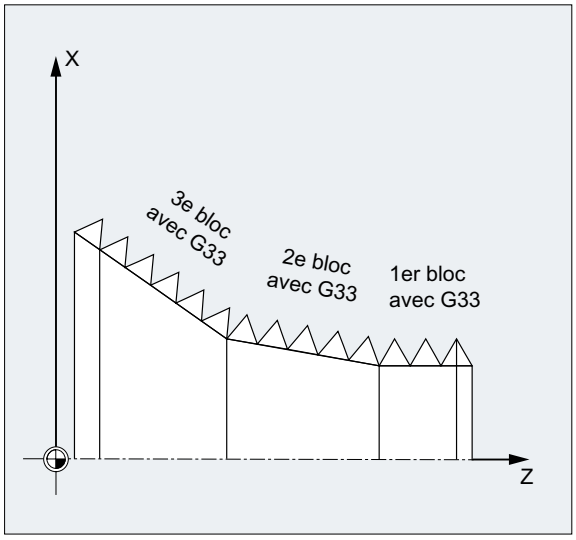


Remarque

Si aucun décalage de point d'attaque n'est indiqué, la commande numérique utilise la valeur qui est rangée dans la donnée de réglage "Angle d'attaque pour filetage".

Concaténation de filetages

La programmation successive de plusieurs blocs G33 permet de réaliser une concaténation de filetages :



Remarque

Avec le contournage `G64`, la transition entre les blocs est régie par un pilotage de vitesse par anticipation qui exclut toute variation brusque de vitesse.

Sens du filetage

Le sens du filetage est défini par le sens de rotation de la broche.

- Une rotation horaire avec `M3` réalise un filetage à droite.
- Une rotation antihoraire avec `M4` réalise un filetage à gauche.

Syntaxe

Filetage sur corps cylindrique :

```
G33 Z... K...
G33 Z... K... SF=...
```

Filetage plan :

```
G33 X... I...
G33 X... I... SF=...
```

Filetage sur corps conique :

```
G33 X... Z... K...
G33 X... Z... K... SF=...
G33 X... Z... I...
G33 X... Z... I... SF=...
```

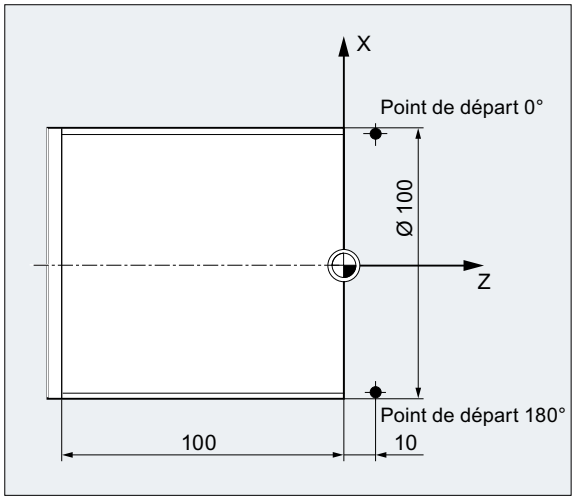
Signification

`G33` :	Instruction de filetage à pas constant
`X... Y... Z...` :	Point final (points finaux) en coordonnées cartésiennes
`I...` :	Pas du filetage dans la direction X

`J...` :	Pas du filetage dans la direction Y	
`K...` :	Pas du filetage dans la direction Z	
`Z` :	axe longitudinal	
`X` :	axe transversal	
`Z... K...` :	Longueur et pas d'un filetage sur corps cylindrique	
`X... I...` :	Diamètre et pas d'un filetage plan	
`I...` ou `K...` :	Pas d'un filetage sur corps conique L'indication (`I...` ou `K...`) dépend de l'angle du corps conique :	
	< 45° :	Le pas du filetage est indiqué par `K...` (pas du filetage dans le sens longitudinal).
	> 45° :	Le pas du filetage est indiqué par `I...` (pas du filetage dans le sens transversal).
	= 45° :	Le pas du filetage peut être indiqué par `I...` ou `K...`.
`SF=...` :	Décalage du point d'attaque (uniquement pour filetages multifilets) Le décalage du point d'attaque est indiqué comme position angulaire absolue.	
	Plage de valeurs :	0,0000 à 359,999 degrés

Exemples

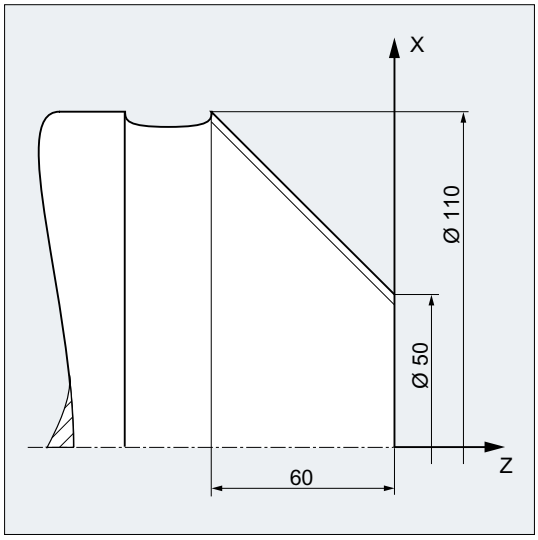
Exemple 1 : filetage sur corps cylindrique à deux filets avec décalage du point d'attaque de 180°



Code de programme	Commentaire
`N10 G1 G54 X99 Z10 S500 F100 M3`	`; Décalage d'origine, accostage du point de départ, mise en marche de la broche`
`N20 G33 Z-100 K4`	`; Filetage sur corps cylindrique : point final en Z`
`N30 G0 X102`	`; Retrait à la position de départ`
`N40 G0 Z10`	
`N50 G1 X99`	

Code de programme	Commentaire
N60 G33 Z-100 K4 SF=180	; 2. filetage : décalage du point de départ de 180°
N70 G0 X110	; Dégagement de l'outil
N80 G0 Z10	
N90 M30	; Fin du programme

Exemple 2 : filetage sur corps conique avec angle inférieur à 45°

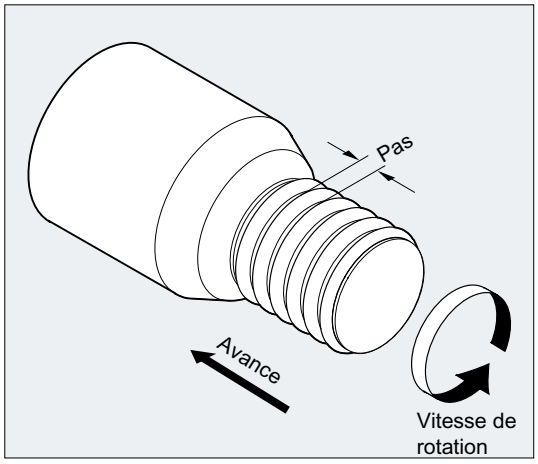


Code de programme	Commentaire
N10 G1 X50 Z0 S500 F100 M3	; Accostage du point de départ, mise en marche de la broche
N20 G33 X110 Z-60 K4	; Filetage sur corps conique : point final en X et Z, pas du filetage indiqué par K... en Z (car angle du corps conique < 45°).
N30 G0 Z0 M30	; Dégagement, fin du programme

Informations complémentaires

Avance pour le filetage avec G33

A partir de la vitesse de rotation de broche et du pas de filetage qui sont programmés, la commande numérique calcule l'avance avec laquelle l'outil de filetage va se déplacer sur la longueur à fileter, dans le sens longitudinal et/ou transversal. L'avance `F` n'est pas prise en considération avec `G33` ; la commande surveille la vitesse des axes (limitation à la vitesse rapide).



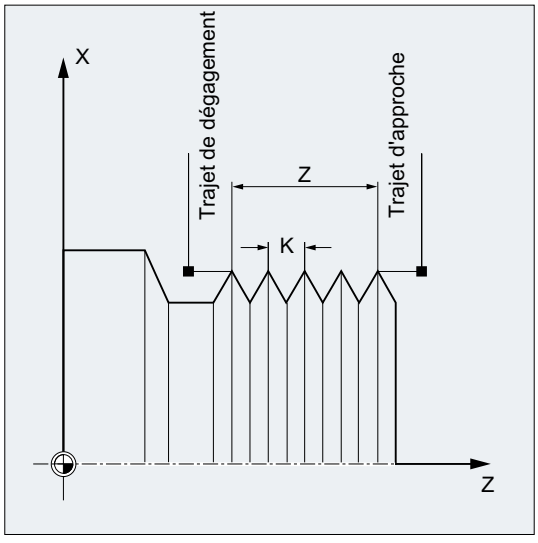
Filetage sur corps cylindrique

Un filetage sur corps cylindrique est défini par :

- Longueur de filetage
- Pas de filetage

La longueur du filetage est introduite en cotes absolues ou relatives avec l'une des coordonnées cartésiennes X, Y ou Z (dans le cas d'un tour, de préférence Z). De plus, il faut prendre en compte les courses d'accélération et de freinage sur lesquelles s'effectuent la montée en vitesse et le freinage de l'axe.

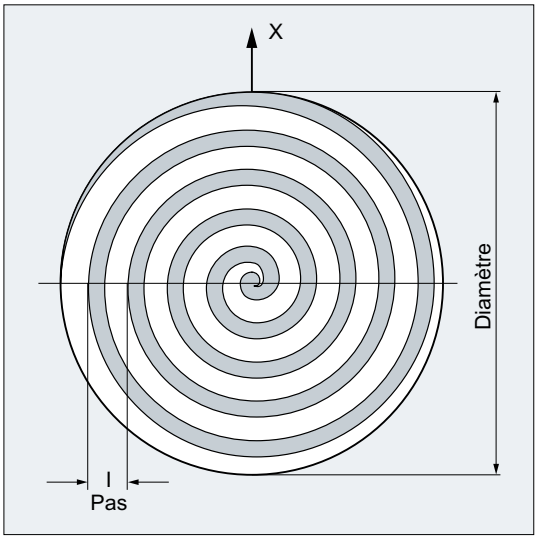
Le pas du filetage est introduit sous les adresses `I`, `J`, `K` (de préférence sous `K` pour un tour).



Filetage plan

Un filetage plan est défini par :

- Diamètre du filetage (de préférence dans l'axe X)
- Pas du filetage (de préférence sous `I`)



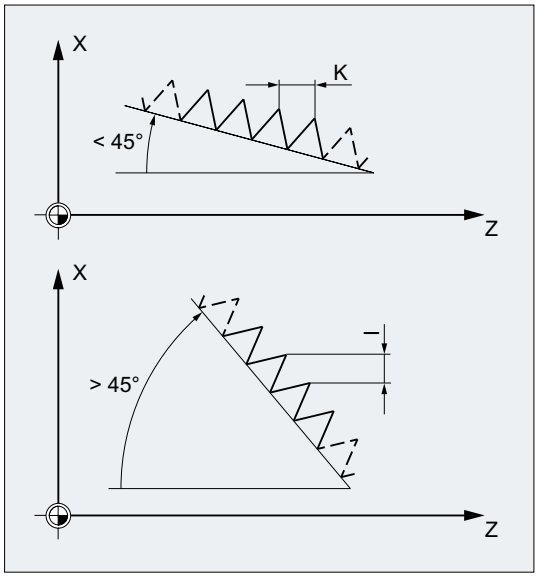
Filetage sur corps conique

Un filetage sur corps conique est défini par :

- Point final dans le sens longitudinal et transversal (contour du corps conique)
- Pas de filetage

Le contour conique est introduit en coordonnées cartésiennes X, Y, Z et en cotes absolues ou relatives ; de préférence suivant X et Z pour un tour. De plus, il faut prendre en compte les courses d'accélération et de freinage sur lesquelles s'effectuent la montée en vitesse et le freinage de l'axe.

L'indication du pas dépend de l'angle du corps conique (angle compris entre l'axe longitudinal et la surface latérale du corps conique) :



2.9.10.2 Filetage avec pas croissant ou décroissant (G34, G35)

Les instructions G34 et G35 complètent la fonctionnalité de G33 en permettant en outre de programmer une variation du pas de filetage sous l'adresse F. G34 entraîne une croissance linéaire et G35 une décroissance linéaire du pas de filetage. Ainsi les instructions G34 et G35 peuvent être utilisées pour la réalisation de filetages à autocisaillement.

Syntaxe

Filetage sur corps cylindrique à pas croissant :
```
G34 Z... K... F...
```

Filetage sur corps cylindrique à pas décroissant :
```
G35 Z... K... F...
```

Filetage plan à pas croissant :
```
G34 X... I... F...
```

Filetage plan à pas décroissant :
```
G35 X... I... F...
```

Filetage sur corps conique à pas croissant :
```
G34 X... Z... K... F...
G34 X... Z... I... F...
```

Filetage sur corps conique à pas décroissant :
```
G35 X... Z... K... F...
G35 X... Z... I... F...
```

Signification

`G34` :	Instruction de filetage avec **croissance** linéaire du pas	
`G35` :	Instruction de filetage avec **décroissance** linéaire du pas	
`X... Y... Z...` :	Point final (points finaux) en coordonnées cartésiennes	
`I...` :	Pas du filetage dans la direction X	
`J...` :	Pas du filetage dans la direction Y	
`K...` :	Pas du filetage dans la direction Z	
`F...` :	Variation du pas de filetage Si le pas de départ et le pas final d'un filetage sont connus, il est possible de calculer avec l'équation suivante la variation du pas de filetage à programmer : $F = \frac{k_e^2 - k_a^2}{2 * l_G}$ [mm/tr^2] Légende :	
	k_e :	pas final du filetage (pas du filetage à la coordonnée du point de destination de l'axe) [mm/tr]
	k_a :	pas de départ du filetage (programmé sous I, J ou K) [mm/tr]
	l_G:	longueur du filetage [mm]

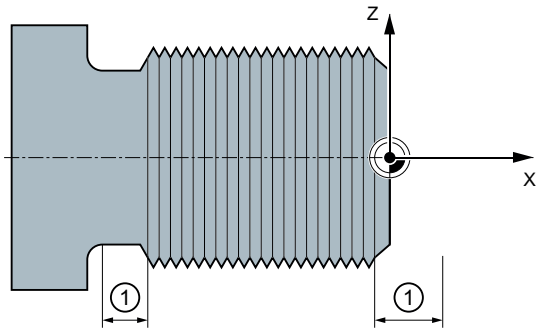
Exemple

Code de programme	Commentaire
N1608 M3 S10	; Mise en marche de la broche
N1609 G0 G64 Z40 X216	; Accostage du point de départ
N1610 G33 Z0 K100 SF=R14	; Filetage avec pas constant (100 mm/tr)
N1611 G35 Z-200 K100 F17.045455	; Décroissance du pas : 17,0454 mm/tr2
	Croissance à la fin du bloc : 50mm/tr
N1612 G33 Z-240 K50	; Exécution du bloc de filetage sans à-coups
N1613 G0 X218	
N1614 G0 Z40	
N1615 M17	

2.9.10.3 Courses d'accélération et de freinage programmées pour G33, G34 et G35 (DITS, DITE)

Les adresses DITS et DITE permettent de spécifier la course d'accélération et de freinage pour le filetage dans le programme pièce.

L'axe du filetage est accéléré ou freiné en respectant la course spécifiée.



① Course d'accélération et de freinage selon le sens d'usinage

Course d'accélération courte

Du fait de l'épaulement à l'entrée du filetage, il n'y a pas assez de place pour l'accélération de l'outil.
Une course d'accélération réduite doit par conséquent être indiquée à l'aide de DITS.

Course de freinage courte

Du fait de l'épaulement à la sortie du filetage, il n'y a pas assez de place pour le freinage de l'outil, ce qui entraîne un risque de collision entre la pièce et le tranchant. DITE permet de spécifier une rampe de freinage plus courte. Toutefois, des collisions ne sont pas à exclure en raison de l'inertie de la mécanique.

Remède : programmer un filetage plus court, réduire la vitesse de la broche.

Remarque

DITE détermine le début de la transition du bloc de filetage au bloc suivant. Ceci garantit une modification sans à-coup de la direction du déplacement.

Conséquences

La course d'accélération et de freinage programmée influe sur la trajectoire en augmentant uniquement l'accélération. Si la valeur de l'une des courses est spécifiée supérieure à la valeur nécessaire pour un axe de filetage avec accélération active, l'axe de filetage est accéléré ou freiné en appliquant l'accélération maximale.

Syntaxe

```
DITS=<valeur> DITE=<valeur>
```

Signification

DITS :	Définir la course d'accélération en filetage
DITE :	Définir la course de freinage en filetage
<Wert> :	DITS et DITE permettent de programmer uniquement des courses, mais pas de position. La course d'accélération / de freinage programmée est traitée selon le système d'unités réglé (anglo-saxon, métrique).

Exemple

Code de programme	**Commentaire**
...	
N40 G90 G0 Z100 X10 SOFT M3 S500	
N50 G33 Z50 K5 SF=180 DITS=1 DITE=3	; Début transition entre blocs à Z=53
N60 G0 X20	

Informations complémentaires

SD42010 $SC_THREAD_RAMP_DISP

Lors du chargement d'un bloc avec DITS et/ou DITE dans l'exécution des blocs, la course d'accélération et de freinage programmée est appliquée dans la donnée de réglage SD42010 $SC_THREAD_RAMP_DISP :

- SD42010 $SC_THREAD_RAMP_DISP[**0**] = valeur programmée de DITS
- SD42010 $SC_THREAD_RAMP_DISP[**1**] = valeur programmée de DITE

Lorsqu'aucune course d'accélération et de freinage n'est programmée avant ou dans le premier bloc de filetage, la valeur actuelle de la donnée de réglage est utilisée.

Comportement après un reset de canal / GMFC / de fin de programme

Les valeurs de SD42010 écrasées par `DITS` et/ou `DITE` restent effectives également après un reset de canal / GMFC / de fin de programme.

Comportement après un démarrage à chaud

Lors d'un démarrage à chaud, la donnée de réglage est réinitialisée aux valeurs effectives avant d'être écrasées par `DITS` et/ou `DITE` (comportement par défaut).

Par contre, si les valeurs programmées à l'aide de `DITS` et `DITE` doivent rester effectives même après un démarrage à chaud, la donnée de réglage SD42010 $SC_THREAD_RAMP_DISP doit être répertoriée dans le paramètre machine PM10710 $MN_PROG_SD_RESET_SAVE_TAB :

MD10710 $MN_PROG_SD_RESET_SAVE_TAB[<n>] = 42010

Comportement dans le cas d'une course d'accélération et/ou de freinage très courte

Dans le cas d'une course d'accélération et/ou de freinage très courte, l'accélération et/ou la décélération de l'axe est supérieure à la valeur maximale réglée. L'entraînement d'avance de l'axe est donc surchargé.

Pour l'entrée du filetage, l'alarme 22280 "Course d'accélération trop courte" est émises (dans le cas d'une configuration correspondante dans le paramètre machine MD11411 $MN_ENABLE_ALARM_MASK). Cette alarme a un caractère purement informatif et n'a aucun effet sur l'exécution du programme pièce.

2.9.10.4 Retrait rapide pendant un filetage (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN)

La fonction "Retrait rapide en cours de filetage (G33)" permet une interruption du filetage évitant tout défaut dans les situations suivantes :

- Suspension de programme déclenchée via DB21, ... DBX7.3
- Suspension de programme déclenchée implicitement via une alarme
- Actionnement d'une entrée rapide (Page 556)

La programmation du mouvement de retrait s'effectue par :

- Trajet et direction du retrait (relatifs)
- Position du retrait (absolue)

Remarque

Signaux de suspension de programme

Les signaux de suspension de programme suivants ne déclenchent pas de retrait rapide lors du filetage :

- DB21, ... DBX3.4 (suspension programme axes plus broches)
- DB21, ... DBX7.2 (suspension programme en fin de bloc)

Taraudage

La fonction "Retrait rapide" n'est **pas** disponible pour le **taraudage** (G331 / G332.

Syntaxe

Débloquer le retrait rapide, mouvement de retrait par trajet et direction de retrait :
`G33 ... LFON DILF=<valeur> LFTXT/LFWP ALF=<valeur>`

Débloquer le retrait rapide, mouvement de retrait par position de retrait :

```
POLF[<Descripteur d'axe>]=<Valeur> LFPOS
POLFMASK/POLFMLIN(<nom d'axe1>,<nom d'axe2>,...)
G33 ... LFON
```

Bloquer retrait rapide en cours de filetage :
`LFOF`

Signification

`LFON` :	Déblocage du retrait rapide en cours de filetage (`G33`)	
`LFOF` :	Blocage du retrait rapide en cours de filetage (`G33`)	
`DILF=` :	Définir la longueur de la distance de retrait	
	La valeur prédéfinie par la configuration MD (MD21200 $MC_LIFTFAST_DIST) peut être modifiée par programmation de `DILF` dans le programme pièce. **Remarque :** Après un RESET de la commande numérique, la valeur active est toujours la valeur PM configurée.	
`LFTXT` `LFWP` :	La direction de retrait est commandée en relation avec `ALF` avec les instructions G `LFTXT` et `LFWP`.	
	`LFTXT` :	Le plan dans lequel est exécuté le mouvement de retrait est déterminé à partir de la tangente à la trajectoire et de la direction de l'outil (réglage par défaut).
	`LFWP` :	Le plan dans lequel est exécuté le mouvement de retrait est le plan de travail actif.
`ALF=` :	Dans le plan du mouvement de retrait, la direction est programmée avec `ALF` par pas discrets en degrés.	
	Avec `LFTXT`, le retrait est défini dans la direction de l'outil pour `ALF=1`. Avec `LFWP`, la direction dans le plan de travail est tributaire des configurations suivantes : • `G17` (plan X/Y) `ALF=1` ; retrait en direction X `ALF=3` ; retrait en direction Y • `G18` (plan Z/X) `ALF=1` ; retrait en direction Z `ALF=3` ; retrait en direction X • `G19` (plan Y/Z) `ALF=1` ; retrait en direction Y `ALF=3` ; retrait en direction Z	
	Des informations complémentaires sur les possibilités de programmation avec `ALF` figurent au chapitre "Sens de retrait rapide du contour (Page 558)".	
`LFPOS` :	Retrait de l'axe introduit avec `POLFMASK` ou `POLFMLIN` sur la position d'axe absolue programmée avec `POLF`	

POLFMASK :	Déblocage des axes (<nom d'axe1>,<nom d'axe1>,...) pour le retrait indépendant sur la position absolue	
POLFMLIN :	Déblocage des axes pour le retrait sur la position absolue en corrélation linéaire **Remarque :** En fonction du comportement dynamique de tous les axes concernés, la corrélation linéaire n'est pas toujours établie jusqu'à la position de relèvement.	
POLF[] :	Position de retrait absolue pour l'axe géométrique ou l'axe machine indiqués dans l'indice	
	Prise d'effet :	modale
	=<valeur> :	Pour les axes géométriques, la valeur affectée est interprétée comme position dans le système de coordonnées pièce (SCP), pour les axes machine, comme position dans le système de coordonnées machine (SCM). L'affectation de valeur peut également être programmée en cotation relative : =IC<valeur>
<Descripteur d'axe> :	Descripteur d'un axe géométrique ou machine	

Remarque

LFON ou LFOF sont toujours programmables, ils sont cependant exploités exclusivement pour le filetage (G33).

Remarque

POLF avec POLFMASK/POLFMLIN ne sont pas limités à l'utilisation pour le filetage.

Exemples

Exemple 1 : Débloquer retrait rapide en cours de filetage

Code de programme	Commentaire
N55 M3 S500 G90 G18	; Plan d'usinage actif
...	; Accostage de la position de départ
N65 MSG ("filetage")	; Approche de l'outil
MM_THREAD :	
N67 $AC_LIFTFAST=0	; Remise à zéro avant le début du filetage
N68 G0 Z5	
N68 X10	
N70 G33 Z30 K5 LFON DILF=10 LFWP ALF=7	; Déblocage du retrait rapide pendant le filetage Distance de retrait = 10 mm Plan de retrait : Z/X (en raison de G18) Sens de retrait : -X (avec ALF=3 : sens de retrait +X)
N71 G33 Z55 X15	

Code de programme	Commentaire
N72 G1	; Désactivation du filetage
N69 IF $AC_LIFTFAST GOTOB MM_THREAD	; Si le filetage a été interrompu
N90 MSG ("")	
...	
N70 M30	

Exemple 2 : désactiver le retrait rapide avant le taraudage

Code de programme	Commentaire
N55 M3 S500 G90 G0 X0 Z0	
...	
N87 MSG ("taraudage")	
N88 LFOF	; Désactivation du retrait rapide avant taraudage
N89 CYCLE...	; Cycle de taraudage avec G33
N90 MSG ("")	
...	
N99 M30	

Exemple 3 : retrait rapide à la position de retrait absolue

En cas d'arrêt, l'interpolation de trajectoire est inhibée par X et à la place, l'interpolation d'un déplacement avec une vitesse maximale a lieu vers la position POLF[X]. Le déplacement des autres axes reste déterminé par le contour programmé ou par le pas de filetage ou encore la vitesse de broche.

Code de programme	Commentaire
N10 G0 G90 X200 Z0 S200 M3	
N20 G0 G90 X170	
N22 POLF[X]=210 LFPOS	
N23 POLFMASK(X)	; Activation (déblocage) du relèvement rapide de l'axe X
N25 G33 X100 I10 LFON	
N30 X135 Z-45 K10	
N40 X155 Z-128 K10	
N50 X145 Z-168 K10	
N55 X210 I10	
N60 G0 Z0 LFOF	
N70 POLFMASK()	; Blocage du relèvement pour tous les axes
M30	

2.9.10.5 Filetages bombés (G335, G336)

Les instructions G G335 et G336 permettent de tourner des filetages bombés (= non cylindriques). L'usinage de pièces de très grande taille qui fléchissent dans la machine en raison de leur propre poids en est une application. Des filetages paraxiaux entraîneraient un filetage trop faible au milieu de la pièce. Les filetages bombés permettent de compenser ce phénomène.

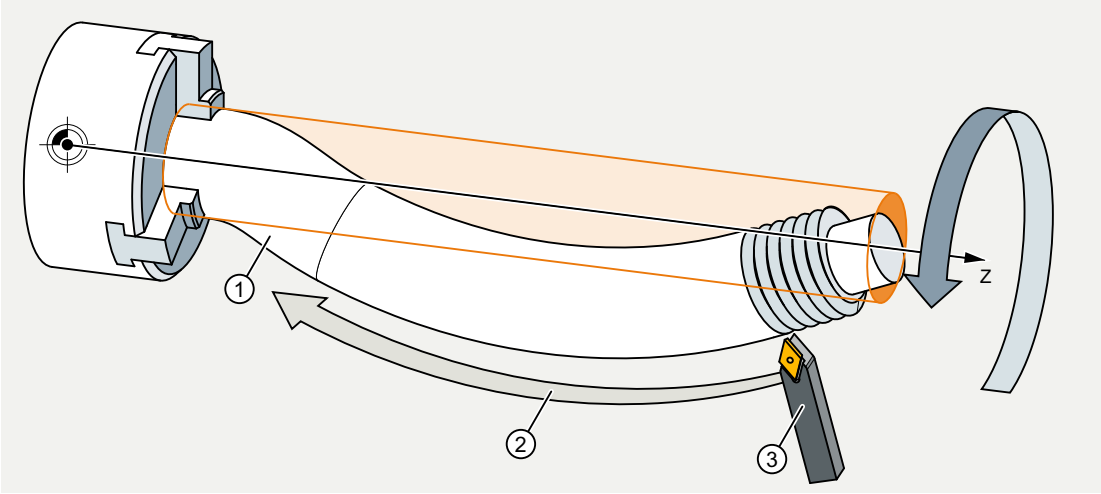


Figure 2-6 Tournage d'un filetage bombé

Programmation

Le tournage de filetages bombés est programmé avec `G335` ou `G336` :

`G335` :	Tournage d'un filetage bombé sur une trajectoire d'outil circulaire dans **le sens horaire**.
`G336` :	Tournage d'un filetage bombé sur une trajectoire d'outil circulaire dans **le sens antihoraire**.

La programmation est effectuée d'abord comme pour un filetage linéaire en indiquant les points de fin de bloc axiaux et le pas via les paramètres `I`, `J`, et `K` (voir "Filetage avec pas constant (G33, SF) (Page 225)").

Ensuite un arc de cercle est également indiqué. Celui-ci peut être programmé comme pour `G2`/`G3` par l'indication du centre, du rayon, de l'angle au centre ou du point intermédiaire (voir "Interpolation circulaire (Page 193)"). Lors de la programmation du filetage bombé avec programmation du centre, respecter les points suivants : Etant donné que pour le filetage `I`, `J`, et `K` sont utilisés pour le pas, les paramètres du cercle doivent être programmés dans la programmation du centre avec `IR=...`, `JR=...` et `KR=...`.

`IR=...` :	Coordonnée cartésienne pour le centre du cercle en X
`JR=...` :	Coordonnée cartésienne pour le centre du cercle en Y
`KR=...` :	Coordonnée cartésienne pour le centre du cercle en Z

Remarque

IR, JR et KR sont les valeurs par défaut des noms réglables via le paramètre machine (PM10651 $MN_IPO_PARAM_THREAD_NAME_TAB) des paramètres d'interpolation pour filetages bombés.

Les écarts par rapport à ces valeurs par défaut figurent dans les indications du constructeur de machines.

En option, un décalage de point de départ `SF` peut également être indiqué (voir "Filetage avec pas constant (G33, SF) (Page 225)").

Syntaxe

La syntaxe pour la programmation d'un filetage bombé présente donc la forme générale suivante :

```
G335/G336 <Coordonnée(s) du point de destination de l'axe> <Pas> <Arc de cercle> [<Décalage du point de départ>]
```

Exemples

Exemple 1 : filetage bombé dans le sens horaire avec programmation du point final et du centre

Code de programme	Commentaire
`N5 G0 G18 X50 Z50`	`; Accostage du point de départ`
`N10 G335 Z100 K=3,5 KR=25 IR=-20 SF=90`	`; Tournage du filetage bombé dans le sens horaire`

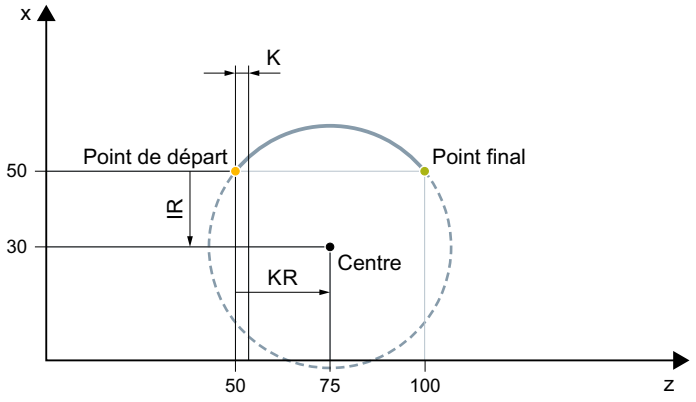


Figure 2-7 Filetage bombé dans le sens horaire avec programmation du point final et du centre

Exemple 2 : filetage bombé dans le sens antihoraire avec programmation du point final et du centre

Code de programme	Commentaire
`N5 G0 G18 X50 Z50`	`; Accostage du point de départ`

Code de programme	Commentaire
N10 **G336 Z100** K=3,5 **KR=25 IR=20** SF=90	; Tournage du filetage bombé dans le sens antihoraire

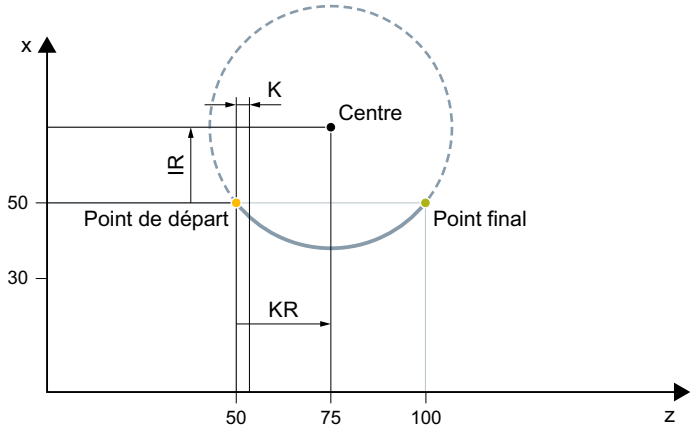


Figure 2-8 Filetage bombé dans le sens antihoraire avec programmation du point final et du centre

Exemple 3 : filetage bombé dans le sens horaire avec programmation du point final et du rayon

Code de programme
N5 G0 G18 X50 Z50
N10 **G335 Z100** K=3,5 **CR=32** SF=90

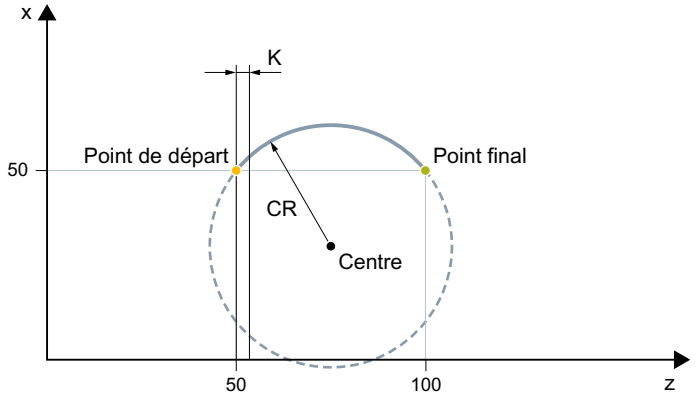


Figure 2-9 Filetage bombé dans le sens horaire avec programmation du point final et du rayon

Exemple 4 : filetage bombé dans le sens horaire avec programmation du point final et de l'angle au centre

Code de programme
N5 G0 G18 X50 Z50
N10 **G335 Z100** K=3,5 **AR=102,75** SF=90

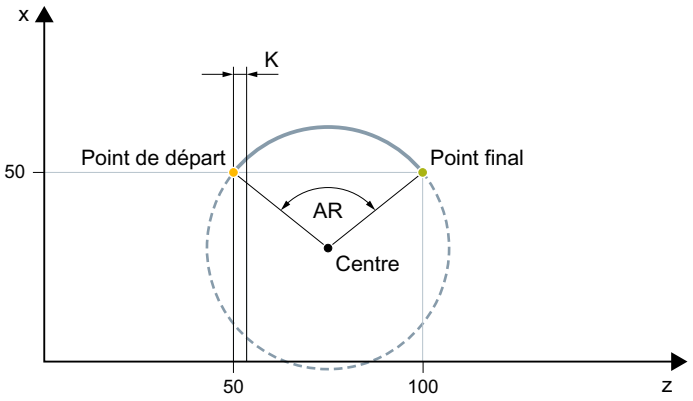


Figure 2-10 Filetage bombé dans le sens horaire avec programmation du point final et de l'angle au centre

Exemple 5 : filetage bombé dans le sens horaire avec programmation du centre et de l'angle au centre

Code de programme

```
N5 G0 G18 X50 Z50
N10 G335 K=3,5 KR=25 IR=-20 AR=102,75 SF=90
```

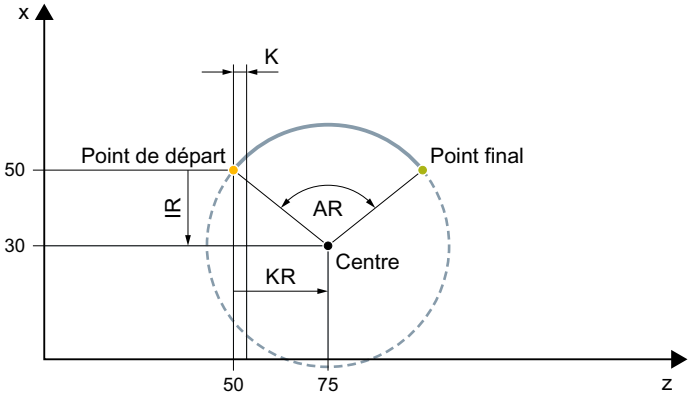


Figure 2-11 Filetage bombé dans le sens horaire avec programmation du centre et de l'angle au centre

Exemple 6 : filetage bombé dans le sens horaire avec programmation du point final et du point intermédiaire

Code de programme

```
N5 G0 G18 X50 Z50
N10 G335 Z100 K=3,5 I1=60 K1=64
```

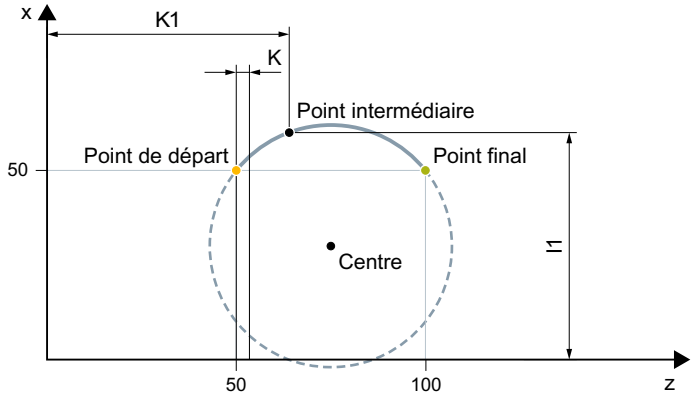
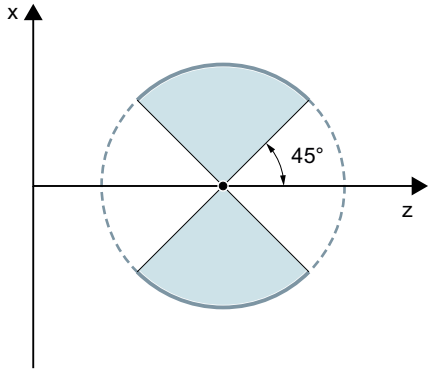


Figure 2-12 Filetage bombé dans le sens horaire avec programmation du point final et du point intermédiaire

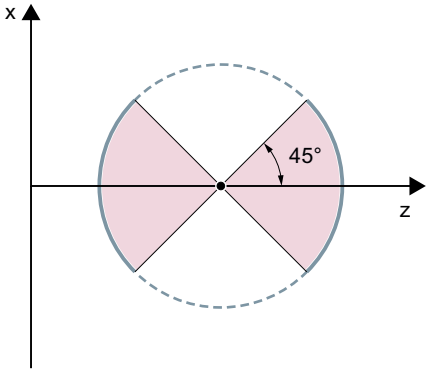
Informations complémentaires

Plages d'arc de cercle autorisées

L'arc de cercle programmé sous G335/G336 doit se trouver dans une plage dans laquelle l'axe principal de filetage spécifié (I, J ou K) présente sur la totalité de l'arc de cercle la part d'axe principal sur l'arc de cercle :



Plages autorisées pour l'axe **Z** (pas programmé avec K)



Plages autorisées pour l'axe **X** (pas programmé avec I)

Un changement de l'axe principal de filetage, tel qu'illustré sur la figure suivante, n'est **pas** autorisé :

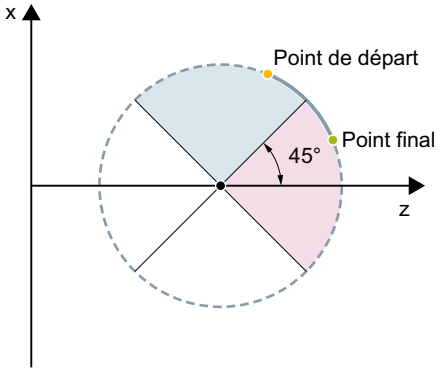


Figure 2-13 Filetages bombés : plage interdite

Frames

Les fonctions G335 et G336 sont également possibles lorsque les frames sont actifs. Il convient cependant de s'assurer que les plages d'arc de cercle autorisées sont respectées dans le système de coordonnées de base (SCB).

Autres conditions à prendre en compte pour la programmation du cercle

Pour la programmation du cercle sous G335/G336, les autres conditions à prendre en compte décrites pour la programmation du cercle avec G2/G3 sont applicables (voir "Interpolation circulaire (Page 193)").

2.9.11 Taraudage sans porte-taraud compensateur

2.9.11.1 Taraudage sans porte-taraud compensateur et déplacement de retrait (G331, G332)

Lors du taraudage sans porte-taraud compensateur, les instructions `G331` et `G332` permettent d'exécuter les déplacements suivants :

- `G331` : Taraudage dans la direction de perçage jusqu'au point final du filetage
- `G332` : Déplacement de retrait vers le bloc de taraudage `G331` avec inversion automatique du sens de rotation de la broche

Syntaxe

```
G331 <axe> <pas de filetage>
G331 <axe> <pas de filetage> S...
G332 <axe> <pas de filetage>
```

Signification

G331 :	Taraudage Le taraudage est décrit par le déplacement de l'axe (profondeur de perçage) et le pas de filetage.	
	Prise d'effet :	modale
G332 :	Déplacement de retrait du taraudage Le déplacement de retrait doit être effectué avec le même pas que le taraudage (G331). L'inversion du sens de rotation de la broche s'effectue automatiquement.	
	Prise d'effet :	modale
<axe> :	Déplacement/position de l'axe géométrique (X, Y ou Z) vers le point final du filetage, par ex. Z50	
<pas de filetage> :	Pas de filetage I (X), J (Y) ou K (Z) : • Pas positif : filetage à droite, par ex. K1,25 • Pas négatif : filetage à gauche, par ex. K-1,25	
	Plage de valeurs :	±0,001 à ±2000,00 mm/tour
S... :	Vitesse de rotation de broche Si aucune vitesse de rotation de broche, le déplacement s'effectue avec la dernière vitesse de rotation de broche effective.	

Remarque

Deuxième jeu de paramètres machine des rapports de transmission

Afin d'atteindre une adaptation effective entre la vitesse de rotation de broche et le moment du moteur et ainsi de pouvoir accélérer plus rapidement lors du taraudage, il est possible, pour deux autres seuils de commutation configurables (vitesse de rotation maximale et vitesse de rotation minimale), de régler dans des paramètres machine spécifiques à l'axe, un deuxième jeu de paramètres machine des rapports de transmission qui diverge du premier jeu de paramètres et qui est également indépendant de ces seuils de commutation de vitesse de rotation. Les indications du constructeur de machines doivent être respectées.

Pour plus d'informations, voir Description fonctionnelle "Axes et broches".

Exemples

- Exemple : Taraudage avec G331 / G332 (Page 246)
- Exemple : édition de la vitesse de taraudage programmée dans le rapport de transmission actuel (Page 246)
- Exemple : application du deuxième jeu de paramètres machine des rapports de transmission (Page 247)
- Exemple : Aucune programmation de vitesse de rotation, surveillance du rapport de transmission (Page 247)
- Exemple : Changement du rapport de transmission impossible, surveillance du rapport de transmission (Page 248)
- Exemple : programmation sans SPOS (Page 248)

2.9.11.2 Exemple : Taraudage avec G331 / G332

Code de programme	Commentaire
N10 SPOS[n]=0	; Broche : Mode d'asservissement de position
	; Position de départ 0 degrés
N20 G0 X0 Y0 Z2	; Axes : Accostage de la position de départ
N30 G331 Z-50 K-4 S200	; Taraudage en Z,
	; Pas K-4 négatif =>
	; Sens de rotation de la broche : Marche à gauche,
	; Vitesse de rotation de la broche 200 tr/min
N40 G332 Z3 K-4	; Déplacement de retrait en Z,
	; Pas K-4 négatif (marche à gauche),
	; Inversion automatique du sens de rotation =>
	; Sens de rotation de la broche Marche à droite
N50 G1 F1000 X100 Y100 Z100 S300 M3	; Broche en mode broche.

2.9.11.3 Exemple : édition de la vitesse de taraudage programmée dans le rapport de transmission actuel

Code de programme	Commentaire
N05 M40 S500	; Vitesse de rotation de broche programmée : 500 tr/min =>
	; Rapport de transmission 1 (20 à 1028 tr/min)
...	
N55 SPOS=0	; Positionnement de la broche
N60 G331 Z-10 K5 S800	; Taraudage
	; Vitesse de rotation de broche 800 tr/min => Rapport de transmission 1

Le rapport de transmission adéquat pour la vitesse de rotation de broche programmée `S500` est déterminé à partir du premier jeu de paramètres machine du rapport de transmission pour `M40`. La vitesse de taraudage programmée `S800` est éditée dans le rapport de transmission actuel et, le cas échéant, limitée à la vitesse maximale du rapport de transmission. Une commutation automatique des rapports de transmission est impossible après l'exécution de `SPOS`. La

condition pour la commutation automatique des rapports de transmission est le mode de régulation de vitesse de la broche.

Remarque

S'il s'agit de sélectionner le rapport de transmission 2 pour une vitesse de rotation de la broche égale à 800 tr/min, les seuils de commutation pour les vitesses de rotation maximale et minimale doivent être configurés dans les paramètres machine correspondants du deuxième jeu de paramètres machine des rapports de transmission (voir exemples suivants).

2.9.11.4 Exemple : application du deuxième jeu de paramètres machine des rapports de transmission

Avec `G331`/`G332` et la programmation d'une valeur `S`, les seuils de commutation du deuxième jeu de paramètres machine des rapports de transmission pour les vitesses de rotation maximale et minimale sont exploités pour la broche maître active. La commutation automatique des rapports de transmission `M40` doit être active. Le rapport de transmission ainsi déterminé est comparé au rapport de transmission actif. Si une différence existe entre les deux, une commutation des rapports de transmission est effectuée.

Code de programme	Commentaire
`N05 M40 S500`	`; Vitesse de rotation de broche programmée : 500 tr/min`
`...`	
`N50 G331 S800`	`; Broche maître : ; Sélection du rapport de transmission 2`
`N55 SPOS=0`	`; Positionnement de la broche`
`N60 G331 Z-10 K5`	`; Taraudage`
	`; Accélération de la broche à partir du deuxième jeu de paramètres des rapports de transmission`

2.9.11.5 Exemple : Aucune programmation de vitesse de rotation, surveillance du rapport de transmission

Si aucune vitesse de rotation n'est programmée pour l'application du deuxième jeu de paramètres des rapports de transmission avec `G331`, le taraudage est réalisé à la vitesse de rotation programmée en dernier. Il n'y a pas de changement de rapport de transmission. Dans ce cas, une surveillance a cependant lieu pour déterminer si la vitesse de rotation programmée en dernier se trouve dans la plage de vitesses de rotation de broche (seuils de commutation pour la vitesse de rotation maximale et minimale) prédéfinie du rapport de transmission actif. Dans le cas contraire, l'alarme 16748 est affichée.

Code de programme	Commentaire
`N05 M40 S800`	`; Vitesse de rotation de broche programmée : 800 tr/min`
`...`	
`N55 SPOS=0`	`; Positionnement de la broche`

Code de programme	Commentaire
N60 G331 Z-10 K5	; Taraudage
	; Surveillance de la vitesse de rotation de broche de 800 tr/min
	; Rapport de transmission 1 actif
	; Le rapport de transmission 2 devrait être actif => Alarme 16748

2.9.11.6 Exemple : Changement du rapport de transmission impossible, surveillance du rapport de transmission

Si la vitesse de rotation de broche est programmée en plus de la géométrie lors de l'application du deuxième jeu de paramètres du rapport de transmission dans le bloc G331 et si la vitesse de rotation ne se trouve pas dans la plage de vitesses (seuils de commutation pour la vitesse de rotation maximale et minimale) prédéfinie du rapport de transmission actif, aucun changement de rapport de transmission ne peut être réalisé, car le déplacement avec interpolation ne pourrait pas être respecté par la broche et les axes de pénétration.

Comme dans l'exemple précédent, la vitesse de rotation et le rapport de transmission sont surveillés dans le bloc G331 et, le cas échéant, l'alarme 16748 est affichée.

Code de programme	Commentaire
N05 M40 S500	; Vitesse de rotation de broche programmée : 500 tr/min =>
	; Rapport de transmission 1
...	
N55 SPOS=0	; Positionnement de la broche
N60 G331 Z-10 K5 S800	; Taraudage
	; Changement du rapport de transmission impossible,
	; Surveillance de la vitesse de rotation de broche de 800 tr/min
	; avec le premier jeu de paramètres du rapport de transmission : Le rapport de boîte de vitesses 2
	; devrait être actif => Alarme 16748

2.9.11.7 Exemple : programmation sans SPOS

Code de programme	Commentaire
N05 M40 S500	; Vitesse de rotation de broche programmée : 500 tr/min =>
	; Rapport de transmission 1 (20 à 1028 tr/min)
...	
N50 G331 S800	; Broche maître : ; Sélection du rapport de transmission 2
N60 G331 Z-10 K5	; Taraudage
	; Accélération de la broche à partir du deuxième jeu de paramètres des rapports de transmission

L'interpolation du taraudage pour la broche débute à la position actuelle, qui dépend de la plage du programme pièce précédemment traitée, p. ex. en cas d'exécution d'un changement de

rapport de transmission. Un traitement ultérieur du réducteur n'est de ce fait pas possible, le cas échéant.

Remarque

Lorsque vous utilisez plusieurs broches, vous devez vous assurer que la broche de taraudage soit également la broche maître. `SETMS(<numéro de broche>)` permet de programmer la broche de taraudage comme broche maître.

2.9.12 Taraudage avec porte-taraud compensateur

2.9.12.1 Taraudage avec porte-taraud compensateur et déplacement de retrait (G63)

Lors du taraudage avec porte-taraud compensateur, l'instruction `G63` permet d'exécuter les déplacements suivants :

- `G63` : Taraudage dans la direction de perçage jusqu'au point final du filetage
- `G63` : Déplacement de retrait avec inversion programmée du sens de rotation de la broche

Remarque

Après un bloc `G63`, le dernier type d'interpolation effectif `G0`, `G1`, `G2` redevient actif.

Syntaxe

```
G63 <axe> <sens de rotation> <vitesse de rotation> <avance>
```

Signification

`G63` :	Taraudage avec porte-taraud compensateur	
	Prise d'effet :	bloc par bloc (non modale)
`<axe>` :	Déplacement/position de l'axe géométrique (X, Y ou Z) vers le point final du filetage, par ex. Z50	
`<sens de rotation>` :	Sens de rotation de la broche : • `M3` : Rotation dans le sens horaire, filetage à droite • `M4` : Rotation dans le sens antihoraire, filetage à gauche	
`<vitesse de rotation>` :	Vitesse de rotation de broche maximale admissible pendant le taraudage, par ex. S100	
`<avance>` :	Avance de l'axe de perçage F, avec F = vitesse de rotation de broche * pas de filetage	

Exemple

Réalisation d'un taraudage M5 :

- Pas de filetage selon la norme : 0,8 mm/tr
- Vitesse de rotation de broche S : 200 tr/min
- Avance F = 200 tr/min * 0,8 mm/tr = 160 mm/min.

Code de programme	**Commentaire**
N10 G1 X0 Y0 Z2 F1000 S200 M3	; Accostage du point de départ
	; Broche dans le sens horaire, 200 tr/min
N20 G63 Z-50 F160	; Taraudage avec porte-taraud compensateur
	; Profondeur de perçage : Z absolu=50 mm
	; Avance : 160 mm/min
N30 G63 Z3 M4	; Déplacement de retrait : Z absolu=3 mm
	; Inversion du sens de rotation
	; Broche dans le sens antihoraire, 200 tr/min

2.9.13 Chanfrein, arrondissement (CHF, CHR, RND, RNDM, FRC, FRCM)

Dans le plan de travail actif, les angles de contour peuvent être réalisés sous forme d'arrondis ou de chanfreins.

Pour optimiser l'état de surface, vous pouvez programmer une avance spécifique pour le chanfreinage/arrondissement. Si aucune avance n'est programmée, l'avance tangentielle F normale est appliquée.

La fonction "Arrondissement modal" permet d'arrondir de façon identique plusieurs angles de contour successifs.

Syntaxe

Chanfreiner un angle de contour :
```
G... X... Z... CHR/CHF=<valeur> FRC/FRCM=<valeur>
G... X... Z...
```

Arrondir un angle de contour :
```
G... X... Z... RND=<valeur> FRC=<valeur>
G... X... Z...
```

Arrondissement modal :

```
G... X... Z... RNDM=<valeur> FRCM=<valeur>
...
RNDM=0
```

Remarque

La technologie (avance, type d'avance, instructions M...) pour le chanfreinage/l'arrondissement est déduite soit du bloc précédent, soit du bloc suivant, en fonction du réglage du bit 0 dans le paramètre machine PM20201 $MC_CHFRND_MODE_MASK (comportement de chanfreinage/arrondissement). La déduction à partir du bloc précédent (bit 0 = 1) correspond au réglage recommandé.

Signification

`CHF=...` :	Chanfreiner un angle de contour	
	`<valeur>` :	Longueur du chanfrein (unité selon G70/G71)
`CHR=...` :	Chanfreiner un angle de contour	
	`<valeur>` :	Largeur du chanfrein dans le sens de déplacement initial (unité selon G70/G71)
`RND=...` :	Arrondir un angle de contour	
	`<valeur>` :	Rayon de l'arrondissement (unité selon G70/G71)
`RNDM=...` :	Arrondissement modal (arrondir de façon identique plusieurs angles de contour successifs)	
	`<valeur>` :	Rayon des arrondissements (unité selon G70/G71)
		`RNDM=0` permet de désactiver l'arrondissement modal.
`FRC=...` :	Avance non modale pour le chanfreinage/l'arrondissement	
	`<valeur>` :	Vitesse d'avance en mm/min (si G94 est actif) ou en mm/tr (si G95 est actif)
`FRCM=...` :	Avance modale pour le chanfreinage/l'arrondissement	
	`<valeur>` :	Vitesse d'avance en mm/min (si G94 est actif) ou en mm/tr (si G95 est actif)
		`FRCM=0` permet de désactiver l'avance modale pour le chanfreinage/l'arrondissement et d'activer l'avance programmée sous F.

Remarque

Chanfrein/arrondi trop grand

Si les valeurs programmées pour les chanfreins (CHF/CHR) et arrondis (RND/RNDM) sont trop grandes pour les éléments de contour concernés, les chanfreins et arrondis sont adaptés automatiquement :

1. Si le PM11411 $MN_ENABLE_ALARM_MASK bit 4 est mis à 1, l'alarme 10833 "Chanfrein ou arrondi doit être réduit" est générée (alarme Cancel).
2. Le chanfrein / l'arrondi est réduit de manière à ce qu'il corresponde à l'angle de contour. Il en découle au moins un bloc sans déplacement. Au niveau de ce bloc, le déplacement est nécessairement interrompu.

Remarque

Chanfrein/arrondi impossible

Aucun chanfrein/ni arrondi n'est inséré si :

- aucun segment de droite ou contour circulaire n'existe dans le plan,
- un déplacement a lieu hors du plan,
- un changement de plan est réalisé,
- le nombre de blocs ne comportant aucune information de déplacement (p. ex. uniquement des instructions) et définis dans les paramètres machine est dépassé.

Remarque

FRC/FRCM

FRC/FRCM ne s'applique pas si un chanfrein est usiné avec G0 ; la programmation est possible sans message d'erreur, selon la valeur F.

FRC est uniquement actif si un chanfrein/arrondi est programmé dans le bloc ou si RNDM a été activé.

FRC écrase la valeur F ou FRCM dans le bloc courant.

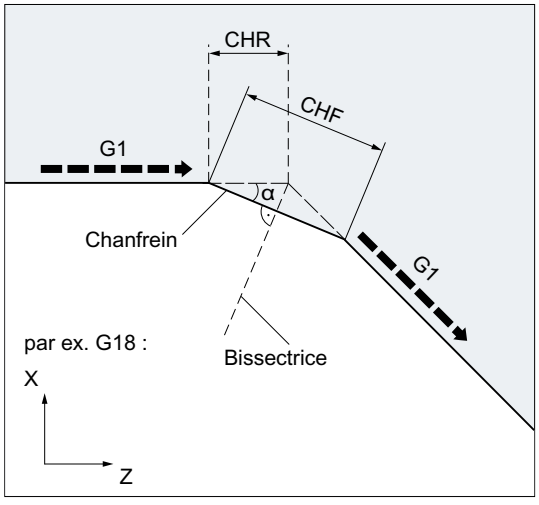
L'avance programmée sous FRC doit être supérieure à zéro.

FRCM=0 active l'avance programmée sous F pour le fraisage/l'arrondissement.

Si FRCM est programmé, la valeur FRCM doit être reprogrammée comme F en cas de changement G94 ↔ G95, etc., Si uniquement F est reprogrammé et si FRCM > 0 avant le changement de type d'avance, un message d'erreur est émis.

Exemples

Exemple 1 : chanfreinage entre deux droites



- PM20201 bit 0 = 1 (déduction à partir du bloc précédent)
- G71 est actif.
- La largeur du chanfrein dans le sens de déplacement (CHR) doit être égale à 2 mm, l'avance pour le chanfreinage à 100 mm/min.

La programmation peut être effectuée de deux façons :

- Programmation avec CHR

Code de programme

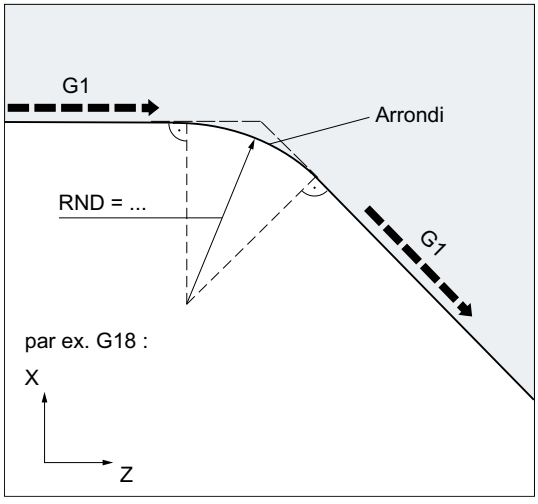
```
...
N30 G1 Z… CHR=2 FRC=100
N40 G1 X…
...
```

- Programmation avec CHF

Code de programme

```
...
N30 G1 Z… CHF=2(cosα*2) FRC=100
N40 G1 X…
...
```

Exemple 2 : arrondissement entre deux droites



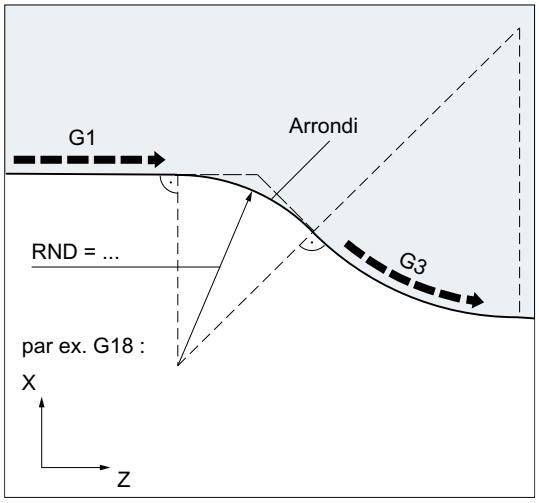
- PM20201 bit 0 = 1 (déduction à partir du bloc précédent)
- G71 est actif.
- Le rayon de l'arrondi doit être égal à 2 mm, l'avance à 50 mm/min.

Code de programme

```
...
N30 G1 Z… RND=2 FRC=50
N40 G1 X…
...
```

Exemple 3 : arrondissement entre une droite et un cercle

La fonction RND avec un raccordement tangentiel permet d'insérer un élément de contour circulaire entre des contours linéaires et circulaires, en combinaison quelconque.



- PM20201 bit 0 = 1 (déduction à partir du bloc précédent)
- G71 est actif.
- Le rayon de l'arrondi doit être égal à 2 mm, l'avance à 50 mm/min.

Code de programme
...
N30 G1 Z… RND=2 FRC=50
N40 G3 X… Z… I… K…
...

Exemple 4 : arrondissement modal pour l'ébarbage des arêtes vives d'une pièce

Code de programme	**Commentaire**
...	
N30 G1 X… Z… RNDM=2 FRCM=50	; Activer l'arrondissement modal
	Rayon de l'arrondi : 2 mm
	Avance de l'arrondissement : 50 mm/min
N40...	
N120 RNDM=0	; Désactiver l'arrondissement modal
...	

Exemple 5 : appliquer la technologie du bloc suivant ou du bloc précédent

- PM20201 bit 0 = 0 : déduction à partir du bloc suivant (réglage par défaut !)

Code de programme	**Commentaire**
N10 G0 X0 Y0 G17 F100 G94	
N20 G1 X10 CHF=2	; Chanfrein N20-N30 avec F = 100 mm/min
N30 Y10 CHF=4	; Chanfrein N30-N40 avec FRC = 200 mm/min
N40 X20 CHF=3 FRC=200	; Chanfrein N40-N60 avec FRCM = 50 mm/min
N50 RNDM=2 FRCM=50	

Code de programme	**Commentaire**
N60 Y20	; Arrondi modal N60-N70 avec FRCM = 50 mm/min
N70 X30	; Arrondi modal N70-N80 avec FRCM = 50 mm/min
N80 Y30 CHF=3 FRC=100	; Chanfrein N80-N90 avec FRC = 100 mm/min
N90 X40	; Arrondi modal N90-N100 avec F = 100 mm/min (désactivation FRCM)
N100 Y40 FRCM=0	; Arrondi modal N100-N120 avec G95 FRC=1 mm/tr
N110 S1000 M3	
N120 X50 G95 F3 FRC=1	
...	
M02	

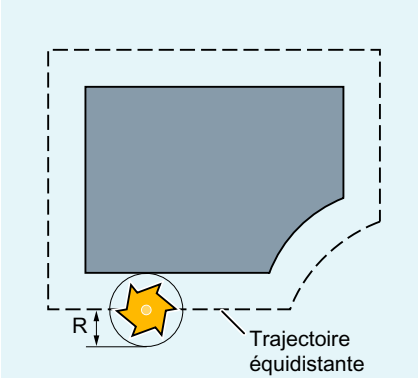
- PM20201 bit 0 = 1 : déduction à partir du bloc précédent (réglage recommandé !)

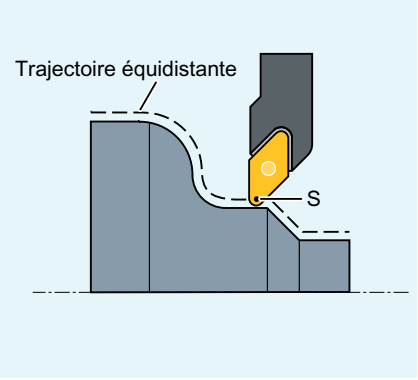
Code de programme	**Commentaire**
N10 G0 X0 Y0 G17 F100 G94	
N20 G1 X10 CHF=2	; Chanfrein N20-N30 avec F = 100 mm/min
N30 Y10 CHF=4 FRC=120	; Chanfrein N30-N40 avec FRC=120 mm/min
N40 X20 CHF=3 FRC=200	; Chanfrein N40-N60 avec FRC = 200 mm/min
N50 RNDM=2 FRCM=50	
N60 Y20	; Arrondi modal N60-N70 avec FRCM = 50 mm/min
N70 X30	; Arrondi modal N70-N80 avec FRCM = 50 mm/min
N80 Y30 CHF=3 FRC=100	; Chanfrein N80-N90 avec FRC = 100 mm/min
N90 X40	; Arrondi modal N90-N100 avec FRCM = 50 mm/min
N100 Y40 FRCM=0	; Arrondi modal N100-N120 avec F=100 mm/min
N110 S1000 M3	
N120 X50 CHF=4 G95 F3 FRC=1	; Chanfrein N120-N130 avec G95 FRC= 1 mm/tr
N130 Y50	; Arrondi modal N130-N140 avec F=3 mm/tr
N140 X60	
...	
M02	

2.10 Corrections de rayon d'outil

2.10.1 Correction du rayon d'outil (G40, G41, G42, OFFN)

Lorsque la correction du rayon d'outil est activée (CRO), la commande calcule automatiquement les trajectoires équidistantes des différents outils.





Syntaxe

`G0/G1 X... Y… Z...` **`G41/G42`** `[`**`OFFN=<valeur>`**`]`	
`...`	
`G40` `X... Y… Z...`	

Signification

`G41` :	Activer CRO avec sens d'usinage **vers la gauche** du contour
`G42` :	Activer CRO avec sens d'usinage **vers la droite** du contour
`OFFN=<valeur>` :	Surépaisseur d'usinage par rapport au contour programmé (offset normal par rapport au contour) (option) Par ex. afin de créer des trajectoires équidistantes pour la finition grossière.
`G40` :	Désactiver CRO

Remarque

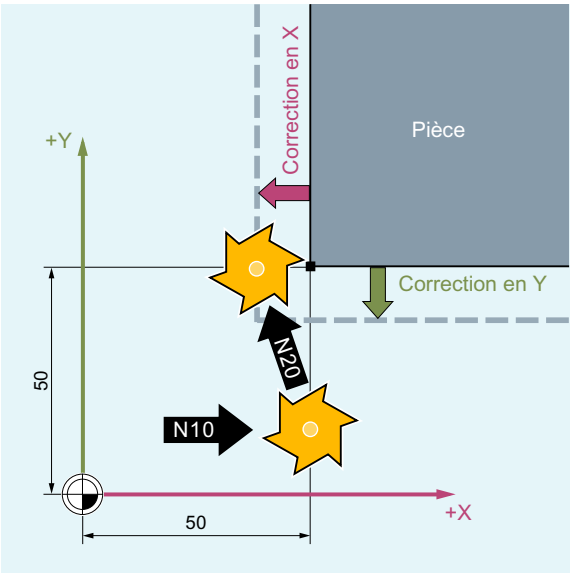
Dans le bloc CN avec `G40`/`G41`/`G42`, `G0` ou `G1` doivent être actifs et au moins un axe du plan de travail sélectionné doit être indiqué.

Si un seul axe est indiqué à l'activation, la dernière position du deuxième axe est complétée automatiquement et le déplacement s'effectue dans les **deux** axes.

Les deux axes doivent être actifs comme axes géométriques dans le canal. Ceci peut être réalisé par programmation de `GEOAX`.

Exemples

Exemple 1 : fraisage

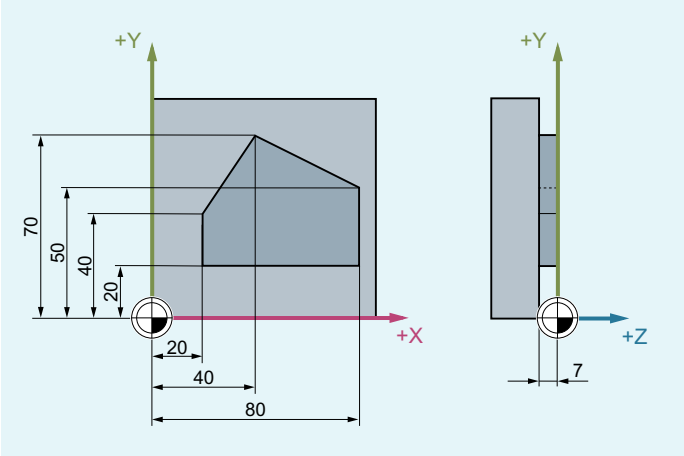


Code de programme	Commentaire
N10 G0 X50 T1 D1	; Seule la correction de longueur de l'outil s'affiche X50 est accosté sans correction.
N20 G1 G41 Y50 F200	; La correction du rayon est activée, le point X50/Y50 est accosté avec correction
N30 Y100	
…	

Exemple 2 : marche à suivre "classique" illustrée à l'exemple du fraisage

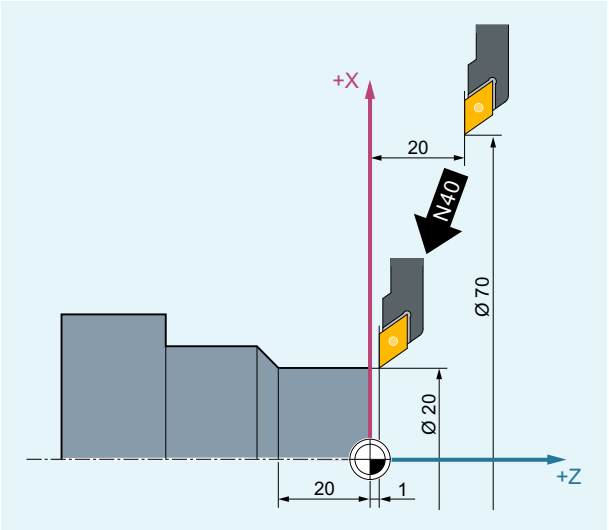
Marche à suivre "classique" :

1. Appel de l'outil.
2. Mise en place de l'outil.
3. Activation du plan de travail et de la correction du rayon d'outil.



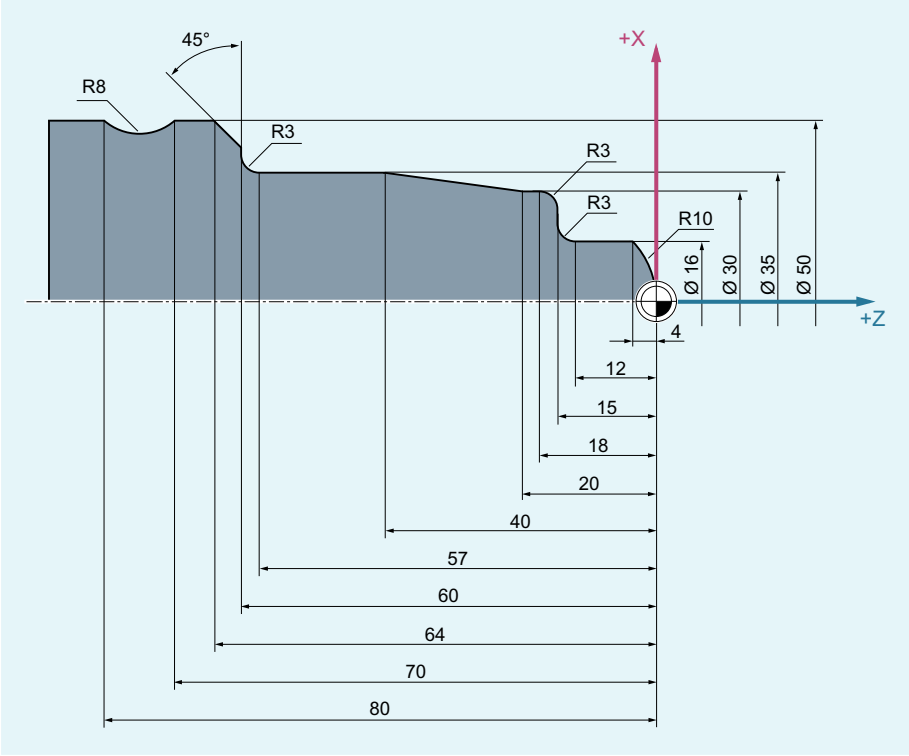
Code de programme	Commentaire
N10 G0 Z100	; Dégagement pour le changement d'outil
N20 G17 T1 M6	; Changement d'outil
N30 G0 X0 Y0 Z1 M3 S300 D1	; Appel des valeurs de correction d'outil, sélection de la correction de longueur
N40 Z-7 F500	; Prise de passe de l'outil
N50 G41 X20 Y20	; Activation de la correction du rayon d'outil, l'outil travaille à gauche du contour
N60 Y40	; Fraisage du contour
N70 X40 Y70	
N80 X80 Y50	
N90 Y20	
N100 X20	
N110 G40 G0 Z100 M30	; Dégagement de l'outil, fin du programme

Exemple 3 : tournage



Code de programme	Commentaire
…	
N20 T1 D1	; Seule la correction de longueur de l'outil s'affiche
N30 G0 X70 Z20	; X70 Z20 est accosté sans correction.
N40 G42 X20 Z1	; La correction du rayon est activée, le point X20/Z1 est accosté avec correction
N50 G1 Z-20 F0.2	
…	

Exemple 4 : tournage



Code de programme	Commentaire
N5 G0 G53 X280 Z380 D0	; Point de départ
N10 TRANS X0 Z250	; Décalage d'origine
N15 LIMS=4000	; Limitation de la vitesse de rotation (G96)
N20 G96 S250 M3	; Sélection de l'avance constante
N25 G90 T1 D1 M8	; Sélection de l'outil et du correcteur
N30 G0 G42 X-1.5 Z1	; Activation de la correction du rayon d'outil et approche de l'outil
N35 G1 X0 Z0 F0.25	
N40 G3 X16 Z-4 I0 K-10	; Tournage du rayon 10
N45 G1 Z-12	

Code de programme	**Commentaire**
N50 G2 X22 Z-15 CR=3	; Tournage du rayon 3
N55 G1 X24	
N60 G3 X30 Z-18 I0 K-3	; Tournage du rayon 3
N65 G1 Z-20	
N70 X35 Z-40	
N75 Z-57	
N80 G2 X41 Z-60 CR=3	; Tournage du rayon 3
N85 G1 X46	
N90 X52 Z-63	
N95 G0 G40 G97 X100 Z50 M9	; Désactivation de la correction du rayon d'outil et accostage de point de changement d'outil
N100 T2 D2	; Appel de l'outil et sélection du correcteur
N105 G96 S210 M3	; Sélection de la vitesse de coupe constante
N110 G0 G42 X50 Z-60 M8	; Activation de la correction du rayon d'outil et approche de l'outil
N115 G1 Z-70 F0.12	; Tournage du diamètre 50
N120 G2 X50 Z-80 I6.245 K-5	; Tournage du rayon 8
N125 G0 G40 X100 Z50 M9	; Relève de l'outil et désactivation de la correction du rayon d'outil
N130 G0 G53 X280 Z380 D0 M5	; Accostage du point de changement d'outil
N135 M30	; Fin du programme

Informations complémentaires

Calcul des trajectoires des outils

Pour le calcul des trajectoires des outils, la commande a besoin des informations suivantes :

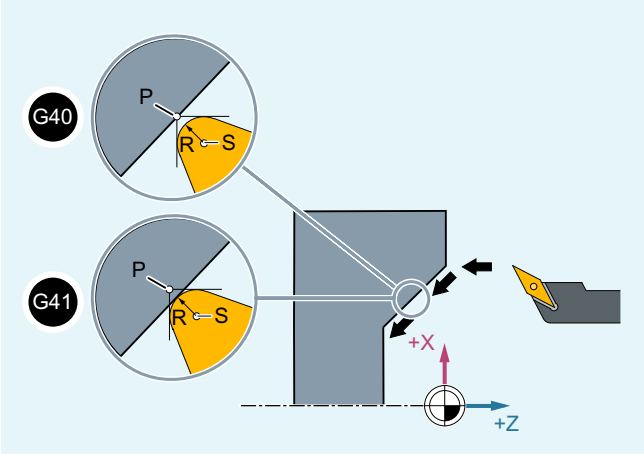
- Numéro d'outil (T...), numéro de tranchant (D...)
- Sens d'usinage (G41/G42)
- Plan de travail (G17/G18/G19)

Numéro d'outil (T...), numéro de tranchant (D...)

La distance entre la trajectoire de l'outil et le contour de la pièce est calculée à partir des rayons de fraise ou de tranchant et des indications relatives à la position du tranchant.

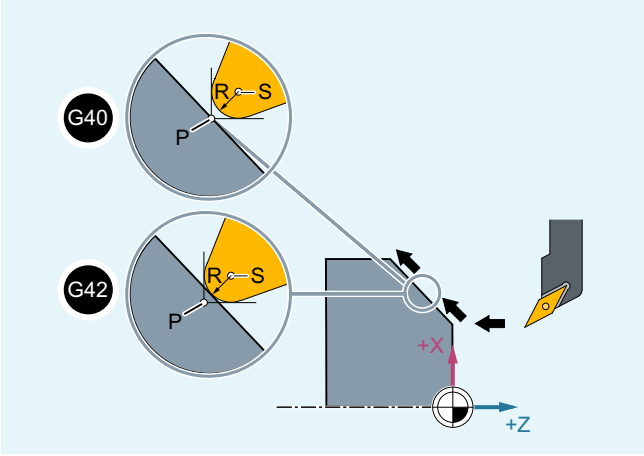
Sens d'usinage (G41/G42)

La commande reconnaît ainsi le sens dans lequel la trajectoire de l'outil est à décaler.



P Pointe de l'outil
S Centre du tranchant
R Rayon du tranchant

Figure 2-14 Correction du rayon d'outil pour le tournage avec sens d'usinage vers la gauche du contour (G41)



P Pointe de l'outil
S Centre du tranchant
R Rayon du tranchant

Figure 2-15 Correction du rayon d'outil pour le tournage avec sens d'usinage vers la droite du contour (G42)

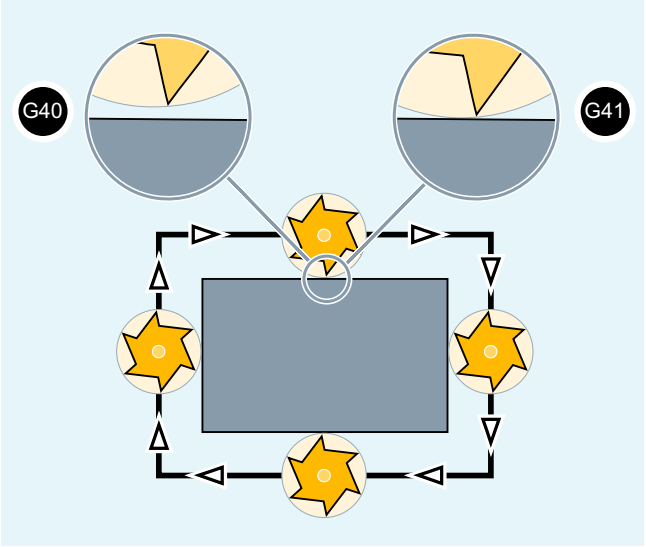


Figure 2-16 Correction du rayon d'outil pour le fraisage avec sens d'usinage vers la gauche du contour (G41)

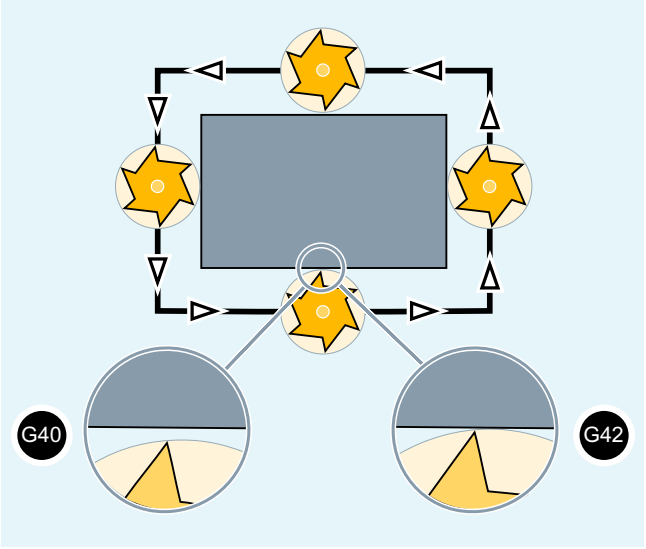


Figure 2-17 Correction du rayon d'outil pour le fraisage avec sens d'usinage vers la droite du contour (G42)

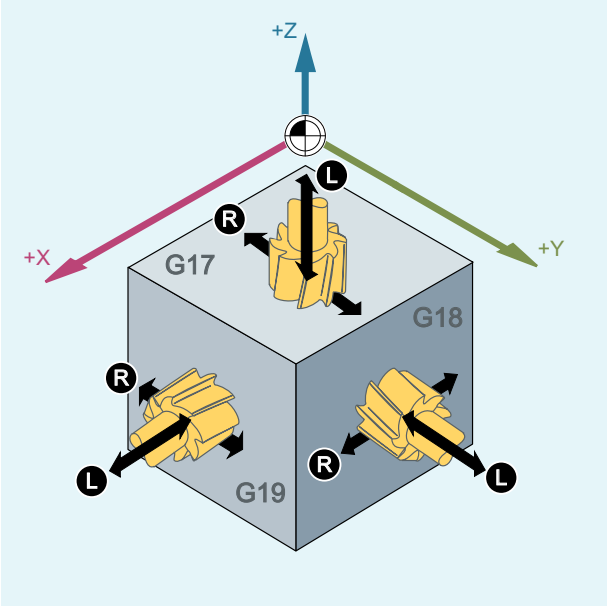
Remarque

Une valeur de correction négative correspond à une permutation du côté de la correction (G41 ↔ G42).

Plan de travail (G17/G18/G19)

La commande reconnaît ainsi le plan de travail et, donc, les directions d'axe dans lesquelles se feront les corrections.

Exemple : Outil de fraisage



L Longueur
R Rayon

Code de programme	Commentaire
...	
N10 G17 G41 …	; La correction du rayon d'outil est effectuée dans le plan X/Y et la correction de longueur d'outil dans la direction Z.
...	

Remarque

Sur les machines à 2 axes, la correction du rayon d'outil est uniquement possible dans de "vrais" plans, généralement sous G18.

Correction de longueur d'outil

Le paramètre d'usure affecté à l'axe du diamètre lors de l'appel de l'outil peut être défini en tant que valeur de diamètre au moyen d'un paramètre machine. Cette affectation n'est pas modifiée automatiquement en cas de changement ultérieur de plan. À cet effet, l'outil doit faire l'objet d'un nouvel appel après le changement de plan.

Comportement de la correction du rayon d'outil pour contour fermé

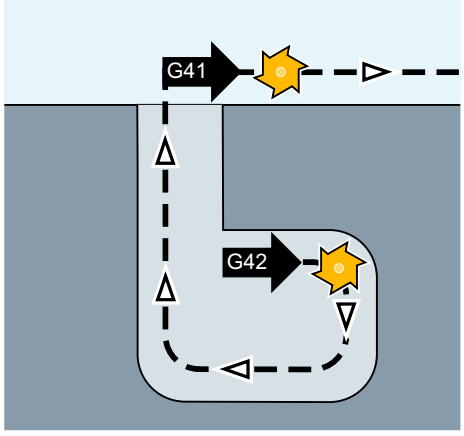
Le comportement de la correction du rayon d'outil pour contour fermé peut être réglé via la donnée de réglage :

SD42496 $SC_CUTCOM_CLSD_CONT

Valeur	Signification
FALSE	Si, sur un contour (presque) fermé composé de deux blocs circulaires successifs ou d'un bloc circulaire et d'un bloc linéaire et en cas de correction de la surface intérieure, deux points d'intersection se manifestent, alors le point d'intersection est sélectionné en fonction des indications standard, c'est-à-dire celui qui se trouve le plus proche de la fin de bloc sur le premier contour de pièce. Un contour est considéré comme étant (presque) fermé lorsque la distance entre le point de départ du premier bloc et le point final du deuxième bloc est inférieure à 10 % du rayon de correction appliqué, toutefois sans être supérieure à 1000 incréments (ce qui correspond à 1 mm pour trois chiffres après la virgule).
TRUE	Dans la même situation que celle décrite plus haut, le point d'intersection sélectionné est celui qui se trouve le plus proche du début de bloc sur le premier contour de pièce.

Changement du sens de correction (G41 ↔ G42)

Un changement du sens de correction (G41 ↔ G42) peut être programmé sans connexion intermédiaire de G40.



Changement du plan de travail

Un changement du plan de travail (G17/G18/G19) n'est **pas** possible si G41/G42 est activé.

Changement du bloc de données de correction d'outil (D...)

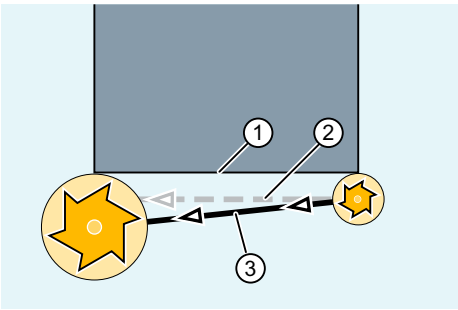
Le bloc de données de correction d'outil peut être changé en mode correction.

La modification du rayon d'outil est opérante à partir du bloc dans lequel figure le nouveau numéro D.

Remarque

La modification de rayon, à savoir le déplacement de compensation, s'effectue pendant toute la durée du bloc et le nouvel écart équidistant n'est atteint qu'au point final programmé.

Dans le cas d'un déplacement linéaire, l'outil décrit une trajectoire oblique entre le point de départ et le point final :



① Contour programmé
② Trajectoire d'outil sans changement du bloc de données de correction d'outil dans le bloc actif
③ Trajectoire d'outil lors d'un changement du bloc de données de correction d'outil dans le bloc actif et d'une modification associée du rayon d'outil

Dans le cas d'une interpolation circulaire, il décrit des mouvements en spirales.

Modification du rayon d'outil

La modification peut être réalisée p. ex. au moyen de variables système. La procédure est la même que pour le changement du bloc de données de correction d'outil (D...).

Remarque

Les valeurs modifiées ne prennent effet qu'après la reprogrammation de T ou de D. La modification n'agit qu'à partir du bloc suivant.

Mode correction

Le mode correction ne peut être interrompu que par un nombre donné de blocs ou d'instructions M successifs ne contenant ni instruction de déplacement, ni indication de distance dans le plan de correction.

Remarque

Le nombre maximal de blocs d'interruption ou d'instructions M successifs est réglable à l'aide d'un paramètre machine (voir les indications du constructeur de machines).

Remarque

Un bloc contenant un déplacement nul est également considéré comme une interruption !

Voir aussi

Accostage et retrait du contour (NORM, KONT, KONTC, KONTT) (Page 265)

2.10.2 Accostage et retrait du contour (NORM, KONT, KONTC, KONTT)

Condition

Les instructions KONTC et KONTT sont uniquement disponibles si l'option "Interpolation d'un polynôme" est validée dans la commande.

Fonction

Les instructions NORM, KONT, KONTC ou KONTT permettent d'adapter le mouvement d'accostage et de retrait de l'outil à la trajectoire du contour souhaitée ou à la forme de la pièce brute, lorsque la correction du rayon d'outil est activée (G41/G42).

KONTC ou KONTT permettent de remplir les conditions de continuité sur les trois axes. Il est alors possible de programmer simultanément une composante de trajectoire perpendiculairement au plan de correction.

Syntaxe

G41/G42 **NORM**/**KONT**/**KONTC**/**KONTT** X... Y... Z...	
...	
G40 X... Y... Z...	

Signification

NORM :	Activer l'accostage/le retrait direct sur une droite L'outil est positionné perpendiculairement au point du contour.
KONT :	Accostage/retrait avec contournement du point du départ/point final, selon le comportement G450 ou G451 programmé pour les angles
KONTC :	Activer l'accostage/le retrait à courbure continue
KONTT :	Activer l'accostage/le retrait tangentiel

Remarque

Seuls des blocs G1 sont autorisés comme blocs d'accostage et de retrait originaux pour KONTC et KONTT. La commande remplace ces blocs par des polynômes pour les trajectoires d'accostage et de retrait correspondantes.

Autres conditions à prendre en compte

KONTT et KONTC ne sont pas disponibles pour les variantes 3D de la correction du rayon d'outil (CUT3DC, CUT3DCC, CUT3DF). Si elles sont malgré tout programmées, une commutation sur NORM a lieu dans la commande, sans message d'erreur.

Exemple

KONTC

En partant du centre du cercle, le cercle complet est accosté. Dans le point final du bloc d'accostage, la direction et le rayon de courbure sont égaux aux valeurs du cercle suivant. Dans

les deux blocs d'accostage et de retrait, la pénétration en Z s'effectue simultanément. La figure suivante montre la projection orthogonale de la trajectoire de l'outil :

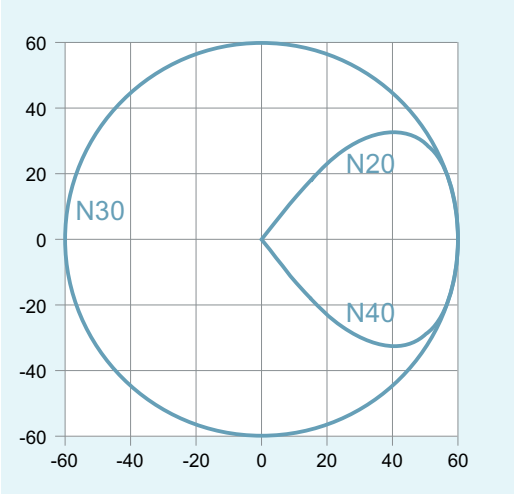


Figure 2-18 Projection orthogonale

Le segment de programme CN correspondant ressemble à ceci :

Code de programme	**Commentaire**
`$TC_DP1[1,1]=121`	`; Fraise`
`$TC_DP6[1,1]=10`	`; Rayon 10 mm`
`N10 G1 X0 Y0 Z60 G64 T1 D1 F10000`	
`N20 G41 KONTC X70 Y0 Z0`	`; Accostage`
`N30 G2 I-70`	`; Cercle complet`
`N40 G40 G1 X0 Y0 Z60`	`; Retrait`
`N50 M30`	

Parallèlement à l'adaptation de la courbure à la trajectoire circulaire du cercle complet, un déplacement a lieu de Z60 au plan du cercle Z0 :

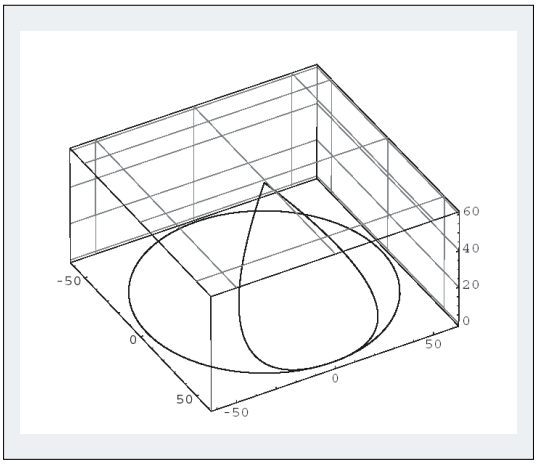
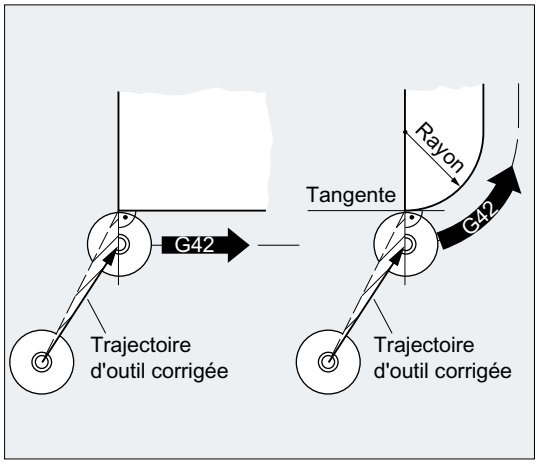


Figure 2-19 Représentation dans l'espace

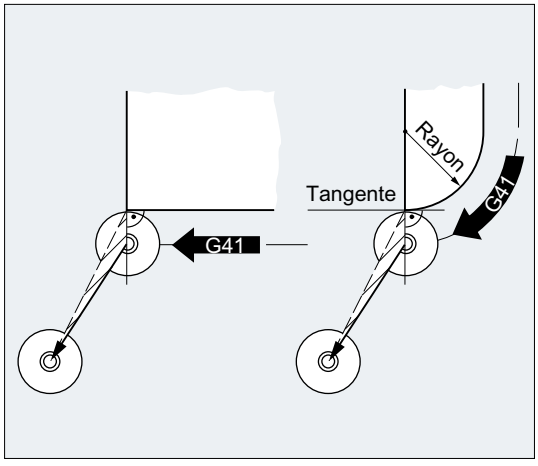
Informations complémentaires

Accostage/retrait avec NORM

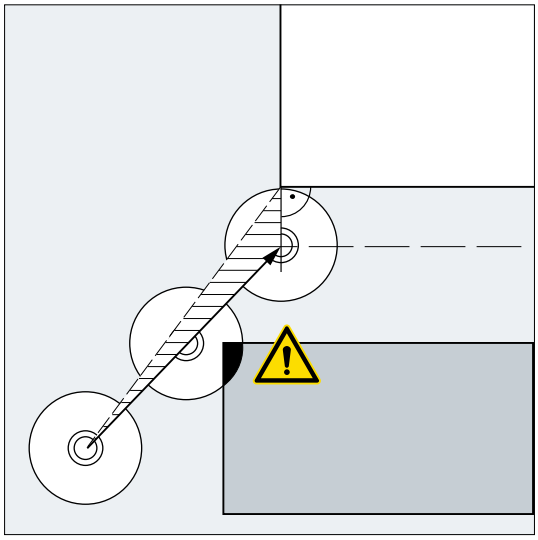
1. Accostage :
   Lorsque NORM est activé, l'outil rejoint directement, sur une droite, la position de départ corrigée (quel que soit l'angle d'accostage prédéfini par le déplacement programmé) et se positionne perpendiculairement à la tangente de la trajectoire au point de départ :



2. Retrait :
   L'outil est sur la normale au dernier point final corrigé de la trajectoire et rejoint ensuite directement (quel que soit l'angle d'accostage prédéfini par le déplacement programmé), sur une droite, la position suivante non corrigée, p. ex. le point de changement d'outil :



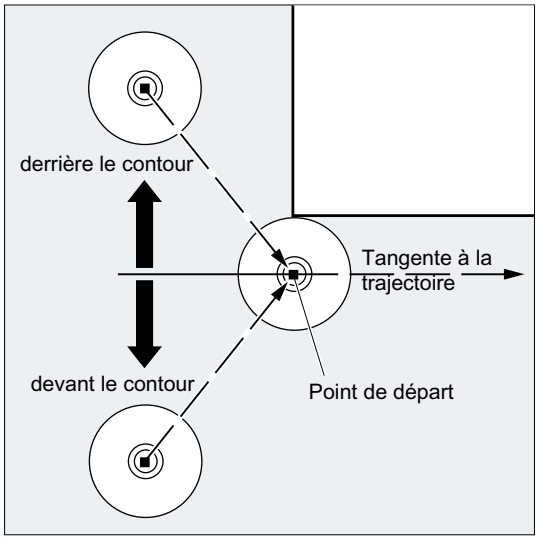
Une modification de l'angle d'accostage/retrait constitue un risque de collision :



IMPORTANT
Risque de collision Pour éviter d'éventuelles collisions, les angles d'accostage / de retrait modifiés doivent être pris en compte lors de la programmation.

Accostage/retrait avec KONT

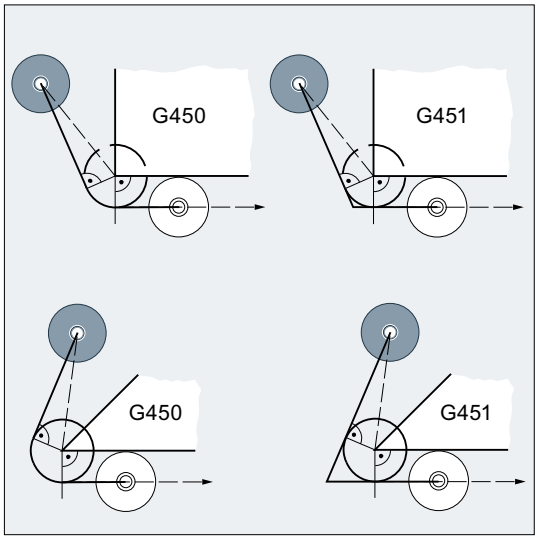
Avant l'accostage, l'outil peut être positionné **devant** ou **derrière** le contour. La tangente à la trajectoire au point de départ constitue la ligne séparatrice :



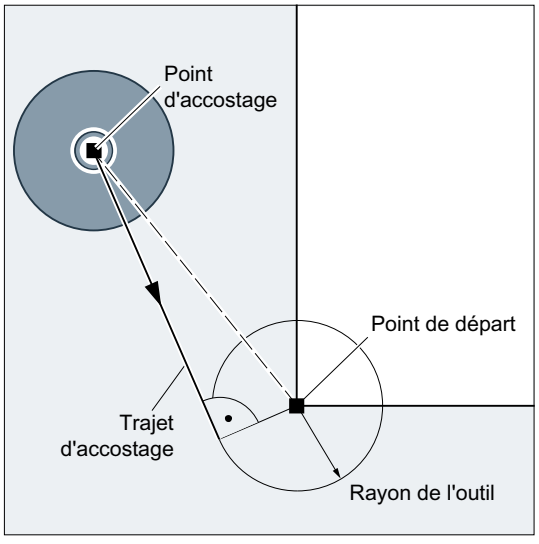
En cas d'accostage/retrait avec KONT, il convient de distinguer deux cas :

1. L'outil est positionné devant le contour.
   → Même stratégie d'accostage/retrait qu'avec NORM.
2. L'outil est positionné derrière le contour

   – Accostage :
   L'outil contourne le point de départ sur une trajectoire circulaire ou en passant par le point d'intersection des équidistantes, selon le comportement programmé pour les angles (G450/G451).
   Les fonctions G450/G451 s'appliquent à la transition entre bloc courant et bloc suivant.



   Dans les deux cas (G450/G451), le trajet suivant est généré :



   Une tangente à un cercle dont le rayon est égal au rayon d'outil est tracée à partir du point d'accostage non corrigé. Le centre du cercle se situe au point de départ.

   – Retrait :
   Pour le retrait, la procédure est la même que pour l'accostage, mais en sens inverse.

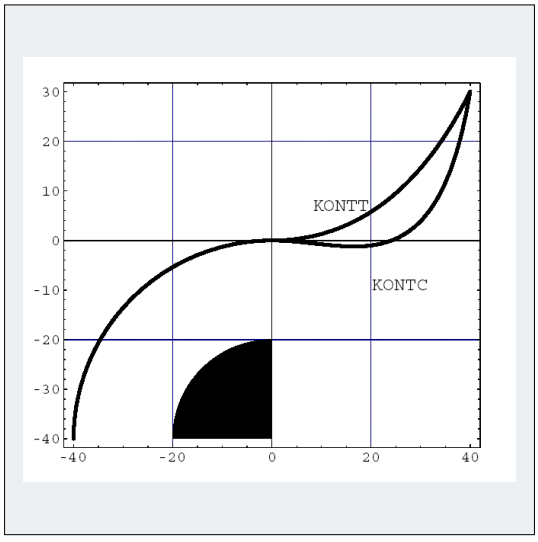
Accostage/retrait avec KONTC

L'accostage/le retrait du point de contour s'effectue avec une courbure continue. Sur le point de contour, aucun saut d'accélération n'a lieu. La trajectoire du point d'attaque au point de contour est interpolée comme polynôme.

Accostage/retrait avec KONTT

L'accostage/le retrait du point de contour s'effectue avec une tangente continue. Sur le point de contour, un saut d'accélération peut avoir lieu. La trajectoire du point d'attaque au point de contour est interpolée comme polynôme.

Différence entre KONTC et KONTT

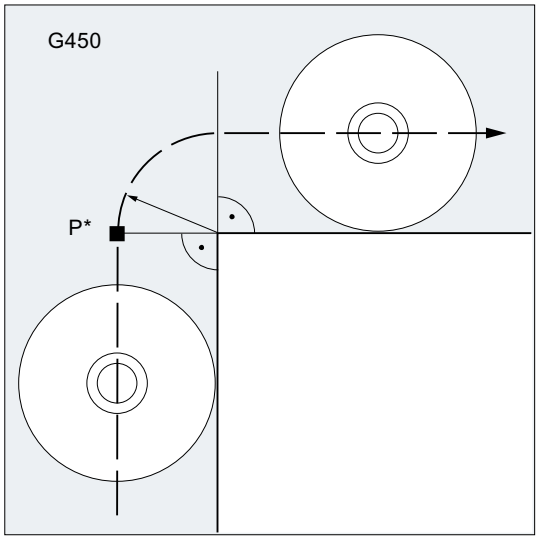


Dans cette figure sont représentés les différents comportements à l'accostage et au retrait pour `KONTT` et `KONTC`. Un cercle avec un rayon de 20 mm autour du centre avec X0 Y-40 est corrigé avec un rayon de 20 mm sur la face extérieure. On obtient ainsi un déplacement circulaire du centre de l'outil avec un rayon de 40 mm. Le point final du bloc de retrait se trouve sur X40 Y30. La transition entre le bloc circulaire et le bloc de retrait se trouve su l'origine. Du fait du rallongement de la courbure continue pour `KONTC` , le bloc de retrait exécute tout d'abord un déplacement avec une composante Y négative. Ceci ne sera souvent pas souhaité. Le bloc de retrait avec `KONTT` ne présente pas ces comportements. Mais dans ce cas, un saut d'accélération a lieu lors de la transition entre les blocs.

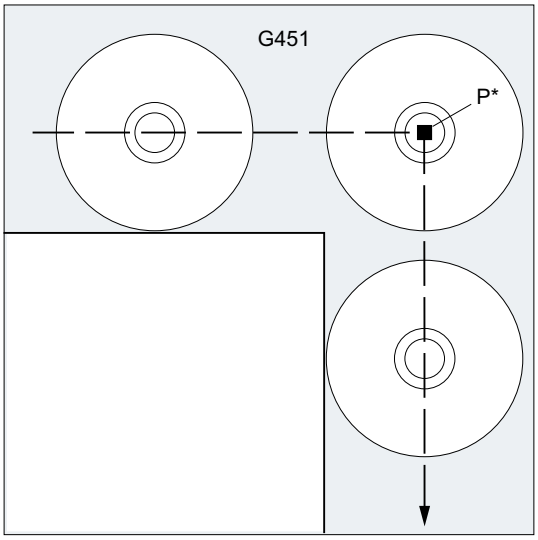
Si le bloc `KONTT` ou `KONTC` n'est pas le bloc de retrait mais le bloc d'accostage, le contour reste exactement le même mais sera seulement parcouru en sens inverse.

2.10.3 Correction aux angles saillants (G450, G451, DISC)

Les fonctions G450 ou G451 définissent la trajectoire corrigée de l'outil pour le contournement des angles saillants lorsque la correction du rayon d'outil est activée (G41/G42) :



Avec G450, le centre de l'outil contourne l'angle de la pièce sur un arc de cercle dont le rayon est celui de l'outil.



Avec G451, le centre de l'outil accoste le point d'intersection des deux équidistantes, dont la distance au contour programmé est égale au rayon de l'outil. G451 s'applique uniquement à des droites et des cercles.

Remarque

G450/G451 permettent également de définir le trajet d'accostage lorsque KONT est actif et que le point d'accostage se trouve derrière le contour (voir "Accostage et retrait du contour (NORM, KONT, KONTC, KONTT) (Page 265)").

L'instruction DISC permet de déformer les arcs de raccordement pour G450et ainsi de créer des angles de contour vifs.

Syntaxe

```
G450 [DISC=<valeur>]
G451
```

Signification

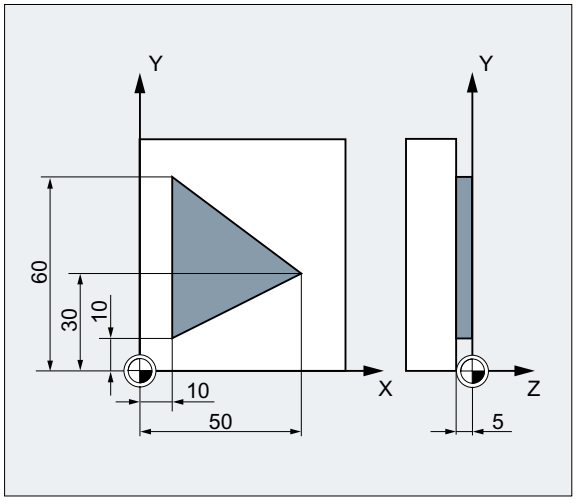
G450 :	G450 permet de contourner les angles de la pièce sur une trajectoire circulaire.			
DISC :	Programmation flexible de la trajectoire circulaire pour G450 (option)			
	<valeur> :	Type :	INT	
		Plage de valeurs :	0, 1, 2, ... 100	
		Signification :	0	arc de raccordement
			100	Point d'intersection des équidistantes (valeur théorique)
G451 :	G451 permet d'accoster le point d'intersection des deux équidistantes des angles de la pièce. L'outil est dégagé aux angles de la pièce.			

Remarque

DISC agit uniquement avec l'appel de G450, mais peut toutefois être programmé sans G450 dans un bloc antérieur. Les deux instructions sont modales.

Exemple

Dans l'exemple suivant, on insère un rayon de raccordement à tous les angles saillants (selon la programmation du comportement aux angles définie dans le bloc N30). Ceci permet d'éviter le dégagement de l'outil pour un changement de direction.



Code de programme	**Commentaire**
N10 G17 T1 G0 X35 Y0 Z0 F500	; Conditions de départ
N20 G1 Z-5	; Prise de passe de l'outil
N30 G41 KONT **G450** X10 Y10	; **Activation** de la CRO avec mode d'accostage / de retrait KONT et **comportement aux angles G450**
N40 Y60	; Fraisage du contour
N50 X50 Y30	
N60 X10 Y10	

Code de programme	Commentaire
N80 G40 X-20 Y50	; Désactivation du mode de correction, retrait sur arc de raccordement
N90 G0 Y100	
N100 X200 M30	

Informations complémentaires

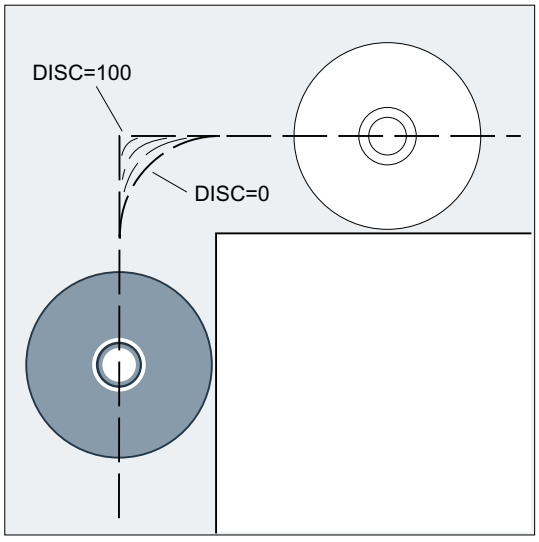
G450/G451

Au point intermédiaire P*, la commande exécute des instructions, comme par exemple des mouvements de pénétration ou des fonctions de commutation. Ces instructions sont programmées dans des blocs situés entre les deux blocs formant l'angle.

L'arc de raccordement pour G450 fait partie de l'instruction de déplacement suivante.

DISC

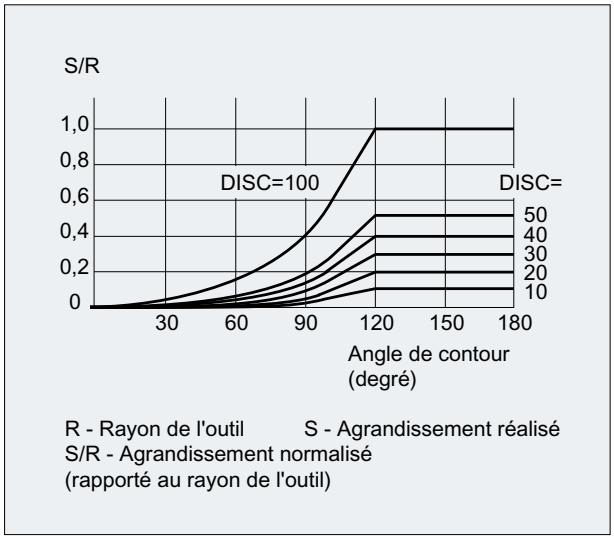
En indiquant des valeurs DISC supérieures à 0, vous augmentez la taille des arcs de raccordement, qui deviennent des ellipses de raccordement, des paraboles ou des hyperboles :



Avec un paramètre machine, on peut définir une valeur limite supérieure, en règle générale DISC=50.

Comportement de déplacement

Lorsque G450 est activé, l'outil décroche du contour quand les angles sont aigus et les valeurs DISC élevées. Quand les angles du contour sont supérieurs ou égaux à 120°, le contournement du contour est régulier :

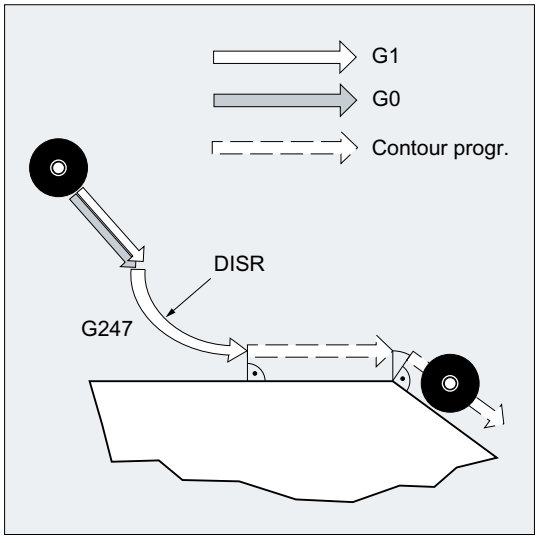


Lorsque `G451` est activé, des déplacements dans le vide superflus de l'outil peuvent se produire en raison de mouvements de décrochement, quand les angles de contour sont aigus. Avec un paramètre machine, on peut spécifier que, dans de tels cas, il y a passage automatique à l'arc de raccordement.

2.10.4 Accostage et retrait en douceur

2.10.4.1 Accostage et retrait (G140 à G143, G147, G148, G247, G248, G347, G348, G340, G341, DISR, DISCL, DISRP, FAD, PM, PR)

La fonction "Accostage et retrait en douceur (ARD)" sert à l'accostage tangentiel d'un contour au point de départ, indépendamment de la position du point d'origine.



La fonction s'utilise principalement en liaison avec la correction du rayon d'outil.

Lors de l'activation de la fonction, la commande prend en charge le calcul des points intermédiaires de telle façon que la transition au bloc suivant (ou la transition depuis le bloc précédent dans le cas du retrait) est effectuée conformément aux paramètres indiqués.

Le mouvement d'accostage se décompose au maximum en 4 mouvements partiels. Le point de départ du déplacement est désigné dans la suite avec P_0, la point final avec P_4. Entre les deux peuvent se trouver maximum trois points intermédiaires P_1, P_2 et P_3. Les points P_0, P_3 et P_4 sont toujours définis. Les points intermédiaires P_1 et P_2 peuvent être omis, selon le paramétrage et la géométrie. Lors du retrait, les points sont passés dans le sens inverse, c.-à-d. en commençant par P_4 pour finir avec P_0.

Syntaxe

Accostage en douceur :

- sur une droite :
  `G147 G340/G341 ... DISR=..., DISCL=..., DISRP=... FAD=...`
- sur un quart de cercle / demi-cercle :
  `G247/G347 G340/G341 G140/G141/G142/G143 ... DISR=... DISCL=... DISRP=... FAD=...`

Retrait en douceur :

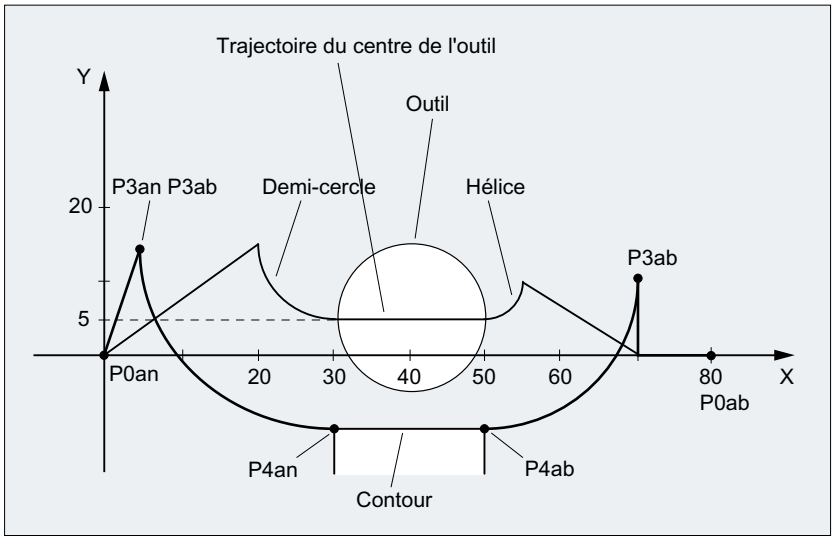
- sur une droite :
  `G148 G340/G341 ... DISR=..., DISCL=..., DISRP=... FAD=...`
- sur un quart de cercle / demi-cercle :
  `G248/G348 G340/G341 G140/G141/G142/G143 ... DISR=... DISCL=... DISRP=... FAD=...`

Signification

`G147` :	Accostage en ligne droite
`G148` :	Retrait en ligne droite
`G247` :	Accostage en quart de cercle
`G248` :	Retrait en quart de cercle
`G347` :	Accostage en demi cercle
`G348` :	Retrait en demi cercle
`G340` :	Accostage et retrait dans l'espace (réglage de base)
`G341` :	Accostage et retrait dans le plan
`G140` :	Sens de l'accostage et du retrait en fonction du côté où se fait la correction (réglage de base)
`G141` :	Accostage depuis la gauche ou retrait vers la gauche
`G142` :	Accostage depuis la droite ou retrait vers la droite
`G143` :	Sens de l'accostage et du retrait en fonction de la position relative du point de départ ou du point final par rapport à la direction de la tangente.

`DISR=...:`	1. Lors de l'accostage et retrait en ligne droite (G147/G148) : Distance entre bord de fraise et point de départ du contour 2. Lors de l'accostage et retrait en cercle (G247, G347/G248, G348) : rayon de la trajectoire du centre de l'outil **Important :** Dans le cas de REPOS avec un demi-cercle, DISR définit le diamètre du cercle
`DISCL=...:`	Distance entre le point final du mouvement de pénétration rapide et le plan d'usinage DISCL=AC(...) Indication de la position absolue du point final du mouvement d'approche rapide
`DISCL=AC(...):`	Indication de la position absolue du point final du mouvement d'approche rapide
`DISRP:`	Distance entre le point P1 (plan de retrait) et le plan d'usinage
`DISRP=AC(...):`	Indication de la position absolue du point P1
`FAD=...:`	Vitesse du mouvement d'approche lente La valeur programmée agit en fonction du type d'avance actif (groupe G 15).
`FAD=PM(...):`	La valeur programmée est interprétée comme avance linéaire (comme G94) indépendamment du type d'avance actif.
`FAD=PR(...):`	La valeur programmée est interprétée comme avance par tour (comme G95) indépendamment du type d'avance actif.

Exemple



- Accostage en douceur (bloc N20 activé)
- Mouvement d'accostage en quart de cercle (G247)
- Sens d'accostage non programmé, c'est G140 qui entre en vigueur, autrement dit la CRO est active (G41)
- Offset de contour OFFN=5 (N10)

- Rayon d'outil courant=10 pour que le rayon de correction effectif du rayon soit CRO=15 et le rayon du contour ARD=25 ; ainsi le rayon de la trajectoire du centre de l'outil devient égal à DISR=10
- Le point final du cercle résulte de N30, puisque seule la position Z est programmée dans N20
- Mouvement de pénétration
  - De Z20 vers Z7 (DISCL=AC(7)) en vitesse rapide.
  - Puis vers Z0 avec FAD=200.
  - Accostage en arc de cercle dans le plan X-Y et blocs suivants avec F1500 (afin que cette vitesse devienne active dans les blocs suivants, la fonction G0 active doit être écrasée dans N30 avec G1 sinon l'usinage du contour continue avec G0).
- Accostage en douceur (bloc N60 activé)
- Mouvement de retrait en quart de cercle (G248) et hélice (G340)
- FAD n'est pas programmé, car sans importance dans G340
- Z=2 au point de départ ; Z=8 au point final, puisque DISCL=6
- Avec DISR=5, le rayon du contour ARD=20, le rayon de la trajectoire du centre de l'outil=5

Mouvements de retrait de Z8 vers Z20 et mouvement parallèle au plan X-Y vers X70 Y0.

Code de programme	**Commentaire**
`$TC_DP1[1,1]=120`	`; Définition de l'outil T1/D1`
`$TC_DP6[1,1]=10`	`; Rayon`
`N10 G0 X0 Y0 Z20 G64 D1 T1 OFFN=5`	`; (acc P0)`
`N20 G41 G247 G341 Z0 DISCL=AC(7) DISR=10 F1500 FAD=200`	`; Accostage (acc P3)`
`N30 G1 X30 Y-10`	`; (acc P4)`
`N40 X40 Z2`	
`N50 X50`	`; (ret P4)`
`N60 G248 G340 X70 Y0 Z20 DISCL=6 DISR=5 G40 F10000`	`; Retrait (ret P3)`
`N70 X80 Y0`	`; (ret P0)`
`N80 M30`	

Informations complémentaires

Choix de la trajectoire d'accostage ou de retrait

Le choix du contour d'accostage ou de retrait s'effectue avec l'instruction G correspondante du 2e groupe G :

G147 :	Accostage en ligne droite
G247 :	Accostage en quart de cercle
G347 :	Accostage en demi cercle
G148 :	Retrait en ligne droite
G248 :	Retrait en quart de cercle
G348 :	Retrait en demi cercle

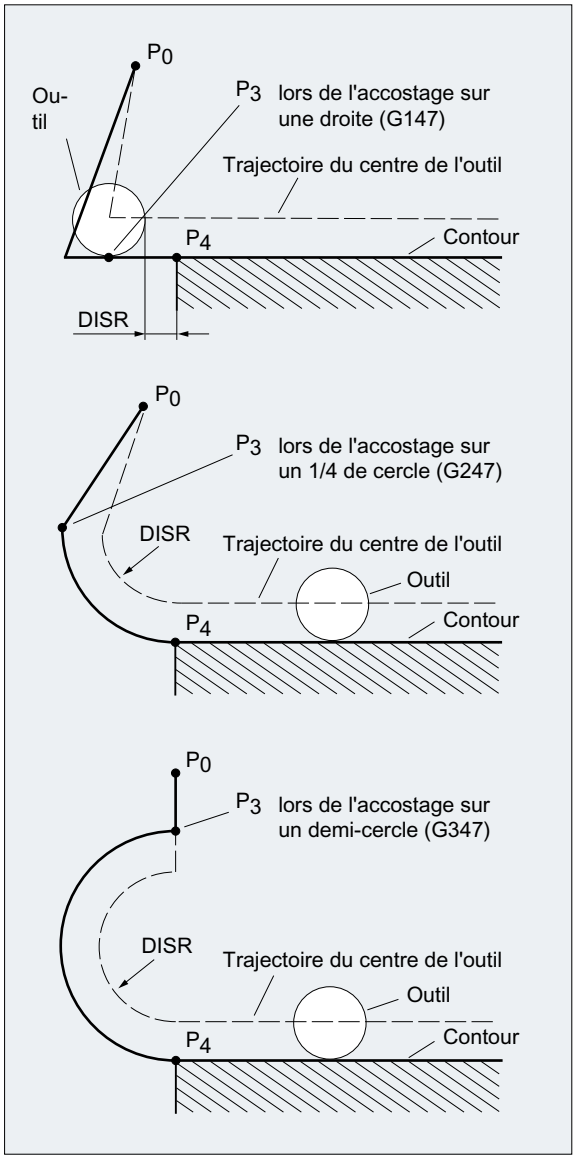


Figure 2-20 Mouvements d'accostage avec activation simultanée de la correction du rayon d'outil

Choix du sens d'accostage ou de retrait

Détermination du sens d'accostage ou de retrait par le biais de la correction de rayon d'outil (G140, réglage de base) pour un rayon d'outil positif :

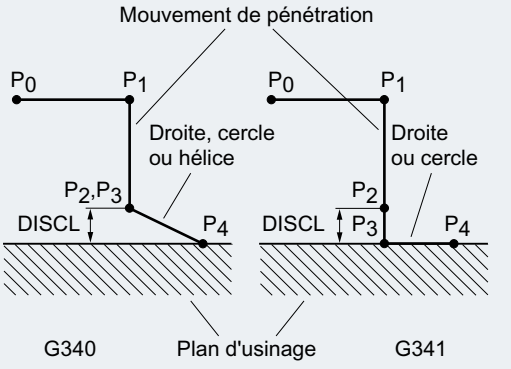
- G41 actif → accostage depuis la gauche
- G42 actif → accostage depuis la droite

D'autres possibilités d'accostage sont fournies avec G141, G142 et G143.

Ces instructions G ont seulement une importance quand l'accostage du contour se fait en arc de cercle ou en demi-cercle.

Décomposition du mouvement du point de départ au point final (G340 et G341)

Les mouvements se composent en tous les cas d'une ou de plusieurs lignes droites ainsi que, selon l'instruction G pour définir le contour d'accostage, d'une autre ligne droite, d'un quart de cercle ou d'un demi-cercle. Les 2 variantes de segmentation des déplacements sont représentées dans la figure suivante :



Mouvement d'accostage en fonction de G340/G341

G340 :	Accostage en ligne droite du point P_0 au point P_1. Cette ligne droite est parallèle au plan d'usinage si le paramètre DISRP n'a pas été programmé. Approche perpendiculaire au plan d'usinage du point P_1 au point P_3 jusqu'à la distance de sécurité par rapport au plan d'usinage définie par le paramètre DISCL. Accostage du point final P4 avec la courbe (ligne droite, cercle, hélice) définie par l'instruction G du deuxième groupe. Si G247 ou G347 est actif (quart de cercle ou demi-cercle) et si le point de départ P_3 ne se trouve pas dans le plan d'usinage défini par le point final P_4, alors une hélice est insérée à la place d'un cercle. Le point P_2 n'est pas défini ou il se confond avec P_3. Le plan du cercle ou l'axe de l'hélice est alors déterminé par le plan actif dans le bloc d'accostage / de retrait en douceur (G17/G18/G19), c.-à-d. que le bloc suivant n'utilise pas la tangente de départ elle-même pour déterminer le cercle, mais plutôt sa projection dans le plan actif. Le mouvement du point P_0 au point P_3 est effectué en deux lignes droites avec la vitesse qui était effective avant le bloc d'accostage / de retrait en douceur.
G341 :	Accostage en ligne droite du point P_0 au point P_1. Cette ligne droite est parallèle au plan d'usinage si le paramètre DISRP n'a pas été programmé. Approche perpendiculaire au plan d'usinage du point P_1 jusqu'à la distance de sécurité par rapport au plan d'usinage définie par le paramètre DISCL au point P_2. Approche perpendiculaire au plan d'usinage du point P_2 au point P_3. Accostage du point final avec la courbe définie par l'instruction G du deuxième groupe. P_3 et P_4 se trouvent dans le plan d'usinage de sorte que, pour G247 ou G347, c'est toujours un cercle qui est inséré et non pas une hélice.

Dans tous les cas où la position du plan actif G17/G18/G19 entre en ligne de compte (plan du cercle, axe d'hélice, mouvements d'approche perpendiculaires au plan actif), un frame rotatif éventuellement actif est pris en considération.

Longueur de la droite d'accostage ou rayon dans le cas d'un accostage en arc de cercle (DISR)

- Accostage/retrait en ligne droite
  DISR fournit la distance du bord de fraise au point de départ du contour, autrement dit, quand CRO est active, la longueur de la droite est égale à la somme du rayon d'outil et de la valeur programmée de DISR. Le rayon de l'outil est uniquement pris en compte lorsqu'il est positif. La longueur de la droite résultante doit être positive, autrement dit, des valeurs négatives sont admises pour DISR, aussi longtemps que la valeur absolue de DISR est inférieure au rayon d'outil.
- Accostage/retrait en cercle
  DISR indique le rayon de la trajectoire du centre de la pièce. Quand la CRO est active, un cercle est généré avec un rayon tel que la trajectoire du centre de l'outil soit décrite suivant le rayon programmé.

Distance du point P2 par rapport au plan d'usinage (DISCL)

Si la position du point P_2 sur l'axe perpendiculaire au plan du cercle doit être indiquée en valeurs absolues, il convient de programmer la valeur sous la forme `DISCL=AC(...)`.

Pour DISCL=0 on a :

- Avec G340 : L'ensemble du mouvement d'accostage est uniquement composé de deux blocs (P_1, P_2 et P_3 coïncident). Le contour d'accostage est formé de P_1 à P_4.
- Avec G341 : L'ensemble du mouvement d'accostage est composé de trois blocs (P_2 et P_3 coïncident). Si P_0 et P_4 se trouvent dans le même plan, seuls deux blocs sont générés (le mouvement d'approche de P_1 vers P_3 disparaît).
- La commande vérifie que le point défini par DISCL se trouve entre P_1 et P_3, c.-à-d. que, dans le cas de tous les mouvements, les composantes perpendiculaires au plan d'usinage, si elles existent, doivent avoir le même signe.
- Lors de la détection d'un changement de sens, une tolérance définie dans le paramètre machine PM20204 $MC_WAB_CLEARANCE_TOLERANCE est prise en considération.

Distance entre le point P1 (plan de retrait) et le plan d'usinage (DISRP)

Si la position du point P_1 sur l'axe perpendiculaire au plan d'usinage doit être indiquée en valeurs absolues, il convient de programmer la valeur sous la forme `DISRP=AC(...)`.

Si ce paramètre n'est pas programmé, le point P_1 est à la même distance du plan d'usinage que le point P_0, c.-à-d. que la droite d'accostage $P_0 \rightarrow P_1$ est parallèle au plan d'usinage.

La commande vérifie que le point défini par DISRP se trouve entre P_0 et P_2 , c.-à-d. que, dans le cas de tous les mouvements ayant une composante perpendiculaire au plan d'usinage (mouvements de pénétration, mouvements d'accostage de P_3 à P_4), cette composante doit avoir le même signe. Une inversion du sens n'est pas autorisée. Le cas échéant, une alarme est générée.

Lors de la détection d'un changement de sens, une tolérance définie dans le paramètre machine PM20204 $MC_WAB_CLEARANCE_TOLERANCE est prise en considération. Si P_1 se trouve hors de la zone définie par P_0 et P_2, mais que l'écart est inférieur ou égal à cette tolérance, alors on admet que P_1 se trouve dans le plan défini par P_0 ou P_2.

Programmation du point final

En règle générale, le point final est programmé avec X... Y... Z...

La programmation du point final de contour pour l'accostage est très différente de celle pour le retrait. Pour cette raison, ces deux cas sont traités ici séparément.

Programmation du point final P4 pour l'accostage

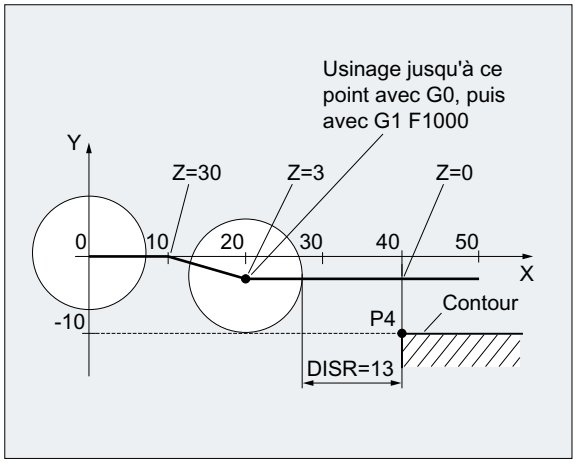
Le point final P_4 peut être programmé dans le bloc d'accostage / de retrait en douceur lui-même. Il est également possible de déterminer P_4 par le point final du bloc de déplacement suivant. Entre le bloc ARD et le bloc de déplacement suivant, il est possible d'intercaler des blocs supplémentaires sans déplacement des axes géométriques.

Exemple :

Code de programme	**Commentaire**
`$TC_DP1[1,1]=120`	`; Outil de fraisage T1/D1`
`$TC_DP6[1,1]=7`	`; Outil de 7 mm de rayon`
`N10 G90 G0 X0 Y0 Z30 D1 T1`	
`N20 X10`	
`N30 G41 G147 DISCL=3 DISR=13 Z=0 F1000`	
`N40 G1 X40 Y-10`	
`N50 G1 X50`	
`...`	

`N30/N40` peut être remplacé par :
```
N30 G41 G147 DISCL=3 DISR=13 X40 Y-10 Z0 F1000
```

ou
```
N30 G41 G147 DISCL=3 DISR=13 F1000
N40 G1 X40 Y-10 Z0
```



Programmation du point final P0 pour le retrait

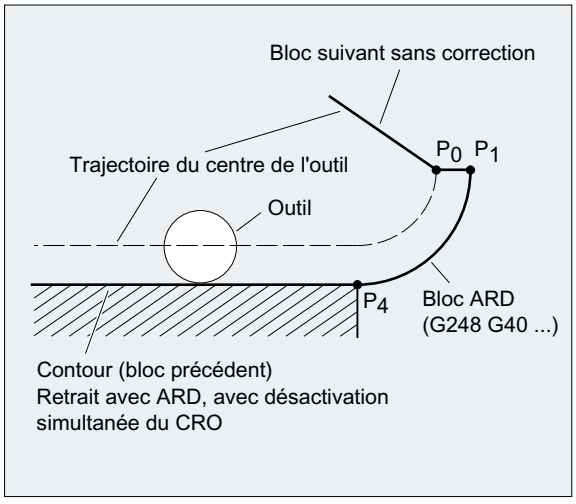
Pour le retrait, la programmation du point final du contour d'accostage / de retrait en douceur n'est pas prévue dans un bloc suivant, c.-à-d. que la position finale est toujours extraite du bloc

d'accostage / de retrait en douceur lui-même, indépendamment du nombre d'axes programmés. Les trois cas suivants sont alors à différencier pour la détermination du point final :

1. Aucun axe géométrique n'est programmé dans le bloc d'accostage / de retrait en douceur. Dans ce cas, le contour se termine au point P_1 (si DISRP est programmé), au point P_2 (si DISCL et non DISRP est programmé) ou au point P_3 (si ni DICLS ni DISRP ne sont programmés). La position sur les axes qui forment le plan d'usinage résulte du mouvement de retrait (point final de la ligne droite ou du cercle). La composante d'axe qui lui est perpendiculaire est définie par DISCL ou DISPR. Si dans ce cas DISCL=0 et DISPR=0, alors le mouvement est effectué entièrement dans le plan, c.-à-d. que les points P_0 à P_3 se confondent.
2. Dans le bloc d'accostage / de retrait en douceur, seul l'axe perpendiculaire au plan d'usinage est programmé. Le contour se termine alors au point P_0. Si DISRP est programmé (c.-à-d. que les deux points P_0 et P_1 ne se confondent pas), la droite $P_1 \rightarrow P_0$ est alors perpendiculaire au plan d'usinage. Les positions des deux autres axes sont obtenues comme en 1.
3. Au moins un axe du plan d'usinage est programmé. Le second axe éventuellement manquant du plan d'usinage est complété en mode modal à partir de sa dernière position dans le bloc précédent.

La position de l'axe perpendiculaire au plan d'usinage est obtenue comme en 1 ou en 2, que cet axe soit programmé ou non. La position ainsi obtenue définit le point final P_0. Si le bloc de retrait en douceur est simultanément le bloc de désactivation de la correction du rayon d'outil, alors dans les deux premiers cas, une composante de course supplémentaire est insérée dans le plan d'usinage de P_1 à P_0 de telle sorte qu'il n'y a pas de mouvement à la fin du contour de retrait lors de la désactivation de la correction du rayon d'outil, c.-à-d. que ce point ne définit pas une position sur un contour à corriger mais le centre de l'outil. Dans le troisième cas, la désélection de la correction du rayon d'outil ne doit pas être traitée séparément puisque le point P_0 programmé définit déjà directement la position du centre de l'outil à la fin du contour entier.

Le comportement dans le premier et le second cas, c.-à-d. en cas de point final non programmé explicitement dans le plan d'usinage avec désélection simultanée de la correction du rayon d'outil, est représenté dans la figure suivante :

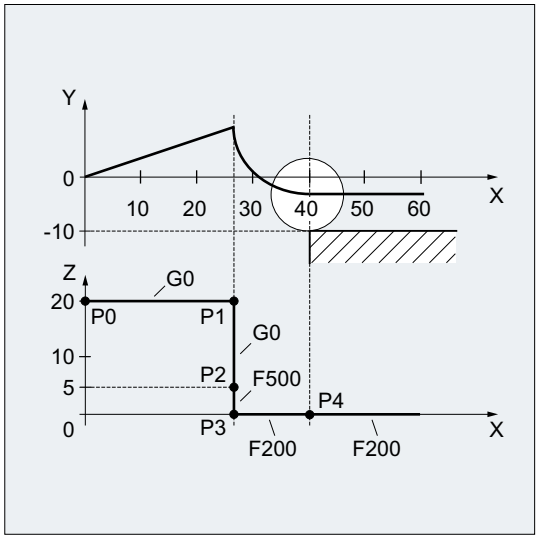


Vitesses d'accostage et de retrait

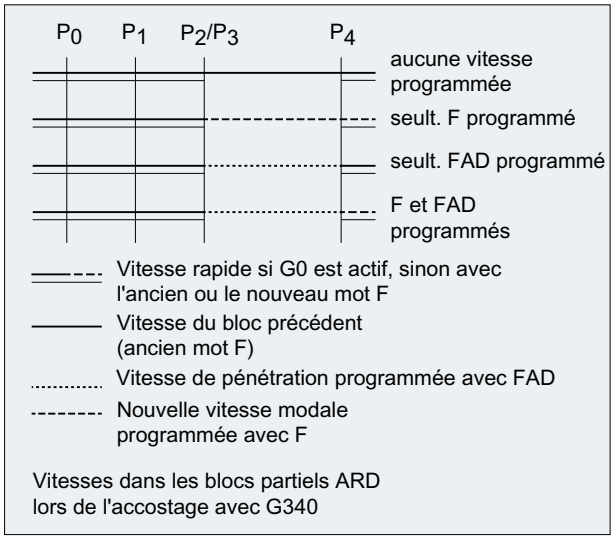
- Vitesse du bloc antérieur (G0)
  Tous les mouvements de P_0 à P_2, autrement dit le mouvement parallèle au plan d'usinage et le mouvement partiel d'approche jusqu'à la distance de sécurité sont exécutés à cette vitesse.
- Programmation avec FAD
  Indication de la vitesse d'avance pour
  - G341 : Mouvement de pénétration perpendiculaire au plan d'usinage de P_2 vers P_3
  - G340 : du point P_2 ou P_3 à P_4
    Lorsque FAD n'est pas programmé et qu'aucun mot F n'est programmé dans le bloc d'accostage / de retrait en douceur, cette partie du contour est également exécutée à la vitesse modale définie qui figure dans le bloc antérieur.
- Avance programmée F
  Cette valeur d'avance est appliquée à partir de P_3 ou P_2, si FAD n'est pas programmé. Si aucun mot F n'est programmé dans le bloc ARD, la vitesse appliquée est celle du bloc antérieur.

Exemple :

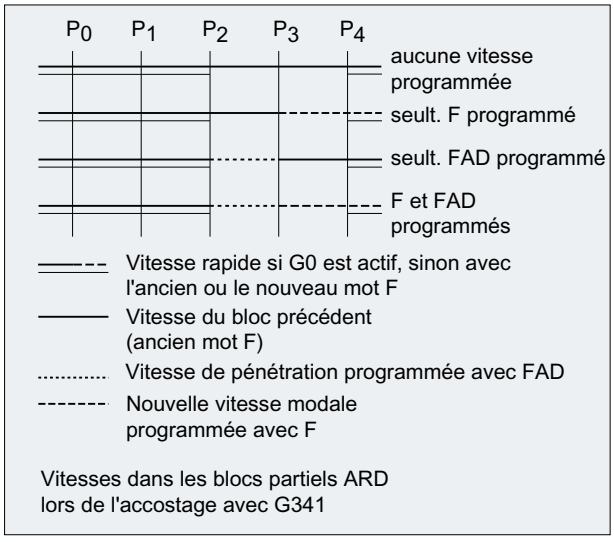
Code de programme	Commentaire
`$TC_DP1[1,1]=120`	`; Outil de fraisage T1/D1`
`$TC_DP6[1,1]=7`	`; Outil de 7 mm de rayon`
`N10 G90 G0 X0 Y0 Z20 D1 T1`	
`N20 G41 G341 G247 DISCL=AC(5) DISR=13 FAD 500 X40 Y-10 Z=0 F200`	
`N30 X50`	
`N40 X60`	
`...`	



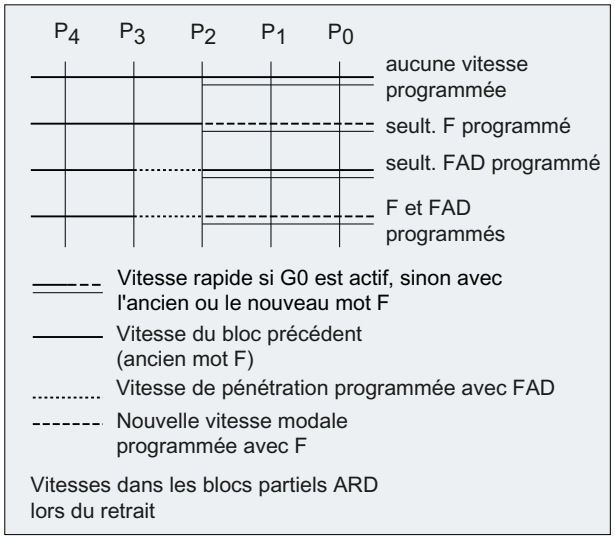
Dans le cas du retrait, le rôle de l'avance modale du bloc antérieur et celui de la valeur d'avance programmée dans le bloc ARD sont permutés, autrement dit le retrait du contour est exécuté avec l'ancienne avance et une nouvelle vitesse programmée avec le mot F est valable de P_2 à P_0.



Vitesses dans les blocs partiels ARD lors de l'accostage avec G340



Vitesses dans les blocs partiels ARD lors de l'accostage avec G341



Lecture de positions

Les points P_3 et P_4 peuvent être lus lors de l'accostage comme variable système dans le SCP.

- $P_APR: Lecture de P
- $_3$ (point d'attaque)
- $P_AEP: Lecture de P
- $_4$ (point d'attaque du contour)
- $P_APDV: lire si $P_APR et $P_AEP contiennent des valeurs valides

2.10.4.2 Accostage et retrait avec des stratégies de retrait étendues (G460, G461, G462)

Dans certains cas géométriques particuliers, par rapport à la réalisation effectuée jusqu'ici avec la surveillance de collision activée pour le bloc de retrait et le bloc d'accostage, des stratégies spécifiques et étendues de retrait et d'accostage sont nécessaires pour activer et désactiver la correction du rayon d'outil. Une surveillance de collision peut par exemple impliquer qu'un segment du contour ne soit pas entièrement usiné, voir la figure suivante :

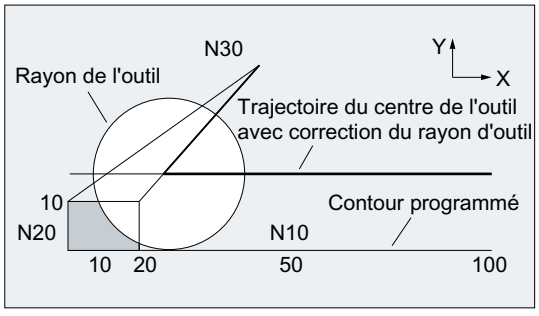


Figure 2-21 Comportement au retrait avec G460

Syntaxe

```
G460
G461
G462
```

Signification

G460 :	Comme précédemment (activation de la détection des violations du contour pour le bloc d'accostage et de retrait)
G461 :	Si aucun point d'intersection n'est trouvé, insertion, dans le bloc de CRO, d'un cercle dont le centre se trouve au point final du bloc non corrigé et dont le rayon est égal au rayon de l'outil Jusqu'au point d'intersection, l'usinage s'effectue avec un **cercle auxiliaire** autour du point final du contour (donc jusqu'à la fin du contour).
G462 :	Si aucun point d'intersection n'est trouvé, insertion, dans le bloc de CRO, d'une droite ; le bloc est prolongé par la tangente au point final (réglage par défaut) L'usinage s'effectue jusqu'au **prolongement** du dernier élément de contour (donc presque jusqu'à la fin du contour).

Remarque

Le comportement à l'accostage est analogue au comportement au retrait.

Le comportement à l'accostage ou au retrait est déterminé par l'état de la fonction G dans le bloc d'accostage ou de retrait. Le comportement à l'accostage peut donc être réglé indépendamment du comportement au retrait.

Exemples

Exemple 1 : comportement au retrait avec G460

Par la suite, seul le cas de la désactivation de la correction du rayon d'outil est représenté. Le comportement dans le cas de l'accostage est absolument analogue.

Code de programme	Commentaire
G42 D1 T1	; Rayon d'outil 20 mm
...	
G1 X110 Y0	
N10 X0	
N20 Y10	
N30 G40 X50 Y50	

Exemple 2 : accostage avec G461

Code de programme	Commentaire
N10 $TC_DP1[1,1]=120	; Type d'outil fraise
N20 $TC_DP6[1,1]=10	; Rayon de l'outil

Code de programme	Commentaire
N30 X0 Y0 F10000 T1 D1	
N40 Y20	
N50 G42 X50 Y5 G461	
N60 Y0 F600	
N70 X30	
N80 X20 Y-5	
N90 X0 Y0 G40	
N100 M30	

Informations complémentaires

G461

Si le dernier bloc de CRO ne présente pas d'intersection avec un bloc précédent, la trajectoire décalée de ce bloc est prolongée par un cercle dont le centre se trouve au point final du bloc non corrigé et dont le rayon est égal au rayon de l'outil.

La commande recherche ensuite un point d'intersection entre ce cercle et un des blocs précédents.

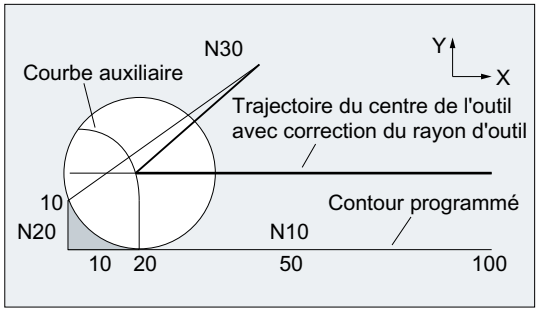


Figure 2-22 Comportement au retrait avec G461

Détection des violations du contour CDON, CDOF

Si CDOF est activé (voir chap. Surveillance de collision, CDON, CDOF), la recherche est interrompue dès qu'un point d'intersection a été trouvé, c.-à-d. que la commande ne vérifie pas s'il existe aussi des points d'intersection avec des blocs encore plus en amont dans le programme.

Si CDON est activé, la recherche se poursuit, même si un point d'intersection a déjà été trouvé.

Un point d'intersection ainsi trouvé est le nouveau point final du bloc précédent concerné et le point de départ du bloc de retrait. Le cercle inséré ne sert qu'à calculer le point d'intersection et n'entraîne aucun déplacement.

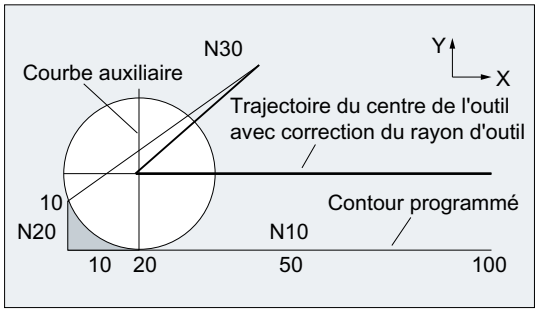
Remarque

Si aucun point d'intersection n'est trouvé, l'alarme 10751 (risque de collision) est affichée.

G462

Si le dernier bloc de CRO ne présente pas d'intersection avec un bloc précédent, une droite est insérée au point final du dernier bloc avec correction de rayon d'outil (le bloc est prolongé par la tangente au point final) lors du retrait avec G642 (fonction utilisée).

Une recherche de point d'intersection analogue au cas de G461 est ensuite effectuée.



Comportement au retrait avec G462 (voir l'exemple)

Avec G462, le coin formé par les blocs N10 et N20 de l'exemple n'est pas dégagé comme cela serait possible avec l'outil utilisé. Ce comportement peut cependant être nécessaire si le contour de la pièce (contrairement au contour programmé) ne doit pas être violé à gauche de N20, même pour des valeurs de y supérieures à 10 mm.

Comportement aux angles avec KONT

Si KONT est activé (contourner le coin au point de départ ou au point final), la commande distingue deux cas, selon que le point final se trouve devant ou derrière le contour.

- **Point final devant le contour**
  Si le point final se trouve devant le contour, le mode de retrait est identique à celui de NORM. Cette propriété n'est pas non plus modifiée lorsque le dernier bloc de contour est prolongé au moyen d'une droite ou d'un cercle pour G451. Des stratégies de contournement supplémentaires, afin d'éviter une violation du contour à proximité de son point final, ne sont donc pas nécessaires.
- **Point final derrière le contour**
  Si le point final se trouve derrière le contour, un cercle ou une droite sont toujours insérés, en fonction de G450/G451. G460 à G462 ne sont pas alors pris en considération. Si dans ce cas le dernier bloc de déplacement ne possède aucune intersection avec le bloc précédent, un point d'intersection peut alors résulter avec l'élément de contour inséré ou avec la section droite du point final du cercle de contournement au point final programmé.
  Lorsque l'élément de contour inséré est un cercle (G450) et que celui-ci forme un point d'intersection avec le bloc précédent, ce point d'intersection est le même que celui qui résulterait aussi avec NORM et G461. Cependant, la partie du cercle à parcourir sera en général plus grande. Pour la partie linéaire du bloc de retrait, un calcul de point d'intersection n'est plus nécessaire.
  Dans le deuxième cas, lorsque aucun point d'intersection de l'élément de contour inséré n'est trouvé avec les blocs précédents, l'outil se déplace sur le point d'intersection entre la droite de retrait et un bloc précédent.
  Un comportement modifié par rapport à G460 ne peut résulter avec G461 ou G462 actif que si soit NORM est actif, soit le comportement avec KONT est identique à celui avec NORM en fonction de la géométrie.

2.10.5 Activer/désactiver la surveillance de collision ("détection des goulets d'étranglement") (CDON, CDOF, CDOF2) :

Lorsque la correction du rayon d'outil est active, la détection des violations du contour ("détection des goulets d'étranglement") est activée ou désactivée dans le programme CN avec les instructions du groupe G 23.

Syntaxe

```
G41/G42 CDON
...
CDOF/CDOF2
```

Signification

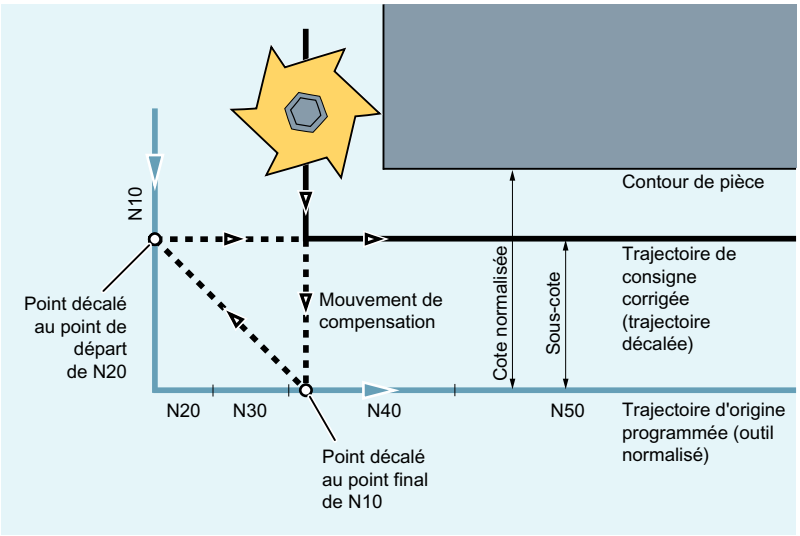
CDON :	Activer la surveillance de collision ("détection des goulets d'étranglement") Au-delà d'un nombre réglable de blocs (MD20240), CDON vérifie si les trajectoires des outils des blocs non voisins ou non consécutifs se croisent. Ceci permet de détecter à temps les risques de violation du contour et de les empêcher activement par la commande.
CDOF :	Désactiver la surveillance de collision ("détection des goulets d'étranglement") Avec CDOF, un point d'intersection commun est recherché pour le bloc en cours dans le bloc de déplacement précédent (aux angles rentrants) ainsi que, le cas échéant, dans des blocs antérieurs. Si un point d'intersection est trouvé, aucun autre bloc n'est pris en compte. En cas d'angles saillants, il existe toujours un point d'intersection entre deux blocs qui se suivent. **Remarque :** CDOF permet d'éviter les erreurs de détection des endroits étroits dues p. ex. à des informations manquantes dans le programme CN.
CDOF2 :	Désactiver la détection des violations du contour pour le fraisage périphérique 3D CDOF2 détermine le sens de correction d'outil des parties de bloc voisines. CDOF2 n'agit qu'en fraisage périphérique 3D (CUT3Dx) et a la même signification que CDOF pour tous les autres types d'usinage (CUT2D ou CUT3DFD par exemple).

Exemple d'effet de la détection des violations du contour

Le programme CN décrit la trajectoire du centre d'un outil normalisé. Le contour d'un outil en cours d'utilisation induit un écart inférieur qui est représenté sur la figure suivante de façon très grande et irréaliste uniquement pour expliciter la géométrie.

De plus, on suppose pour cet exemple que la commande ne concerne que trois blocs :

MD20240 $MC_CUTCOM_MAXNUM_CHECK_BLOCKS = 3



Du fait qu'un point d'intersection n'existe qu'entre les courbes d'offset des deux blocs N10 et N40, les deux blocs N20 et N30 devraient être omis. Dans l'exemple, la commande du bloc N40 n'est pas encore connue si N10 doit être usiné en dernier. Un seul bloc peut alors être omis.

En cas de CDOF2 active, le déplacement de compensation représenté sur la figure est exécuté et n'est pas suspendu. Dans ce cas, un CDOF ou CDON actifs entraîneraient une alarme.

2.10.6 Correction d'outil 2D ½ (CUT2D, CUT2DD, CUT2DF, CUT2DFD)

La correction du rayon d'outil 2D ½ doit être utilisée lorsque la **pièce**, et non l'orientation de l'outil, est pivotée pour l'usinage de surfaces inclinées. L'activation s'effectue par le biais des instructions `CUT2D`, `CUT2DD`, `CUT2DF` ou `CUT2DFD`.

Correction de longueur d'outil

La correction de longueur d'outil est toujours prise en compte par référence au plan de travail figé dans l'espace, non pivoté.

Correction du rayon d'outil 2D ½ pour outils de contour

La correction du rayon d'outil 2D ½ pour outils de contour est activée lorsque l'une des deux instructions `G41` (correction du rayon d'outil à gauche du contour) ou `G42` (correction du rayon d'outil à droite du contour) est programmée en même temps que `CUT2D`, `CUT2DD`, `CUT2DF` ou `CUT2DFD`. Elle sert à sélectionner automatiquement le tranchant d'outil pour les outils autres que de symétrie de révolution avec lesquels les segments de contour individuels peuvent être traités par morceaux.

Remarque

Si la correction du rayon d'outil 2D ½ n'est pas active, l'outil de contour se comporte comme un outil normal composé uniquement du premier tranchant.

Correction du rayon d'outil 2D ½ rapportée à un outil différentiel

La correction du rayon d'outil 2D ½ rapportée à un outil différentiel est activée au moyen des instructions `CUT2DD` ou `CUT2DFD`. Elle doit être utilisée lorsque le contour programmé se rapporte à la trajectoire du centre d'un outil différentiel et que l'usinage s'effectue avec un outil différent. Ainsi, lors du calcul de la correction du rayon d'outil 2D ½, seuls la valeur d'usure du rayon de l'outil actif ($TC_DP_15) et les offsets de correction d'outil OFFN (Page 256) et TOFFR (Page 85) éventuellement programmés sont pris en compte. Le rayon de base ($TC_DP6) de l'outil actif n'est **pas** pris en compte.

Syntaxe

```
CUT2D
CUT2DD
CUT2DF
CUT2DFD
```

Signification

`CUT2D` :	Activation de la correction du rayon 2D ½
`CUT2DD` :	Activation de la correction du rayon 2D ½ rapportée à un outil différentiel
`CUT2DF` :	Activation de la correction du rayon 2D ½, correction du rayon d'outil relative au frame courant ou à un plan oblique
`CUT2DFD` :	Activation de la correction du rayon 2D ½ rapportée à un outil différentiel, relative au frame courant ou à un plan oblique

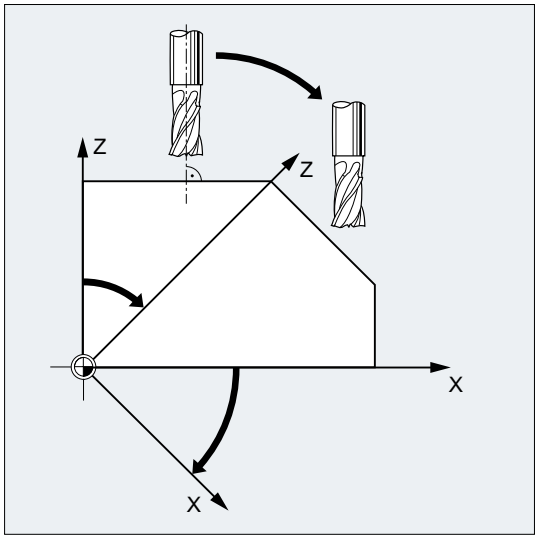
Informations complémentaires

Outils de contour

- Déblocage
  Le déblocage de la correction du rayon d'outil pour les outils de contour s'effectue de manière spécifique au canal par le biais de :
  PM28290 $MC_MM_SHAPED_TOOLS_ENABLE
- Type d'outil
  Les types d'outil des outils de contour sont définis de manière spécifique au canal par le biais de :
  PM20370 $MC_SHAPED_TOOL_TYPE_NO
- Tranchants
  Un nombre de tranchants (numéros D) peut être affecté à chaque outil de contour dans l'ordre de votre choix. Le nombre maximal de tranchants par outil est paramétré par le biais de :
  PM18106 $MN_MM_MAX_CUTTING_EDGE_PERTOOL
  Le premier tranchant d'un outil de contour est le tranchant sélectionné lors de l'activation de l'outil. Si, par exemple, le cinquième tranchant (D5) du troisième outil (T3) est activé dans un programme par les instructions `T3 D5`, D5 et les tranchants consécutifs définissent alors l'outil de contour soit avec une pièce soit avec toutes les pièces. Les tranchants qui précèdent D5 sont ignorés.

Correction du rayon d'outil 2D ½ sans rotation du plan de correction (CUT2D, CUT2DD)

Quand un frame contenant une rotation est programmé, le plan dans lequel se trouve la correction du rayon d'outil (plan de correction) n'est **pas pivoté** en même temps avec `CUT2D` ou `CUT2DD`. La correction du rayon d'outil rapportée au plan de travail **non pivoté** (`G17`, `G18`, `G19`) est prise en compte. La correction de longueur d'outil agit par rapport au plan de correction.

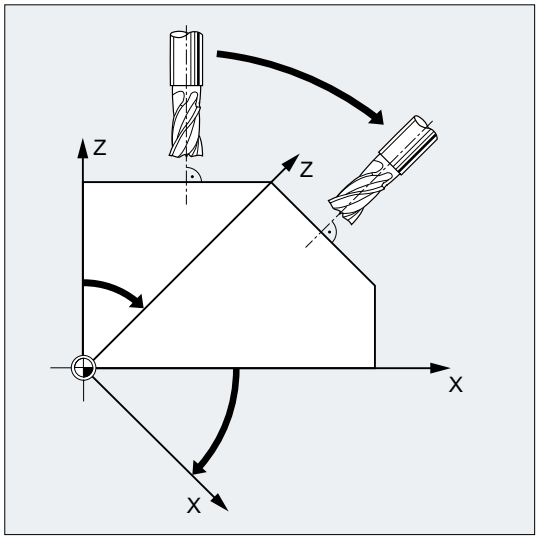


Pour l'usinage de surfaces inclinées, les valeurs de correction d'outil doivent être définies en conséquence ou calculées en utilisant la fonctionnalité "Correction de longueur d'outil pour outils orientables".

Correction du rayon d'outil 2D ½ avec rotation du plan de correction (CUT2DF, CUT2DFD)

Quand un frame contenant une rotation est programmé, le plan dans lequel se trouve la correction du rayon d'outil (plan de correction) est également **pivoté** avec `CUT2DF` ou `CUT2DFD`. La correction du rayon d'outil rapportée au plan de travail **pivoté** (`G17`, `G18`, `G19`) est prise en compte. Cependant, la correction de longueur d'outil continue d'agir par rapport au plan de travail **non pivoté**.

Condition : Sur la machine, l'orientation d'outil doit être réglable perpendiculairement au plan de travail pivoté et doit être réglée pour l'usinage.



Remarque

La correction de longueur d'outil continue d'agir par rapport au plan de travail non pivoté.

Pour plus d'informations, voir Description fonctionnelle "Outils".

2.10.7 Maintenir constante la correction du rayon d'outil (CUTCONON, CUTCONOF)

La fonction "Maintenir constante la correction du rayon d'outil" sert à inhiber la correction du rayon d'outil pendant un certain nombre de blocs, un décalage entre la trajectoire programmée et la trajectoire effective du centre de l'outil résultant de la correction du rayon d'outil dans les blocs précédents étant conservé. Elle peut être utilisée en fraisage par balayage par exemple si plusieurs blocs de déplacement sont nécessaires aux points d'inversion de sens, les trajectoires générées par la correction du rayon d'outil (stratégies de contournement) n'étant cependant pas désirées. Elle est indépendante du type de correction du rayon d'outil ($2^1/_2$D, fraisage en bout 3D, fraisage périphérique 3D).

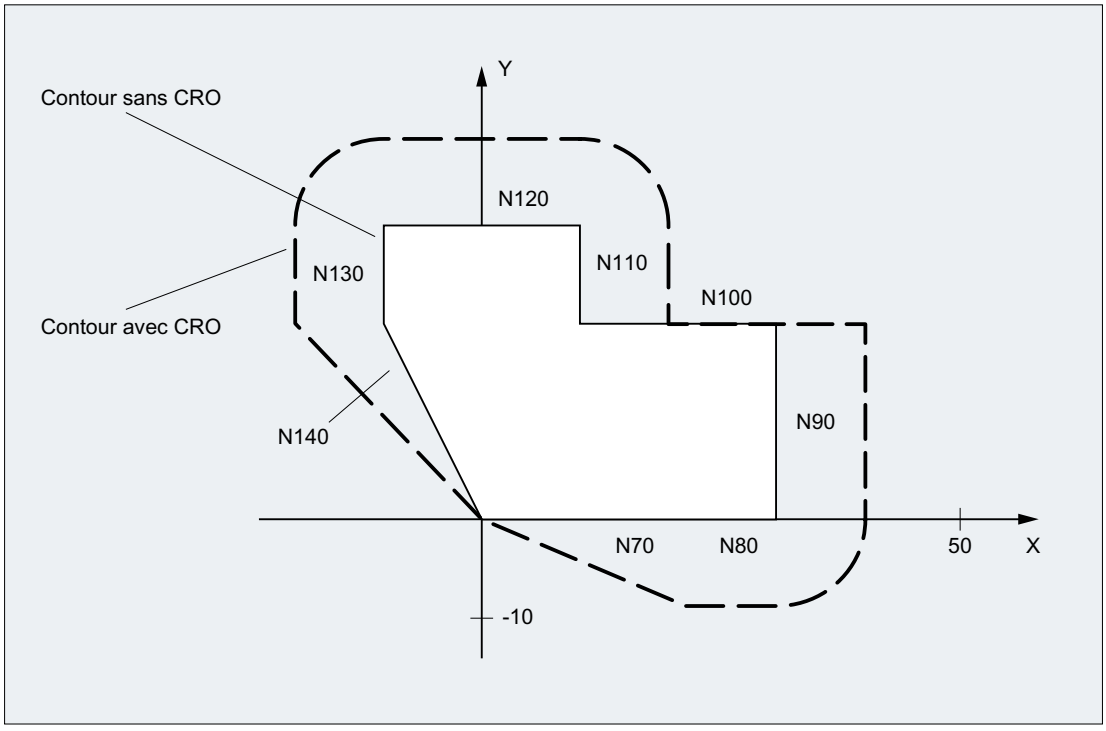
Syntaxe

```
CUTCONON
CUTCONOF
```

Signification

`CUTCONON` :	Instruction d'activation de la fonction "Maintenir constante la correction du rayon d'outil"
`CUTCONOF` :	Instruction de désactivation de la fonction "Maintenir constante la correction du rayon d'outil"

Exemple



Code de programme	Commentaire
N10	; Définition de l'outil d1
N20 $TC_DP1[1,1]=110	; Type
N30 $TC_DP6[1,1]= 10.	; Rayon
N40	
N50 X0 Y0 Z0 G1 G17 T1 D1 F10000	
N60	
N70 X20 G42 NORM	
N80 X30	
N90 Y20	
N100 X10 CUTCONON	; Activation de l'inhibition de la correction
N110 Y30 KONT	; Insertion possible d'un arc de contournement lors de la désactivation de l'inhibition de contours
N120 X-10 CUTCONOF	
N130 Y20 NORM	; Pas d'arc de contournement lors de la désactivation de la CRO
N140 X0 Y0 G40	
N150 M30	

Autres informations

La correction de rayon d'outil est en principe déjà active avant l'activation de son inhibition et elle est encore active lors de la désactivation de l'inhibition. Dans le dernier bloc de déplacement qui précède CUTCONON, l'offset au point final du bloc est accosté. Tous les blocs suivants, dans lesquels l'inhibition de la correction est activée, sont exécutés sans correction. Ils sont cependant translatés du vecteur reliant le point final du dernier bloc avec correction et son point décalé. Le type d'interpolation de ces blocs (linéaire, circulaire, polynomiale) peut être quelconque.

Le bloc de désactivation de l'inhibition de la correction, autrement dit le bloc qui contient CUTCONOF, est corrigé normalement. Il commence sur l'offset au point de départ. Entre le point final du bloc précédent, c'est-à-dire le dernier bloc de déplacement programmé avec CUTCONON actif, et ce point, un bloc linéaire est inséré.

Les blocs à interpolation circulaire avec un plan de cercle perpendiculaire au plan de correction (cercles verticaux) sont traités comme s'ils contenaient une programmation de CUTCONON. Cette activation implicite de l'inhibition de la correction est automatiquement annulée dans le premier bloc de déplacement qui contient un déplacement dans le plan de correction qui n'est pas un tel cercle. Vous ne rencontrerez des cercles verticaux satisfaisant à ces critères qu'en fraisage périphérique.

2.10.8 Outils à position de tranchant définie

Dans le cas des outils à position de tranchant définie (outils de tournage et de rectification – types 400 à 599 ; voir le chapitre "Exploitation du signe de l'usure"), le basculement de G40 vers G41/G42 ou inversement est interprété comme un changement d'outil. Ceci provoque, si une transformation (par ex. TRANSMIT) est active, un arrêt du prétraitement des blocs (arrêt du décodage) et, par conséquent, des écarts par rapport au contour à obtenir sont possibles.

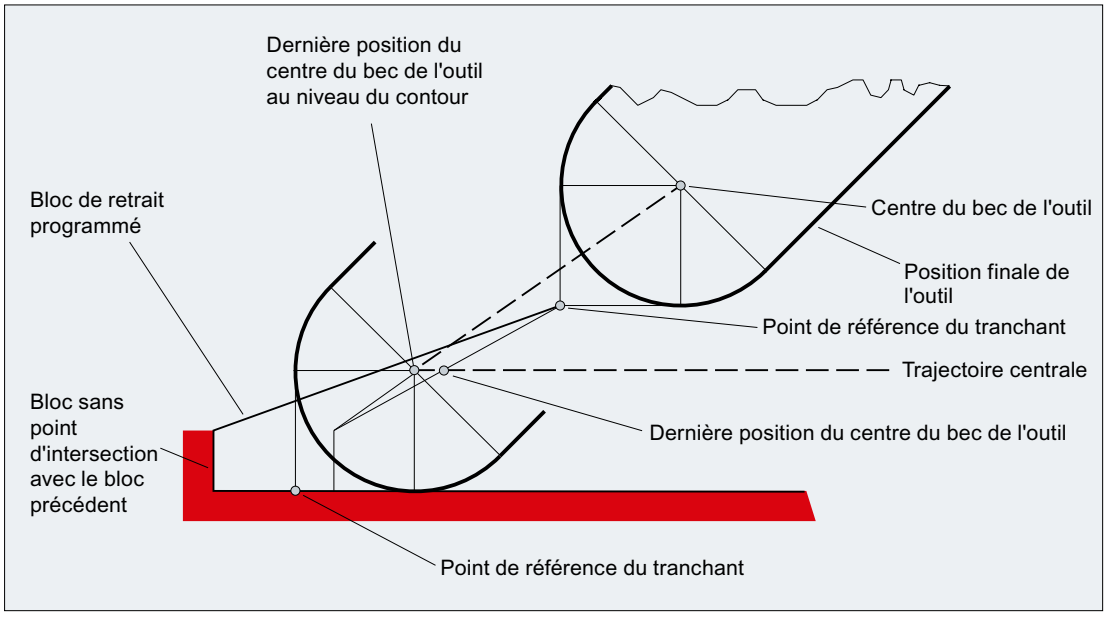
Cette fonctionnalité initiale est modifiée au niveau de :

1. arrêt du prétraitement des blocs en présence de TRANSMIT
2. calcul de points d'intersection pour l'accostage / le retrait avec KONT
3. changement d'un outil avec correction active du rayon d'outil
4. correction du rayon d'outil avec orientation de l'outil en cas de transformation.

Informations complémentaires

La fonctionnalité initiale a été modifiée comme suit :

- Le basculement de G40 vers G41/G42 ou inversement n'est plus interprété comme un changement d'outil. Avec TRANSMIT, un arrêt du prétraitement des blocs n'a donc plus lieu.
- La droite qui relie les centres de bec d'outil en début et en fin de bloc est utilisée pour le calcul de points d'intersection avec le bloc d'accostage ou de retrait. La différence entre le point de référence du tranchant et le centre du bec de l'outil se superpose à ce déplacement.
  Pour l'accostage ou le retrait avec KONT (l'outil contourne le point de contour ; voir chapitre "Accostage et retrait du contour"), la correction s'effectue dans le sous-bloc linéaire du mouvement d'accostage ou de retrait. Les rapports géométriques sont donc identiques pour les outils avec et sans position de l'arête tranchante. Des différences par rapport au comportement connu jusqu'à présent ne se rencontrent que dans les cas relativement rares où le bloc d'accostage ou de retrait possède un point d'intersection avec un bloc de déplacement non voisin, comme le montre la figure suivante :



- Si la correction de rayon d'outil est activée, un changement d'outil qui entraîne une modification de la distance entre centre du bec de l'outil et point de référence du tranchant est interdit dans les blocs circulaires et les blocs de déplacement à polynômes rationnels dont le dénominateur a un degré > 4. Dans le cas des autres types d'interpolation, un changement d'outil est permis même si une transformation est active (par ex. TRANSMIT), ce qui n'était pas le cas jusqu'à présent.
- Dans le cas de la correction de rayon d'outil avec orientation variable de l'outil, la transformation assurant le passage du point de référence du tranchant au centre du bec de l'outil n'est plus réalisable par un simple décalage d'origine. C'est pourquoi les outils à position de tranchant définie sont interdits en fraisage périphérique 3D (alarme).

Remarque

Ces indications ne concernent pas le fraisage en bout, pour lequel uniquement des outils sans position de tranchant définie sont autorisés, comme jusqu'à présent. (Les outils d'un type non autorisé explicitement sont considérés comme des fraises à bout hémisphérique de rayon indiqué. Si une position de tranchant est indiquée, elle est ignorée.)

2.11 Modes de déplacement

2.11.1 Arrêt précis (G60, G9, G601, G602, G603)

L'arrêt précis est un mode de déplacement dans lequel tous les axes à interpolation et les axes supplémentaires concernés par le déplacement et dont le déplacement ne s'étend pas sur plusieurs blocs freinent jusqu'à l'immobilisation complète à la fin de chaque bloc de déplacement.

L'arrêt précis est utilisé pour l'exécution d'angles saillants ou la finition d'angles rentrants.

Le critère d'arrêt précis détermine la précision d'accostage du coin et l'instant où a lieu le changement de bloc.

- "Arrêt précis fin"
  Le changement de bloc est déclenché dès que tous les axes concernés par le déplacement ont atteint les limites de tolérance "Arrêt précis fin" spécifiques à l'axe.
  Un "arrêt précis fin" est réglé par : MD36010 $MA_STOP_LIMIT_FINE[<axe>]
- "Arrêt précis grossier"
  Le changement de bloc est déclenché dès que tous les axes concernés par le déplacement ont atteint les limites de tolérance "Arrêt précis grossier" spécifiques à l'axe.
  Un "arrêt précis grossier" est réglé par : MD36000 $MA_STOP_LIMIT_COARSE[<axe>]
- "Fin de l'interpolation"
  Le changement de bloc est déclenché dès que la commande a calculé une vitesse de consigne nulle pour tous les axes concernés par le déplacement. La position réelle et l'écart de traînage des axes concernés ne sont pas pris en compte.

Syntaxe

```
G60 ...
G9 ...
G601/G602/G603 ...
```

Signification

G60 :	Instruction d'activation de l'arrêt précis à effet **modal**
G9 :	Instruction d'activation de l'arrêt précis à effet **non modal**
G601 :	Instruction d'activation du critère d'arrêt précis "**Arrêt précis fin**"
G602 :	Instruction d'activation du critère d'arrêt précis "**Arrêt précis grossier**"
G603 :	Instruction d'activation du critère d'arrêt précis "**Fin de l'interpolation**"

Remarque

Les instructions d'activation des critères d'arrêt précis (G601 / G602 / G603) s'appliquent uniquement lorsque G60 ou G9 sont actifs.

Exemple

Code de programme	Commentaire
N5 G602	; Sélection du critère "Arrêt précis grossier"
N10 G0 G60 Z...	; Arrêt précis modal actif
N20 X... Z...	; G60 continue d'agir
...	
N50 G1 G601	; Sélection du critère "Arrêt précis fin"
N80 G64 Z...	; Commutation sur contournage
...	
N100 G0 G9	; L'arrêt précis s'applique uniquement à ce bloc
N110 ...	; Réactivation du contournage

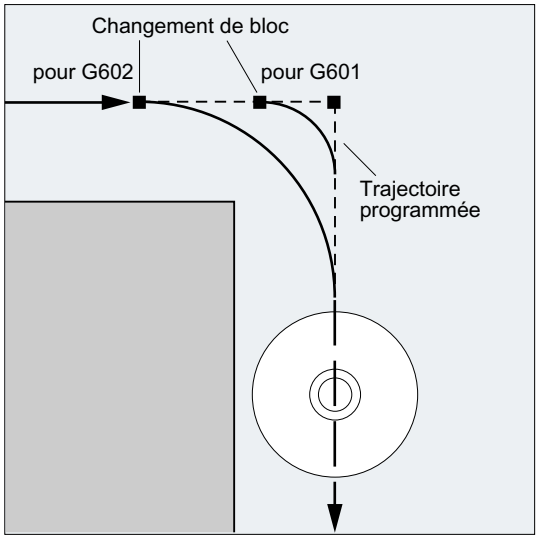
Informations complémentaires

G60, G9

G9 génère l'arrêt précis dans le bloc courant ; G60 le génère dans le bloc courant et dans tous les blocs suivants.

Avec les instructions de contournage G64 ou G641 ... G645, G60 est désactivé.

G601, G602



Le déplacement est freiné et arrêté brièvement au point d'angle.

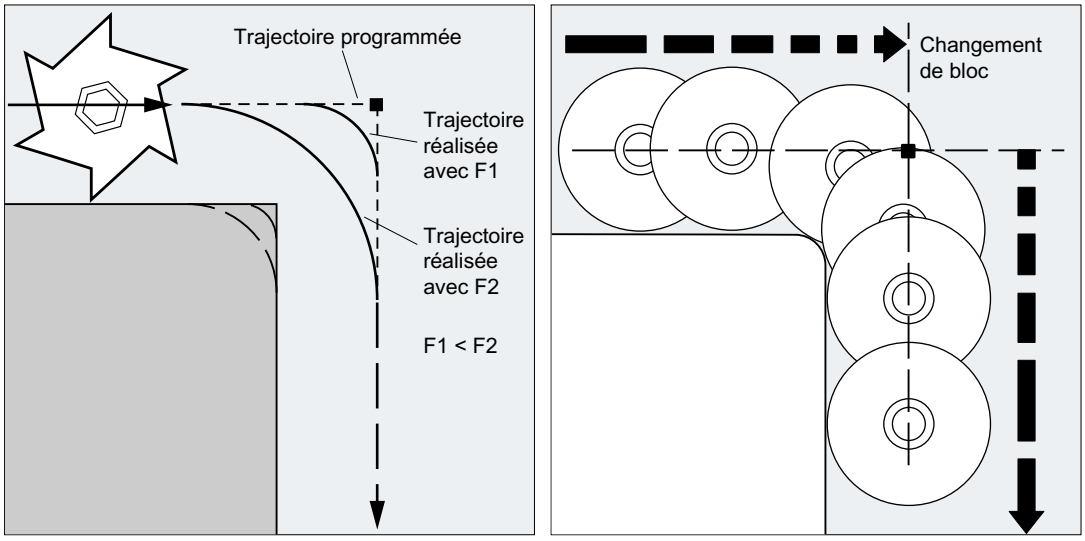
Remarque

Pour les critères d'arrêt précis, il est recommandé de définir des limites aussi petites que possible. Plus les limites sont petites, plus l'accostage de la position de destination sera long.

G603

Le changement de bloc est déclenché dès que la commande a calculé une vitesse de consigne nulle pour les axes concernés. A ce moment-là, la valeur réelle de position est en retard, compte

tenu de l'écart de traînage qui est fonction de la dynamique des axes et de la vitesse tangentielle. Ceci permet d'arrondir les angles de la pièce.



Critère d'arrêt précis configuré

Pour G0 et les autres instructions du 1er groupe G, il est possible de définir spécifiquement pour chaque canal l'utilisation automatique d'un critère prédéfini, différent du critère d'arrêt précis programmé (voir les indications du constructeur de machines !).

2.11.2 Contournage (G64, G641, G642, G643, G644, G645, ADIS, ADISPOS)

Pour le contournage, la vitesse tangentielle en fin de bloc n'est pas freinée jusqu'à une vitesse permettant d'atteindre le critère d'arrêt précis, en vue du changement de bloc. Le but est, par contre, d'éviter une décélération importante des axes à interpolation au point de changement de bloc, afin de passer au bloc suivant avec, si possible, la même vitesse tangentielle. Pour cela, la fonction "Pilotage anticipé de la vitesse (LookAhead)" est activée en même temps que le contournage.

Pour le contournage avec arrondissement, les changements de blocs coudés sont formés ou lissés tangentiellement au moyen de modifications locales de la trajectoire programmée.

Effets du contournage :

- Un arrondi du contour.
- Des temps d'usinage plus courts du fait de l'absence des phases de freinage et d'accélération nécessaires lorsqu'il s'agit de satisfaire le critère d'arrêt précis.
- Des conditions de coupe meilleures du fait d'une plus grande régularité de la vitesse.

Le contournage est recommandé lorsque :

- le retrait d'un contour doit être aussi continu que possible (p. ex. avec la vitesse rapide),
- une déviation de la trajectoire précise programmée est autorisée en cas de critère d'erreur, afin d'atteindre une trajectoire parfaitement régulière.

Le contournage n'est pas recommandé lorsque :

- le retrait du contour doit être précis,
- une constante de vitesse absolue est nécessaire.

Remarque

Le contournage est interrompu par les blocs qui déclenchent implicitement l'arrêt du prétraitement des blocs, p. ex. par :

- Accès à certaines données d'état de la machine ($A...)
- Sorties de fonctions auxiliaires

Syntaxe

```
G64 ...
G641 ADIS=...
G641 ADISPOS=...
G642 ...
G643 ...
G644 ...
G645 ...
```

Signification

`G64` :	Contournage avec décélération selon le facteur de surcharge
`G641` :	Contournage avec arrondissement selon le critère de trajet
`ADIS=...` :	Critère de trajet pour `G641`, pour les fonctions de trajectoire `G1`, `G2`, `G3`, ...
`ADISPOS=...` :	Critère de trajet pour `G641`, pour la vitesse rapide `G0`
	Le critère de trajet (= distance de transition) `ADIS` ou `ADISPOS` décrit le parcours que le bloc d'arrondissement peut commencer au plus tôt avant la fin du bloc ou celui que le bloc d'arrondissement doit terminer après la fin du bloc. **Remarque :** Si aucun `ADIS/ADISPOS` n'est programmé, la valeur nulle et, de ce fait, le comportement de déplacement, s'applique de la même manière que pour `G64`. Dans les trajets courts, la distance de transition est réduite automatiquement (au maximum jusqu'à 36 %).
`G642` :	Contournage avec arrondissement avec prise en compte des tolérances définies Dans ce mode, l'arrondissement s'effectue normalement avec prise en compte de l'écart de trajectoire maximum autorisé. A la place de cette tolérance spécifique à l'axe, vous pouvez cependant également configurer la prise en compte de l'écart maximum du contour (tolérance de contour) ou de l'écart maximum par rapport à l'angle d'orientation de l'outil (tolérance d'orientation). **Remarque :** L'extension à la tolérance de contour et à la tolérance d'orientation existe uniquement dans les systèmes disposant de l'option "Interpolation polynomiale".

G643 :	Contournage avec arrondissement avec prise en compte des tolérances définies (internes au bloc) Contrairement à G642, G643 ne forme pas de bloc d'arrondissement spécifique, mais insère des arrondissement internes au bloc, spécifiques à chaque axe. Le trajet d'arrondissement peut différer pour chaque axe.
G644 :	Contournage avec arrondissement utilisant la dynamique maximale possible **Remarque :** G644 n'est pas possible lorsque la transformation cinématique est active. La commutation s'effectue de manière interne sur G642.
G645 :	Contournage avec arrondissement d'angles et changements de blocs tangentiels avec prise en compte de tolérances définies G645 usine les angles de façon identique à G642. Pour G645, des blocs d'arrondissement sont également uniquement formés pour des changements de blocs tangentiels, si la trajectoire de courbure du contour original présente un saut sur un axe au moins.

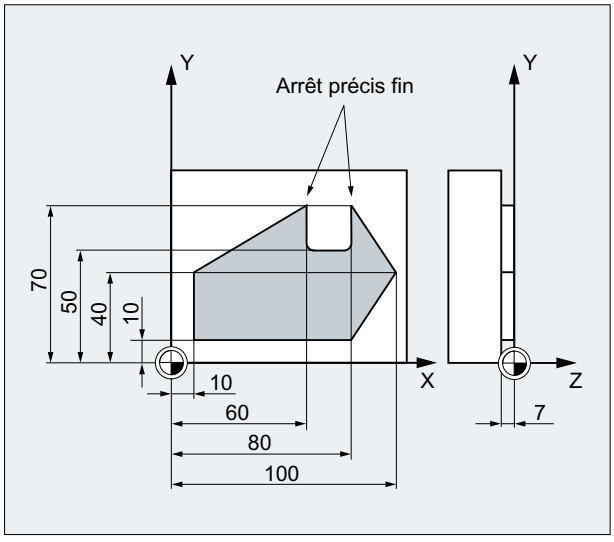
Remarque

L'arrondissement par contournage ne remplace pas la fonction d'arrondissement (RND). En effet, il est impossible de prédire l'allure du contour dans la zone d'arrondissement. L'arrondissement peut, en particulier, dépendre de paramètres dynamiques, tels que la vitesse tangentielle. C'est pourquoi il ne présente d'intérêt que pour les faibles valeurs d'ADIS. Si vous souhaitez appliquer un contour défini à l'angle, vous devez utiliser RND.

Remarque

Si un arrondissement généré par G641, G642, G643, G644 ou G645 est interrompu, le repositionnement suivant (REPOS) ne se fait pas par accostage du point d'interruption, mais du point de départ ou du point final du bloc de déplacement original (selon le mode REPOS).

Exemple



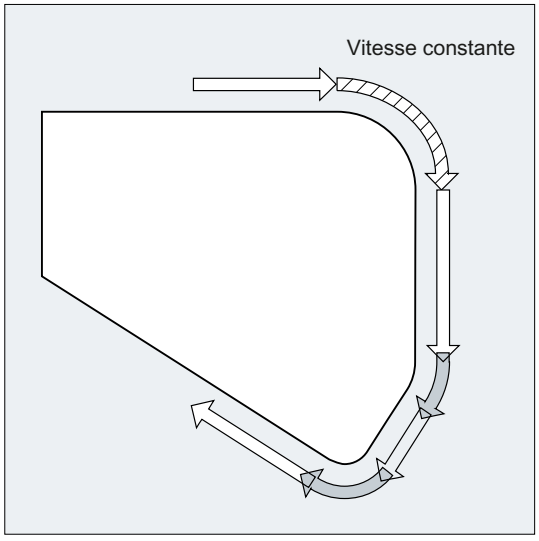
Les deux angles saillants de la rainure doivent être accostés avec précision. Sinon l'usinage doit être effectué par contournage.

Code de programme	**Commentaire**
N05DIAMOF	; Cotation sur forme de rayons
N10 G17 T1 G41 G0 X10 Y10 Z2 S300 M3	; Accostage de la position de départ, mise en marche de la broche, correction de trajectoire
N20 G1 Z-7 F8000	; Prise de passe de l'outil
N30 G641 ADIS=0.5	; Les transitions sont arrondies
N40 Y40	
N50 X60 Y70 G60 G601	; Accostage précis de la position avec arrêt précis fin
N60 Y50	
N70 X80	
N80 Y70	
N90 G641 ADIS=0.5 X100 Y40	; Les transitions sont arrondies
N100 X80 Y10	
N110 X10	
N120 G40 G0 X-20	; Désactivation de la correction de trajectoire
N130 Z10 M30	; Dégagement de l'outil, fin du programme

Informations complémentaires

Contournage G64

En contournage, l'outil se déplace à une vitesse aussi constante que possible au niveau des transitions tangentielles (pas de ralentissement aux limites de blocs). Un freinage anticipé (LookAhead) a lieu devant les angles et les blocs avec arrêt précis.



L'outil ne s'arrête pas au niveau des angles. Pour minimiser l'erreur de contour, la vitesse est réduite en tenant compte d'une limite d'accélération et d'un facteur de surcharge.

Remarque

L'importance de l'arrondissement des angles dépend de la vitesse d'avance et du facteur de surcharge. Le réglage du facteur de surcharge s'effectue dans MD32310 $MA_MAX_ACCEL_OVL_FACTOR.

Avec la définition de MD20490 $MC_IGNORE_OVL_FACTOR_FOR_ADIS, l'arrondissement des changements de blocs s'effectue toujours indépendamment du facteur de surcharge réglé.

Pour éviter un arrêt indésirable du déplacement avec interpolation (dégagement de l'outil), tenir compte des points suivants :

- Les fonctions auxiliaires activées après la fin du déplacement ou avant le déplacement suivant interrompent le contournage (exception : fonctions auxiliaires rapides).
- Les axes de positionnement se déplacent toujours selon le principe de l'arrêt précis, fenêtre de positionnement fine (comme G601). Si, dans un bloc CN, il est nécessaire d'attendre des axes de positionnement, le contournage des axes à interpolation est suspendu.

Les blocs intermédiaires contenant uniquement des commentaires, des blocs de calcul ou des appels de sous-programmes ne provoquent par contre pas l'arrêt du contournage.

Remarque

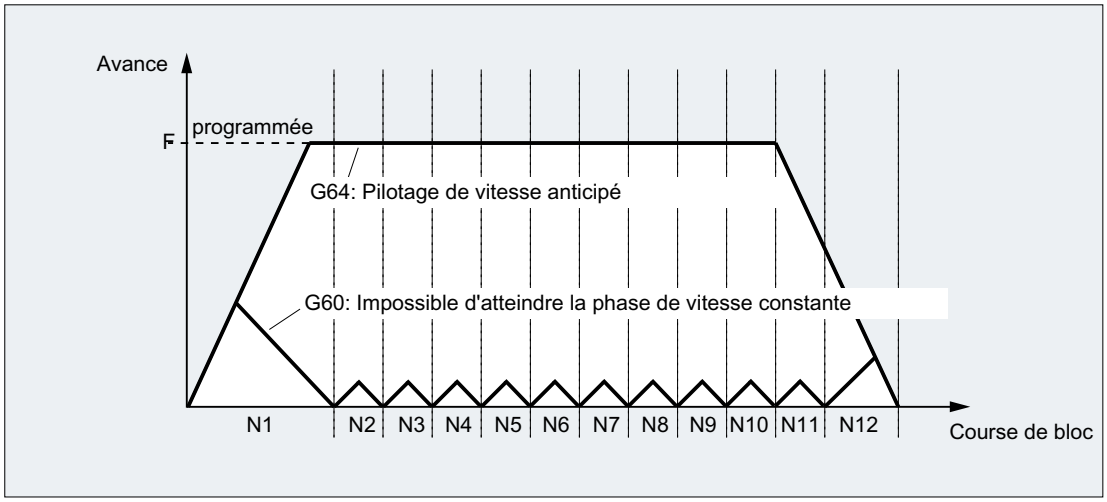
Quand les axes à interpolation ne figurent pas tous dans FGROUP, on remarquera fréquemment au moment des changements de blocs une brusque variation de vitesse des axes absents, variation de vitesse que la commande ramène à la limite admise dans les MD32300 : $MA_MAX_AX_ACCEL et MD32310 : $MA_MAX_ACCEL_OVL_FACTOR en réduisant la vitesse au changement de bloc. Ce freinage peut être évité en effectuant un arrondissement entre les blocs qui adoucit le rapport entre les positions existantes des axes à interpolation.

Pilotage de vitesse anticipé LookAhead

Pour le contournage, la commande détermine automatiquement, par anticipation, le pilotage de la vitesse pour plusieurs blocs CN. Ceci permet d'accélérer ou de freiner sur plusieurs blocs dans le cas de transitions approximativement tangentielles.

Le pilotage anticipé de la vitesse permet surtout de réaliser des séquences de déplacement composées de trajets courts avec des avances tangentielles élevées.

Le nombre maximal de blocs CN pris en compte dans l'anticipation de vitesse est à spécifier dans un paramètre machine.



Contournage avec arrondissement selon le critère de trajet (G641)

Avec G641, la commande insère des éléments de transition entre les blocs. L'écart d'arrondissement ADIS (ou ADISPOS pour G0) indique l'arrondissement maximal autorisé pour les angles. Dans cet écart d'arrondissement, la commande peut annuler la trajectoire prévue et la remplacer par un trajet dynamique optimisé.

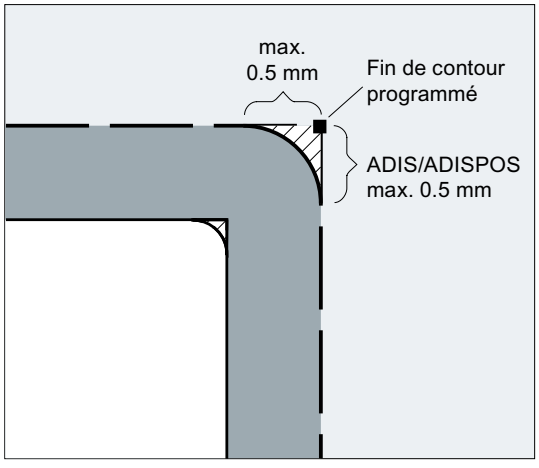
Inconvénient : Pour tous les axes, une seule valeur ADIS est disponible.

G641 est analogue à RNDM, mais n'est pas limitée aux axes du plan de travail.

Comme G64, G641 utilise le pilotage de vitesse anticipé LookAhead. Les blocs de transition à courbure élevée sont exécutés à vitesse réduite.

Exemple :

Code de programme	**Commentaire**
`N10 G641 ADIS=0.5 G1 X... Y...`	`; Le bloc d'arrondissement peut commencer au plus tôt 0,5 mm avant et se terminer au plus tard 0,5 mm après la fin de bloc programmée. Ce réglage est modal.`



Remarque

L'arrondissement ne peut et ne doit pas remplacer les fonctions de lissage défini (RND, RNDM, ASPLINE, BSPLINE, CSPLINE).

Arrondissement avec précision axiale pour G642

Pour G642, l'arrondissement n'est pas effectué à l'intérieur de la plage ADIS définie, mais avec prise en compte des tolérances axiales définies avec MD33100 $MA_COMPRESS_POS_TOL. Le trajet d'arrondissement est déterminé à partir du trajet d'arrondissement le plus court de l'ensemble des axes. Cette valeur est prise en compte lors de la création d'un bloc d'arrondissement.

Arrondissement interne à un bloc pour G643

Pour l'arrondissement avec G643, les écarts maximum du contour précis sont définis pour chaque axe avec le paramètre machine MD33100 $MA_COMPRESS_POS_TOL.

G643 ne crée pas un bloc d'arrondissement propre, mais insère dans les blocs des arrondissements spécifiques à chaque axe. G643 permet de définir un trajet d'arrondissement différent pour chaque axe.

Arrondissement avec tolérance de contour et tolérance d'orientation pour G642/G643

MD20480 $MC_SMOOTHING_MODE permet de configurer l'arrondissement avec G642 et G643 de sorte à appliquer une tolérance de contour et une tolérance d'orientation à la place des tolérances spécifiques aux axes.

Le réglage de la tolérance de contour et de la tolérance d'orientation s'effectue dans les données de réglage spécifiques aux canaux :

SD42465 $SC_SMOOTH_CONTUR_TOL (écart maximum par rapport au contour)

SD42466 $SC_SMOOTH_ORI_TOL (écart maximum par rapport à l'angle d'orientation de l'outil)

Les données de réglage peuvent être programmées dans le programme CN et donc être prédéfinies différemment pour chaque changement de bloc. Une divergence importante entre la tolérance de contour et la tolérance d'orientation peut uniquement être appliquée pour G643.

Remarque

L'extension à la tolérance de contour et à la tolérance d'orientation existe uniquement dans les systèmes disposant de l'option "Interpolation de polynômes".

Remarque

L'arrondissement avec prise en compte de la tolérance d'orientation nécessite l'activation d'une transformation d'orientation.

Arrondissement avec dynamique maximale possible pour G644

La configuration de l'arrondissement avec dynamique maximale s'effectue en milliers avec MD20480 $MC_SMOOTHING_MODE :

Valeur	Signification
0	Spécification des écarts axiaux maximum avec : MD33100 $MA_COMPRESS_POS_TOL
1	Spécification du trajet d'arrondissement maximum par programmation de : ADIS=... ou ADISPOS=...
2	Spécification des fréquences maximales apparaissant pour chaque axe dans la plage d'arrondissement avec : MD32440 $MA_LOOKAH_FREQUENCY La plage de transition est définie de façon que, pendant le déplacement d'arrondissement, aucune fréquence dépassant la fréquence maximale prescrite n'apparaisse.
3	Pendant l'arrondissement avec G644, ni la tolérance, ni la distance de transition entre blocs ne sont surveillées. Chaque axe se déplace avec la dynamique maximale possible sur l'angle. Avec SOFT, aussi bien l'accélération maximale que l'à-coup maximal de chaque axe sont respectés. Avec BRISK, l'à-coup n'est pas limité mais chaque axe se déplace avec l'accélération maximale possible.

Arrondissement de changements de blocs tangentiel pour G645

Le déplacement d'arrondissement pour G645 est défini de telle sorte qu'il ne se produit aucune accélération brutale des axes impliqués, ni aucun dépassement de l'écart maximal paramétré par rapport au contour initial (MD33120 $MA_PATH_TRANS_POS_TOL).

Dans le cas de changements de blocs coudés, non tangentiels, le comportement d'arrondissement est identique à celui de G642.

Aucun bloc d'arrondissement intermédiaire

Dans les cas suivants, aucun bloc d'arrondissement intermédiaire n'est inséré :

- Entre deux blocs, il y a arrêt.
  Cette situation se présente quand :
  - Il y a sortie de fonction auxiliaire avant exécution du déplacement dans le bloc suivant
  - Le bloc suivant ne contient aucun déplacement
  - Dans le bloc suivant, un axe qui était auparavant axe de positionnement est déplacé pour la première fois comme axe à interpolation.
  - Dans le bloc suivant, un axe qui était auparavant axe à interpolation est déplacé pour la première fois comme axe de positionnement.
  - Le bloc précédent déplace des axes géométriques, mais pas le bloc suivant.
  - Le bloc suivant déplace des axes géométriques, mais pas le bloc précédent.
  - Avant le filetage, le bloc suivant a G33 comme fonction préparatoire, mais le bloc précédent non.
  - La bascule s'effectue entre BRISK et SOFT.
  - Des axes significatifs pour la transformation ne sont pas affectés entièrement à la trajectoire (par exemple les axes de positionnement pour l'oscillation)
- Le bloc d'arrondissement aurait pour effet de ralentir l'exécution du programme pièce.
  Cela se produit :
  - Entre des blocs très courts.
    Étant donné que chaque bloc a besoin d'au moins un cycle d'interpolation, la présence d'un bloc intermédiaire inséré doublerait le temps d'exécution.
  - Si un changement de bloc avec G64 (contournage sans arrondissement) peut avoir lieu sans réduction de vitesse.
    L'arrondissement augmenterait le temps d'usinage. Autrement dit la valeur du facteur de surcharge autorisé (MD32310 $MA_MAX_ACCEL_OVL_FACTOR) influe sur la décision d'arrondir ou non une transition entre blocs. Le facteur de surcharge est pris en compte uniquement dans l'arrondissement avec G641/G642. Lors de l'arrondissement avec G643, le facteur de surcharge n'a aucun effet (ce comportement peut aussi être réglé pour G641 et G642, par activation de
    MD20490 $MC_IGNORE_OVL_FACTOR_FOR_ADIS = TRUE).

- L'arrondissement n'est pas paramétré.
  Cette situation se présente quand :
  - ADISPOS=0 dans les blocs G0 pour G641 (réglage par défaut !).
  - ADIS=0 dans les blocs non G0 pour G641 (réglage par défaut !).
  - Lors de la transition entre un bloc G0 et un bloc non G0 ou vice-versa pour G641, c'est la plus petite valeur de ADISPOS et ADIS qui est effective.
  - Pour G642/G643, toutes les tolérances spécifiques à un axe sont égales à zéro.
- Le bloc ne contient pas de déplacement (bloc nul).
  Cette situation se présente quand :
  - Des actions synchrones sont actives.
    Normalement, les blocs nuls sont éliminés par l'interpréteur. Par contre, si des actions synchrones sont actives, ce bloc nul est pris en compte et exécuté. Cette prise en compte génère un arrêt précis conformément à la programmation active. Ainsi l'action synchrone doit avoir la possibilité de commuter si nécessaire.
  - Des sauts de programme génèrent des blocs nuls.

Contournage en rapide G0

Pour le déplacement en marche rapide, il faut également indiquer l'une des fonctions G60/G9 ou G64 ou G641 - G645 mentionnées. Si ce n’est pas le cas, c'est le préréglage spécifié dans le paramètre machine qui sera pris en compte.

2.12 Transformations des coordonnées (Frames)

2.12.1 Frames

Frame

Un frame est une règle opératoire qui transpose un système de coordonnées cartésiennes en un autre système de coordonnées cartésiennes.

Frame de base (décalage de base)

Le frame de base décrit la transformation des coordonnées du système de coordonnées de base (SCB) dans le système de coordonnées de base décalé (SCD) et agit comme les frames réglables.

Voir Système de coordonnées de base (SCB) (Page 38).

Frames réglables

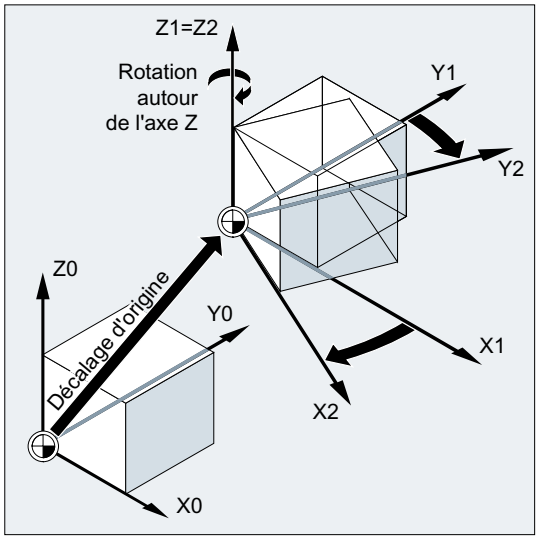
Les frames réglables correspondent aux décalages d'origine paramétrables pouvant être appelés avec les instructions `G54` à `G57` et `G505` à `G599` à partir d'un programme CN quelconque. Les valeurs des décalages sont préréglées par l'opérateur et enregistrées dans la mémoire des origines de la commande. Elles servent à définir le système de coordonnées de base réglable (SCR).

Voir :

- Système de coordonnées réglable (SCR) (Page 40)
- Décalage d'origine réglable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153) (Page 147)

Frames programmables

Dans certains cas, il peut s'avérer préférable, voire nécessaire, de procéder à l'intérieur d'un programme CN à un déplacement ou à une rotation du système de coordonnées pièce (ou du "système de coordonnées de base réglable") choisi initialement, voire d'appliquer une fonction miroir ou un changement d'échelle. Ceci s'effectue au moyen de frames programmables.



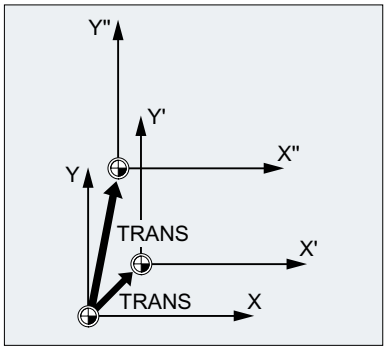
Voir Instructions de frame (Page 313).

2.12.2 Instructions de frame

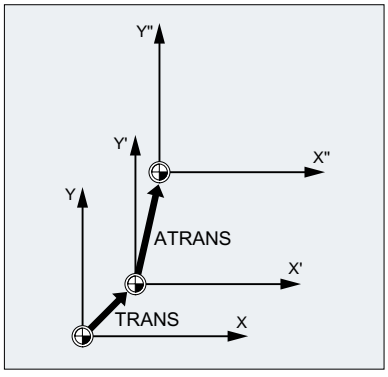
Fonction

Les instructions pour les frames programmables sont valables dans le programme CN actuel. Elles s'appliquent par addition ou substitution :

- Instruction substitutive
  Supprime toutes les instructions frame programmées auparavant. Le décalage d'origine réglable appelé en dernier (G54 ... G57, G505 ... G599) fait office de référence.

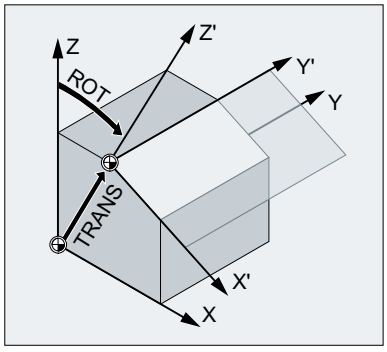


- Instruction additive
  S'ajoute aux frames existants. L'origine pièce actuellement réglée ou programmée en dernier avec une instruction frame fait office de référence.



Exemple d'application

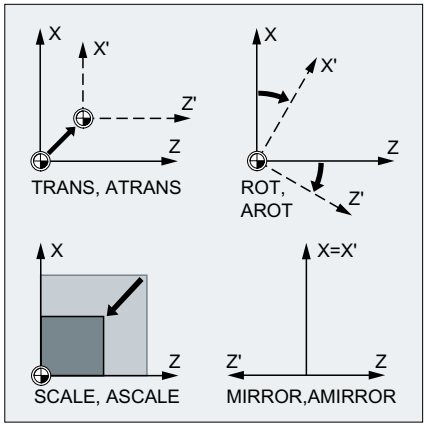
1. Décalage de l'origine du système de coordonnées pièces (SCP).
2. Rotation du système de coordonnées pièces (SCP) pour le pivotement d'un plan parallèlement au plan de travail souhaité.

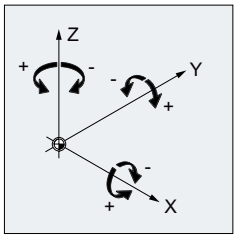


Syntaxe

Instructions substitutives	Instructions additives
TRANS X… Y… Z…	ATRANS X… Y… Z…
ROT X… Y… Z…	AROT X… Y… Z…
ROT RPL=…	AROT RPL=…
ROTS/CROTS X... Y...	AROTS X... Y...
SCALE X… Y… Z…	ASCALE X… Y… Z…
MIRROR X0/Y0/Z0	AMIRROR X0/Y0/Z0

Signification



TRANS/ATRANS :	Décalage SCP en direction de l'axe ou des axes géométriques indiqués		
ROT/AROT :	Rotation SCP : • autour de l'axe ou des axes géométriques indiqués par concaténation de rotations individuelles ou • autour de l'angle RPL=... dans le plan de travail actuel (G17/G18/G19)		
	Sens de rotation :		
	Ordre de rotation :	avec l'ordre RPY :	Z, Y', X''
		avec angles d'Euler :	Z, X', Z''
	Plage de valeurs :	La définition des angles de rotation est sans équivoque uniquement dans les plages suivantes :	
		avec l'ordre RPY :	-180 ≤ x ≤ 180
			-90 < y < 90
			-180 ≤ z ≤ 180
		avec angles d'Euler :	0 ≤ x < 180
			-180 ≤ y ≤ 180
			-180 ≤ z ≤ 180
ROTS/AROTS :	Rotation SCP par indication d'angles solides L'orientation d'un plan dans l'espace est déterminée de manière univoque par l'indication de deux angles solides. Au maximum 2 angles solides peuvent de ce fait être programmés : ROTS/AROTS X... Y... / Z... X... / Y... Z...		
CROTS :	CROTS agit comme ROTS, mais se rapporte au frame valide dans la base de données.		
SCALE/ASCALE :	Mise à l'échelle dans la direction de l'axe ou des axes géométriques indiqués, afin d'agrandir/réduire un contour		
MIRROR/AMIRROR :	Fonction miroir SCP par symétrie (changement de direction) de l'axe géométrique indiqué		
	Valeur :	de votre choix (dans le cas présent : "0")	

Autres conditions à prendre en compte

- Les instructions frame doivent être programmées dans un bloc CN spécifique.
- Les instructions frames peuvent être utilisées individuellement ou en combinaison quelconque.

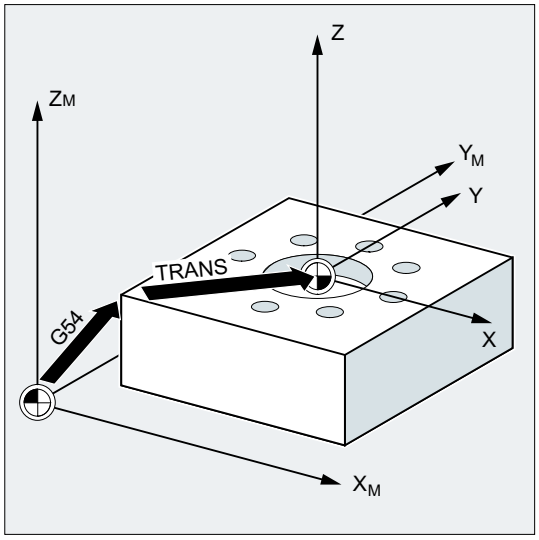
- Elles sont exécutées dans l'ordre de leur programmation.
- les instructions additives sont fréquemment utilisées dans des sous-programmes. Les instructions de base définies dans le programme principal sont réactivées après la fin du sous-programme si ce dernier a été programmé avec l'attribut SAVE.

2.12.3 Décalage d'origine programmable (TRANS, ATRANS)

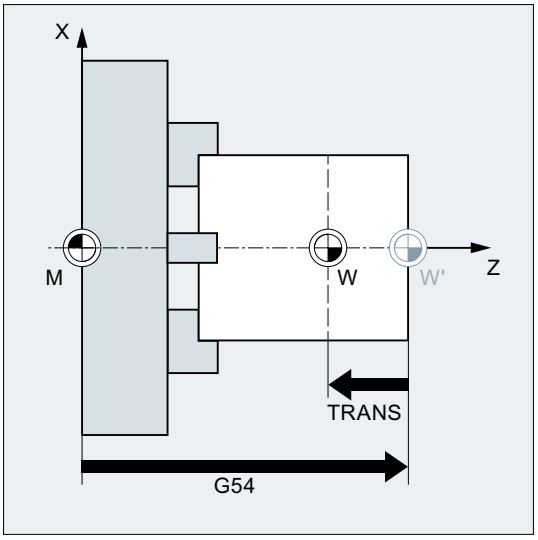
Avec l'instruction `TRANS`, le décalage du SCP rapporté au SCD créé avec un décalage d'origine réglable (G54 ... G57, G505 ... G599) est absolu.

Avec l'instruction `ATRANS`, le décalage du SCP créé avec `TRANS` est additif.

Fraisage :



Tournage :



Syntaxe

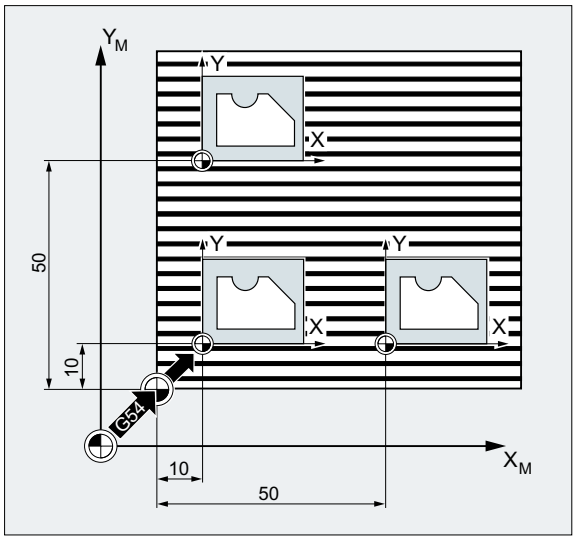
```
TRANS X... Y... Z...
ATRANS X... Y... Z...
```

Signification

`TRANS` :	Décalage absolu du SCP rapporté au système de coordonnées de base décalé (SCD) avec un décalage d'origine réglable (G54 ... G57, G505 ... G599)	
	Uniquement dans le bloc :	oui
`ATRANS` :	Décalage d'origine additif du SCP rapporté à l'origine pièce réglée avec `TRANS`	
	Uniquement dans le bloc :	oui
`X... Y... Z...` :	Valeurs de décalage en direction des axes géométriques indiqués	

Exemples

Exemple 1 : Fraisage



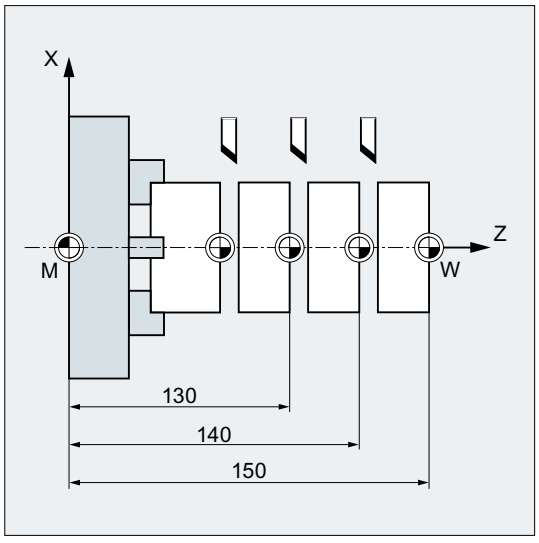
Sur cette pièce, les formes indiquées figurent plusieurs fois dans un programme.

La gamme opératoire pour cette forme figure dans le sous-programme.

Le décalage d'origine permet de spécifier les origines pièce respectivement requises, puis d'appeler le sous-programme.

Code de programme	**Commentaire**
N10 G1 G54	; Plan de travail X/Y, origine pièce
N20 G0 X0 Y0 Z2	; Accostage du point de départ
N30 TRANS X10 Y10	; Décalage absolu
N40 L10	; Appel du sous-programme
N50 TRANS X50 Y10	; Décalage absolu
N60 L10	; Appel du sous-programme
N70 M30	; Fin du programme

Exemple 2 : Tournage



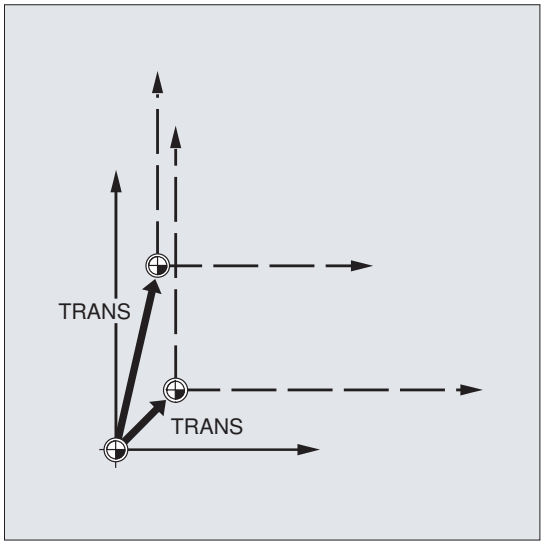
Code de programme	Commentaire
...	
N10 TRANS X0 Z150	; Décalage absolu
N15 L20	; Appel du sous-programme
N20 TRANS X0 Z140 (ou ATRANS Z-10)	; Décalage absolu
N25 L20	; Appel du sous-programme
N30 TRANS X0 Z130 (ou ATRANS Z-10)	; Décalage absolu
N35 L20	; Appel du sous-programme
...	

Informations complémentaires

TRANS X... Y... Z...

Décalage d'origine suivant les valeurs de décalage programmées dans les axes indiqués (axes à interpolation, axes de positionnement, axes synchrones). Le décalage d'origine réglable indiqué en dernier (G54 ... G57, G505 ... G599) fait office de référence.

IMPORTANT
Pas de frame d'origine L'instruction TRANS initialise toutes les composantes du frame programmable spécifié précédemment.

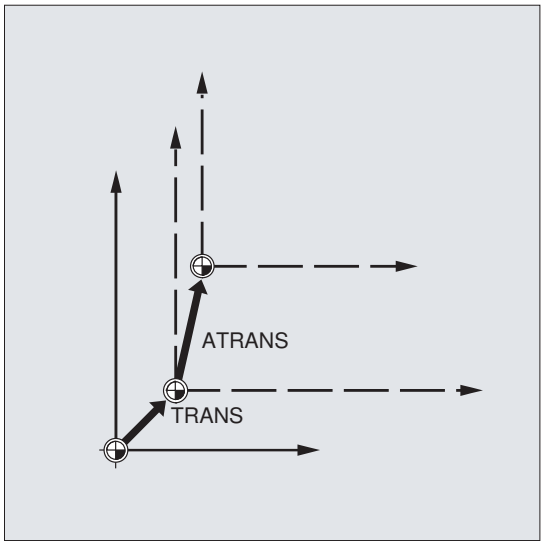


Remarque

Un décalage basé sur des frames existants doit être programmé avec`ATRANS`.

ATRANS X... Y... Z...

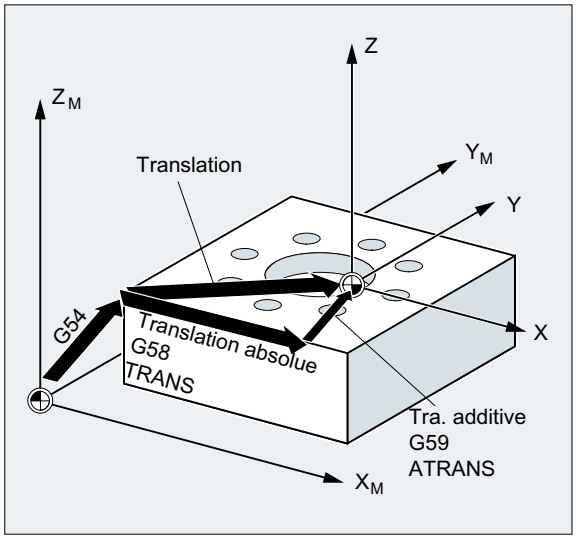
Décalage d'origine suivant les valeurs de décalage programmées dans les axes indiqués. L'origine momentanément en vigueur ou la dernière origine programmée fait office de référence.



2.12.4 Décalage d'origine programmable (G58, G59)

Les fonctions G58 et G59 permettent de remplacer des composantes de translation du décalage d'origine programmable (TRANS/ATRANS) (Page 316) de manière spécifique aux axes :

- G58 : composante de translation absolue (décalage grossier)
- G59 : composante de translation additive (décalage fin)



Conditions

Les fonctions G58 et G59 sont utilisables uniquement lorsque le décalage fin est configuré (PM24000 $MC_FRAME_ADD_COMPONENTS = 1).

Syntaxe

```
G58 <axe_1><valeur_1> ... <axe_3><valeur_3>
G59 <axe_1><valeur_1> ... <axe_3><valeur_3>
```

Signification

`G58` :	`G58` remplace la composante de translation absolue du décalage d'origine programmable pour l'axe indiqué, le décalage additif programmé étant conservé. Le décalage d'origine réglable appelé en dernier (`G54` ... `G57`, `G505` ... `G599`) fait office de référence.
	Uniquement dans le bloc : oui
`G59` :	`G59` remplace la composante de translation additive du décalage d'origine programmable pour l'axe indiqué, le décalage absolu programmé étant conservé.
	Uniquement dans le bloc : oui
`<axe_n>` :	Axe géométrique dans le canal
`<valeur_n>` :	Valeurs de décalage en direction de l'axe géométrique indiqué

Exemple

Code de programme	Commentaire
...	
N50 TRANS X10 Y10 Z10	; Composante de translation absolue X10 Y10 Z10
N60 ATRANS X5 Y5	; Composante de translation additive X5 Y5
	→ Décalage total : X15 Y15 Z10
N70 G58 X20	; Composante de translation absolue X20
	→ Décalage total X25 Y15 Z10
N80 G59 X10 Y10	; Composante de translation additive X10 Y10
	→ Décalage total X30 Y20 Z10
...	

Informations complémentaires

La composante de translation absolue (**décalage grossier**) est modifiée par les instructions suivantes :

- TRANS
- G58
- CTRANS
- CFINE
- $P_PFRAME[X,TR]

La composante de translation additive (**décalage fin**) est modifiée par les instructions suivantes :

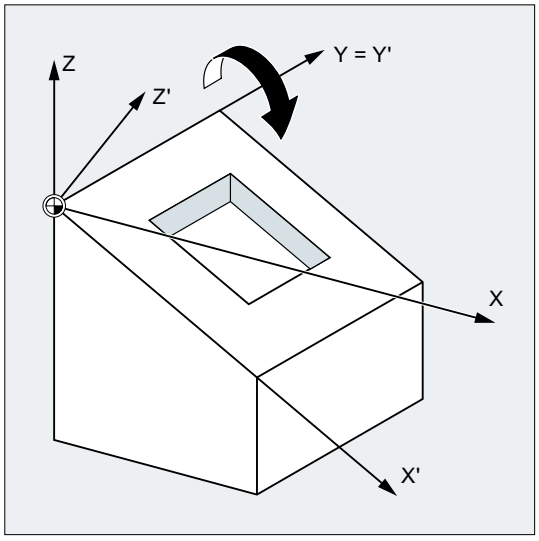
- ATRANS
- G59
- CTRANS
- CFINE
- $P_PFRAME[X,FI]

Exemples

Instruction	Décalage grossier V_C	Décalage fin V_F
TRANS X10	$V_C = 10$	inchangé
G58 X10	$V_C = 10$	inchangé
$P_PFRAME[X,TR] = 10	$V_C = 10$	inchangé
ATRANS X10	inchangé	$V_F = V_F + 10$
G59 X10	inchangé	$V_F = 10$
$P_PFRAME[X,FI] = 10	inchangé	$V_F = 10$
CTRANS(X,10)	$V_C = 10$	$V_F = 0$
CTRANS()	$V_C = 0$	$V_F = 0$
CFINE(X,10)	$V_C = 0$	$V_F = 10$

2.12.5 Rotation programmable (ROT, AROT, RPL)

Les instructions `ROT` / `AROT` permettent de faire pivoter le système de coordonnées pièce dans l'espace. Les instructions se rapportent exclusivement au frame programmable `$P_PFRAME`.



Syntaxe

```
ROT <1er GeoAx><Angle> <2e GeoAx><Angle> <3e GeoAx><Angle>
ROT RPL=<Angle>
AROT <1er GeoAx><Angle> <2e GeoAx><Angle> <3e GeoAx><Angle>
AROT RPL=<Angle>
```

Remarque

Les pivotements du système de coordonnées pièce sont effectués à l'aide des angles d'Euler.

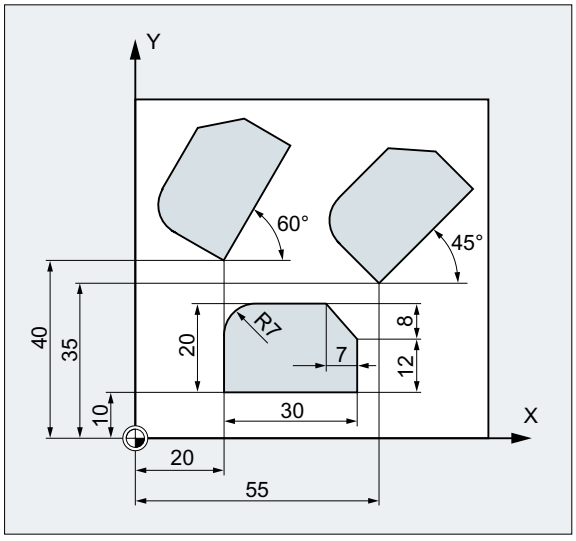
Signification

`ROT` :	Rotation absolue	
	Frame de référence :	Frame programmable `$P_PFRAME`
	Point de référence :	Origine du système de coordonnées pièce actuel réglé avec `G54` ... `G57`, `G505` ... `G599`
`AROT` :	Rotation additive	
	Frame de référence :	Frame programmable `$P_PFRAME`
	Point de référence :	Origine du système de coordonnées pièce actuel réglé avec `G54` ... `G57`, `G505` ... `G599`
`<n-ième AxeGéo>` :	Descripteur du n-ième axe géométrique autour duquel une rotation de l'angle indiqué doit être effectuée. Pour les axes géométriques non programmés, la valeur 0° est réglée implicitement comme angle de rotation.	

RPL :	Rotation de l'angle indiqué autour de l'axe géométrique situé à la verticale du plan actif (G17, G18, G19).	
	Frame de référence :	Frame programmable $P_PFRAME
	Point de référence :	Origine du système de coordonnées pièce actuel réglé avec G54 ... G57, G505 ... G599
<Angle>	Indication de l'angle en degrés	
	Plage de valeurs :	-360° ≤ angle ≤ 360°

Exemples

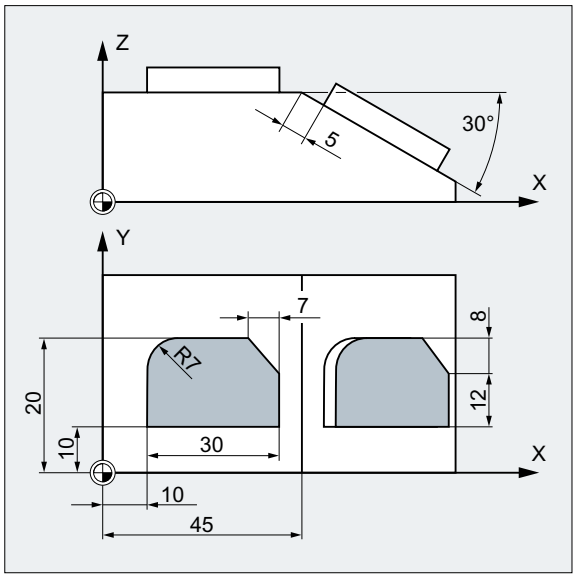
Exemple 1 : Rotation dans le plan G17



Sur cette pièce, les formes indiquées figurent plusieurs fois dans un programme. En plus du décalage d'origine, il est nécessaire d'effectuer des rotations, car les formes ne sont pas disposées parallèlement aux axes.

Code de programme	Commentaire
N10 G17 G54	; Plan de travail X/Y, origine pièce
N20 TRANS X20 Y10	; Décalage absolu
N30 L10	; Appel du sous-programme
N40 TRANS X55 Y35	; Décalage absolu
N50 AROT RPL=45	; Rotation additive de 45° autour de l'axe Z
	; situé à la verticale du plan G17
N60 L10	; Appel du sous-programme
N70 TRANS X20 Y40	; Décalage absolu
	; (efface tous les décalages précédents)
N80 AROT RPL=60	; Rotation additive de 60° autour de l'axe Z
	; situé à la verticale du plan G17
N90 L10	; Appel du sous-programme
N100 G0 X100 Y100	; Dégagement
N110 M30	; Fin du programme

Exemple 2 : Rotation dans l'espace autour de l'axe Y

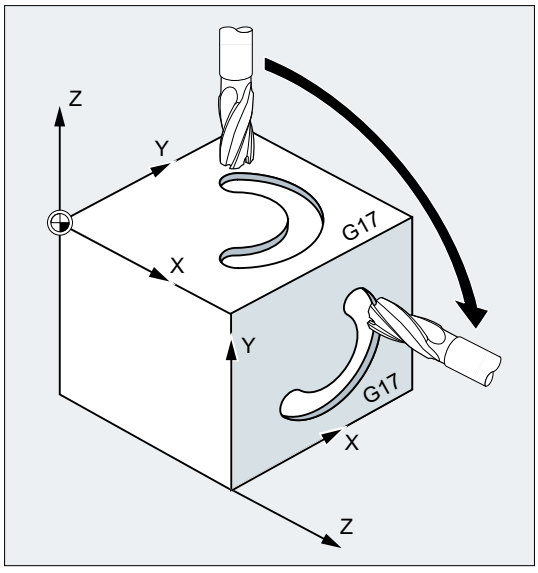


Dans cet exemple, il s'agit d'usiner des surfaces obliques et des surfaces parallèles aux axes, en un seul ablocage.

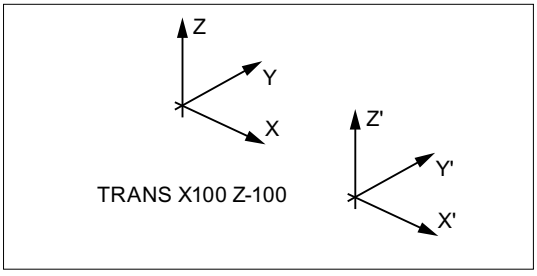
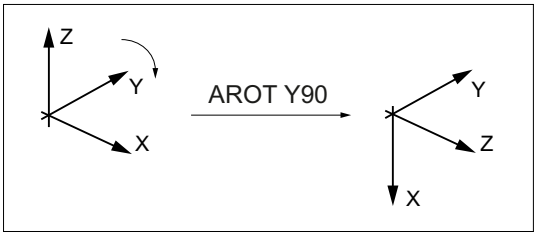
Condition :
L'outil doit être positionné perpendiculairement à la surface oblique dans la direction Z pivotée.

Code de programme	**Commentaire**
N10 G17 G54	; Plan de travail X/Y, origine pièce
N20 TRANS X10 Y10	; Décalage absolu
N30 L10	; Appel du sous-programme
N40 ATRANS X35	; Décalage additif
N50 AROT Y30	; Rotation additive autour de l'axe Y
N60 ATRANS X5	; Décalage additif
N70 L10	; Appel du sous-programme
N80 G0 X300 Y100 M30	; Dégagement, fin de programme

Exemple 3 : Usinage multiface



Dans cet exemple, des formes identiques sont exécutées par des sous-programmes sur deux faces de la pièce, perpendiculaires l'une par rapport à l'autre. Dans le nouveau système de coordonnées sur la face droite de la pièce, la direction de pénétration, le plan de travail et l'origine se présentent comme sur la face supérieure. De ce fait, les conditions nécessaires pour l'exécution du sous-programme s'appliquent à nouveau : plan de travail `G17`, plan de coordonnées X/Y, direction de pénétration Z.

Code de programme	**Commentaire**
`N10 G17 G54`	`; Plan de travail X/Y, origine pièce`
`N20 L10`	`; Appel du sous-programme`
`N30 TRANS X100 Z-100`	`; Décalage absolu du SCP`
	
`N40 AROT Y90`	`; Rotation additive du SCP de 90° autour de Y`
	

Code de programme	Commentaire
N50 AROT Z90	; Rotation additive du SCP de 90° autour de Z
N60 L10	; Appel du sous-programme
N70 G0 X300 Y100 M30	; Dégagement, fin de programme

Informations complémentaires

Rotation dans le plan actif

Pour une programmation au moyen de `RPL=...`, le SCP pivote autour de l'axe situé à la verticale du plan actif.

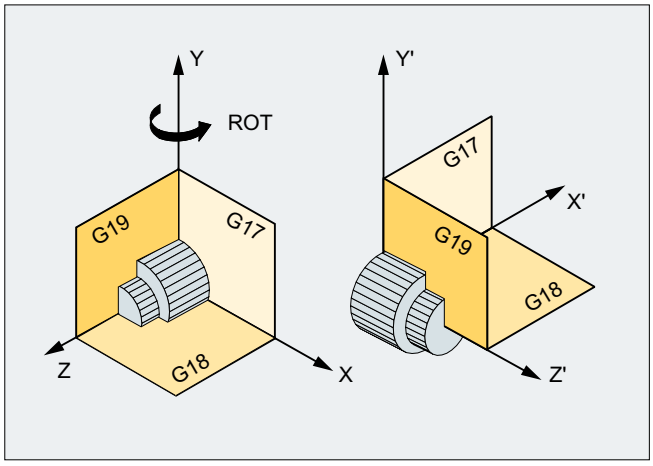


Figure 2-23 Rotation autour de l'axe Y ou dans le plan G18

⚠ ATTENTION

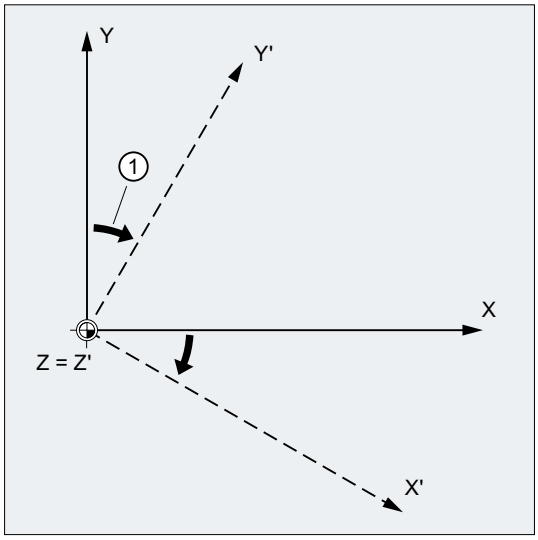
changement de plan

Si un changement de plan (`G17`, `G18`, `G19`) est programmé après une rotation, les angles de rotation actuels des axes respectifs sont conservés et s'appliquent également dans le nouveau plan. De ce fait, nous recommandons vivement de remettre à 0 les angles de rotation actuels avant tout changement de plan :

- `N100 ROT X0 Y0 Z0 ; programmation d'angle explicite`
- `N100 ROT ; programmation d'angle implicite`

Rotation absolue avec ROT X... Y... Z...

Le SCP effectue une rotation autour des axes indiqués aux angles de rotation programmés.

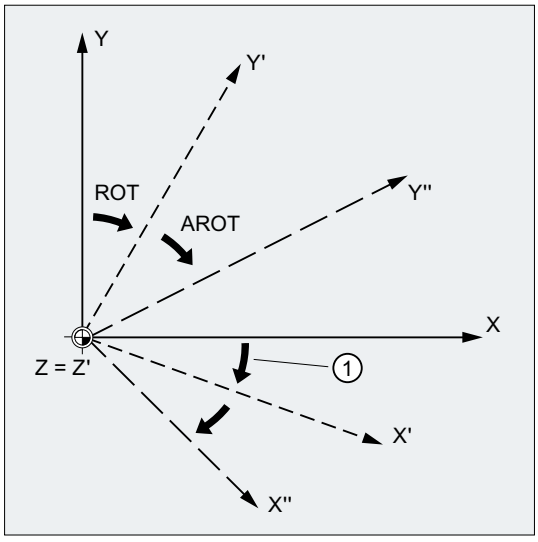


① Angle de rotation

Figure 2-24 Rotation absolue autour de l'axe Z

Rotation additive avec AROT X... Y... Z...

Le SCP effectue une rotation supplémentaire autour des axes indiqués suivant les valeurs des angles de rotation programmés.



① Angle de rotation

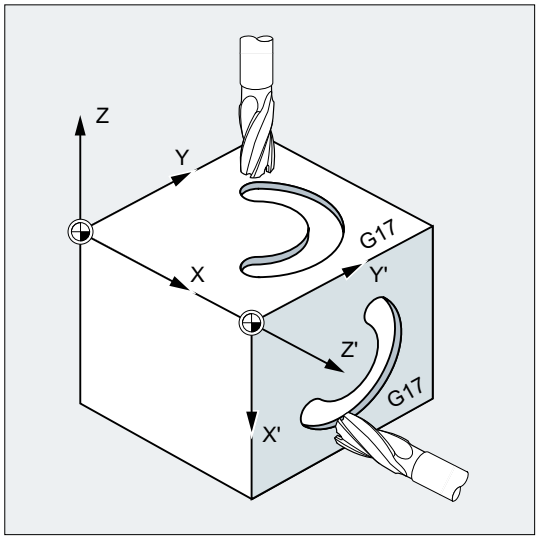
Figure 2-25 Rotation absolue et additive autour de l'axe Z

Rotation du plan de travail

Pour une rotation avec ROT/AROT, le plan de travail (G17, G18, G19) pivote également.

Exemple : Plan de travail G17
Le SCP se trouve sur la face supérieure de la pièce. Par décalage et rotation, le système de coordonnées est décalé sur l'une des faces latérales. Le plan de travail G17 pivote également. De ce fait, on peut continuer à programmer des déplacements dans le plan G17 via X et Y et des pénétrations via Z.

Condition :
L'outil doit être perpendiculaire au plan de travail et le sens positif de l'axe de pénétration est dirigé vers le porte-outil.
Avec l'indication de `CUT2DF`, la correction du rayon d'outil s'applique au plan pivoté.



2.12.6 Programmation de rotations de frames avec des angles solides (ROTS, AROTS, CROTS)

Les instructions `ROTS`, `AROTS` et `CROTS` permettent de spécifier des rotations du système de coordonnées pièce en angles solides. Les angles solides sont des angles qui forment des lignes d'intersection du plan souhaité, pivoté dans l'espace, avec les plans principaux du SCP non encore pivoté

Remarque

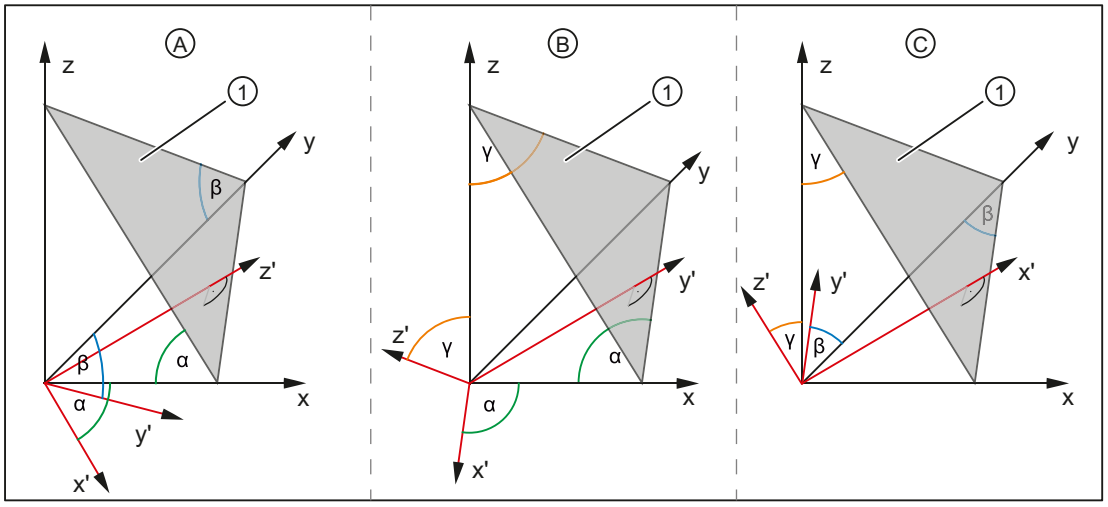
Descripteurs d'axes géométriques

A titre d'exemple, dans la description ci-après la définition suivante s'applique :

- 1er axe géométrique : X
- 2e axe géométrique : Y
- 3e axe géométrique : Z

Par exemple, comme le montre la figure ci-après, la programmation `ROTS Xα Yβ` entraîne une orientation du plan G17 du SCP parallèlement au plan oblique représenté. La position de l'origine du SCP reste alors inchangée.

L'orientation du SCP pivoté est définie de telle sorte que le premier axe pivoté se trouve dans le plan formé par celui-ci et le 3e axe du système de coordonnées d'origine. Dans notre exemple : X' se trouve dans le plan X/Z d'origine.



①	Plan oblique
α, β, γ	Angles solides
A	Orientation du plan G17 parallèlement au plan oblique :

- 1re rotation
  Rotation de x autour de y de l'angle α ⇒
  axe x' parallèle au plan oblique
- 2e rotation
  Rotation de y' autour de x' de β ⇒
  axe y' parallèle au plan oblique
  ⇒ axe z' perpendiculaire au plan oblique
  ⇒ G17 parallèle au plan oblique

B Orientation du plan G18 parallèlement au plan oblique :

- 1re rotation
  Rotation de z autour de x de l'angle γ ⇒
  axe z' parallèle au plan oblique
- 2e rotation
  Rotation de x' autour de z' de l'angle α ⇒
  axe x' parallèle au plan oblique
  ⇒ axe y' perpendiculaire au plan oblique
  ⇒ G18 parallèle au plan oblique

C Orientation du plan G19 parallèlement au plan oblique :

- 1re rotation
  Rotation de y autour de z de l'angle β ⇒
  axe y' parallèle au plan oblique
- 2e rotation
  Rotation de z' autour de y' de l'angle γ ⇒
  axe z' parallèle au plan oblique
  ⇒ axe x' perpendiculaire au plan oblique
  ⇒ G19 parallèle au plan oblique

Syntaxe

Règles

La position d'un plan dans l'espace est déterminée de manière univoque par deux angles solides. Le plan serait surdéterminé si un troisième angle solide était indiqué. Cela est donc inadmissible.

Lorsqu'un seul angle solide est programmé, la rotation du SCP est réalisée de la même façon que pour `ROT`, `AROT` (voir chapitre "Rotation programmable (ROT, AROT, RPL) (Page 322)").

Les deux axes programmés définissent un plan selon les définitions de plan pour `G17`, `G18`, `G19`. L'ordre des axes de coordonnées (1er axe / 2e axe du plan) ou l'ordre des rotations autour des angles solides est ainsi défini :

Plan	1er axe	2e axe
G17	X	Y
G18	Z	X
G19	Y	Z

Orientation du plan G17 ⇒ angles solides pour X et Y

- 1re rotation : X autour de Y de l'angle α
- 2e rotation : Y autour de X' de l'angle β
- Orientation : X' se trouve dans le plan Z/X d'origine.

```
ROTS  X<α> Y<β>
AROTS X<α> Y<β>
CROTS X<α> Y<β>
```

Orientation du plan G18 ⇒ angles solides pour Z et X

- 1re rotation : Z autour de X de l'angle γ
- 2e rotation : X autour de Z' de l'angle α
- Orientation : Z' se trouve dans le plan Y/Z d'origine

```
ROTS  Z<γ> X<α>
AROTS Z<γ> X<α>
CROTS Z<γ> X<α>
```

Orientation du plan G19 ⇒ angles solides pour Y et Z

- 1re rotation : Y autour de Z de l'angle β
- 2e rotation : Z autour de Y' de l'angle γ
- Orientation : Y' se trouve dans le plan X/Y d'origine.

```
ROTS  Y<β> Z<γ>
AROTS Y<β> Z<γ>
CROTS Y<β> Z<γ>
```

Signification

ROTS :	Rotations de frame avec angles solides absolus, Frame de référence : Frame programmable $P_PFRAME
AROTS :	Rotations de frame avec angles solides additifs, Frame de référence : Frame programmable $P_PFRAME
CROTS :	Rotations de frame avec angles solides absolus, Frame de référence : Frame $P_... programmé
X, Y, Z :	Descripteurs d'axes géométriques (voir remarque ci-dessus : Descripteurs d'axes géométriques)
α, β, γ :	Angle solide se rapportant à l'axe géométrique correspondant : • α → X • β → Y • γ → Z

2.12.7 Facteur d'échelle programmable (SCALE, ASCALE)

SCALE/ASCALE permettent de programmer, pour tous les axes à interpolation, axes synchrones et axes de positionnement, des facteurs d'échelle pour l'agrandissement ou la réduction dans la direction des axes respectifs indiqués. Il est ainsi possible de prendre en compte des formes analogues aux formes géométriques ou des taux d'évanouissement lors de la programmation.

Syntaxe

```
SCALE X... Y... Z...
ASCALE X... Y... Z...
```

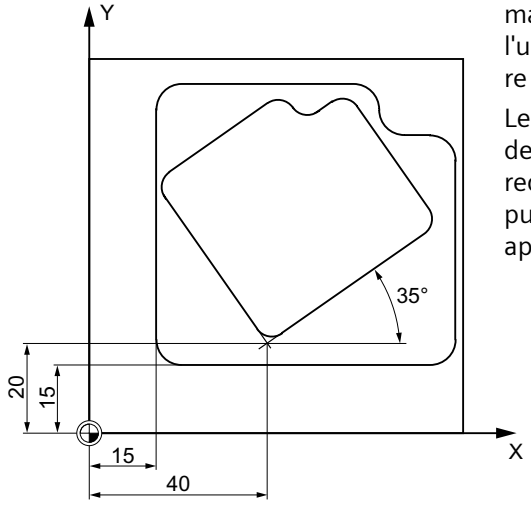
Remarque

La programmation des instructions frame s'effectue respectivement dans un bloc CN spécifique.

Signification

SCALE :	Agrandissement / réduction absolus, rapportés au système de coordonnées en vigueur, réglé avec G54 ... G57, G505 ... G599
ASCALE :	Agrandir / réduire de façon additive, par rapport au système de coordonnées courant, réglé ou programmé
X... Y... Z... :	Facteurs d'échelle dans la direction des axes géométriques indiqués

Exemple



Cette pièce présente deux fois la même poche, mais avec des tailles différentes et pivotées l'une par rapport à l'autre. La gamme opératoire figure dans le sous-programme.

Le décalage d'origine et la rotation permettent de spécifier les origines pièce respectivement requises, la mise à l'échelle réduit le contour, puis le sous-programme peut à nouveau être appelé.

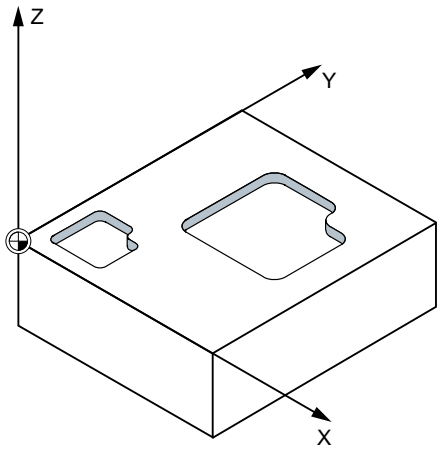
Code de programme	Commentaire
N10 G17 G54	; Plan de travail X/Y, origine pièce
N20 TRANS X15 Y15	; Décalage absolu
N30 L10	; Exécuter la grande poche
N40 TRANS X40 Y20	; Décalage absolu
N50 AROT RPL=35	; Rotation dans le plan de 35°
N60 ASCALE X0.7 Y0.7	; Facteur d'échelle pour la petite poche
N70 L10	; Exécuter la petite poche
N80G0 X300 Y100 M30	; Dégagement, fin de programme

Informations complémentaires

SCALE X... Y... Z...

Pour chaque axe, vous pouvez indiquer un facteur d'échelle qui entraînera un agrandissement ou une réduction. Le facteur d'échelle se rapporte au système de coordonnées pièce réglé avec G54 ... G57, G505 ... G599.

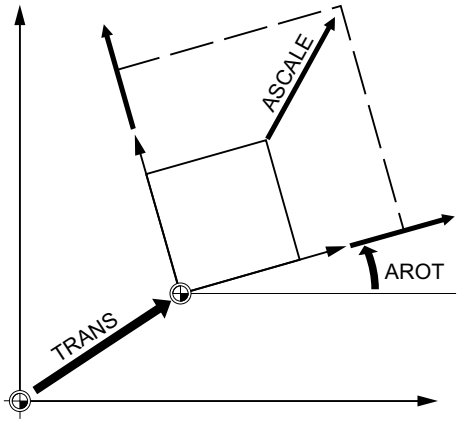
IMPORTANT
Pas de frame d'origine
L'instruction SCALE réinitialise toutes les composantes du frame programmable spécifié précédemment.



ASCALE X... Y... Z...

Un changement d'échelle qui fait appel à des frames existants doit être programmé avec ASCALE. Dans ce cas, le dernier facteur valide sera multiplié par le nouveau facteur d'échelle.

La référence pour le changement d'échelle est le système de coordonnées actuellement réglé ou le dernier système de coordonnées programmé.



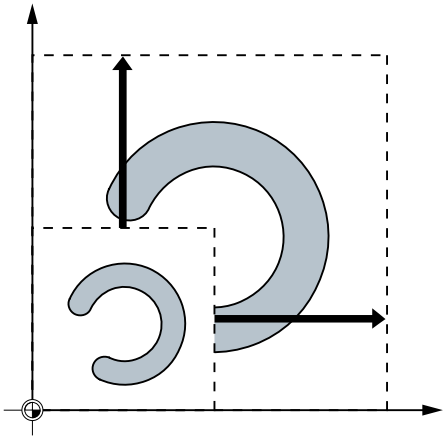
Mise à l'échelle et décalage

Remarque

Si un décalage est programmé avec ATRANS après SCALE, les valeurs de décalage sont également mises à l'échelle.

Différents facteurs d'échelle

IMPORTANT
Risque de collision Prudence quand vous utilisez différents facteurs d'échelle ! Les interpolations circulaires peuvent par ex. uniquement être mises à l'échelle avec des facteurs identiques.



Remarque

Pour la programmation de cercles déformés, il est toutefois possible d'utiliser différents facteurs d'échelle.

2.12.8 Fonction miroir programmable (MIRROR, AMIRROR)

`MIRROR`/`AMIRROR` permettent de créer des formes de pièces symétriques par rapport aux axes de coordonnées. Tous les déplacements programmés ultérieurement, p. ex. dans le sous-programme, sont exécutés avec la fonction miroir.

Syntaxe

```
MIRROR X... Y... Z...
AMIRROR X... Y... Z...
```

Remarque

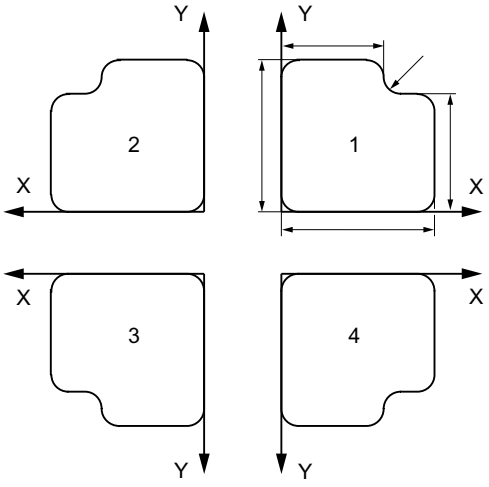
La programmation des instructions frame s'effectue respectivement dans un bloc CN spécifique.

Signification

`MIRROR:`	Fonction miroir absolue, rapportée au système de coordonnées en vigueur, réglé avec G54 ... G57, G505 ... G599
`AMIRROR:`	Fonction miroir additive, par rapport au système de coordonnées courant, réglé ou programmé
`X... Y... Z...:`	Axe géométrique dont le sens doit être permuté. La valeur indiquée ici peut être choisie librement, par exemple X0 Y0 Z0.

Exemples

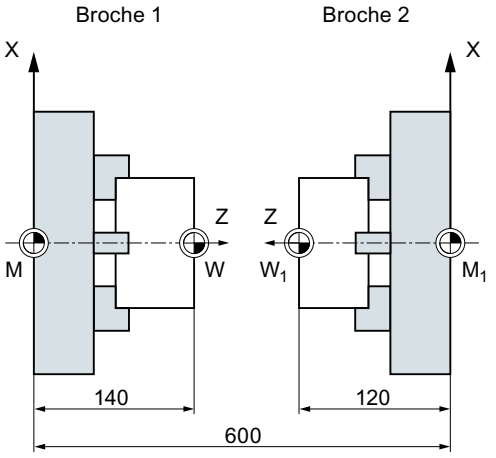
Exemple 1 : Fraisage



Le contour indiqué ici est programmé une seule fois comme sous-programme. Les trois autres contours sont obtenus avec la fonction miroir. L'origine pièce est située au centre des contours.

Code de programme	**Commentaire**
`N10 G17 G54`	`; Plan de travail X/Y, origine pièce`
`N20 L10`	`; Exécuter le premier contour en haut à droite`
`N30 MIRROR X0`	`; Fonction miroir de l'axe X (le sens est inversé en X)`
`N40 L10`	`; Exécuter le second contour en haut à gauche`
`N50 AMIRROR Y0`	`; Fonction miroir de l'axe Y (le sens est inversé en Y)`
`N60 L10`	`; Exécuter le troisième contour en bas à gauche`
`N70 MIRROR Y0`	`; MIRROR efface les frames antérieurs Fonction miroir de l'axe Y (le sens est inversé en Y)`
`N80 L10`	`; Exécuter le quatrième contour en bas à droite`
`N90 MIRROR`	`; Désactiver la fonction miroir`
`N100 G0 X300 Y100 M30`	`; Dégagement, fin de programme`

Exemple 2 : Tournage



L'usinage proprement dit figure dans un sous-programme, l'exécution sur les différentes broches est réalisée avec des fonctions miroir et des décalages.

Code de programme	Commentaire
N10 TRANS X0 Z140	; Décalage de l'origine en W
...	; Usinage du 1er côté avec la broche 1
N30 TRANS X0 Z600	; Décalage de l'origine à la broche 2
N40 AMIRROR Z0	; Fonction miroir de l'axe Z
N50 ATRANS Z120	; Décalage de l'origine en W1
...	; Usinage du 2e côté avec la broche 2

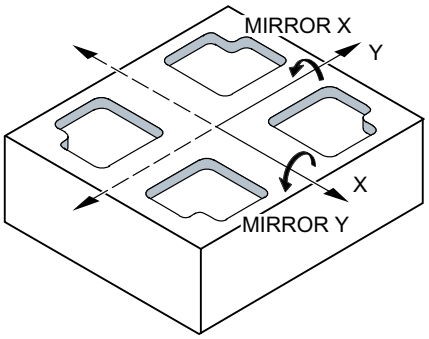
Informations complémentaires

MIRROR X... Y... Z...

La fonction miroir est programmée par le changement de sens axial dans le plan de travail sélectionné.

Exemple : plan de travail G17 X/Y

La fonction miroir (par rapport à l'axe Y) requiert un changement de sens en X, puis est programmée avec `MIRROR X0`. Le contour est alors usiné symétriquement du côté opposé de l'axe de symétrie Y.

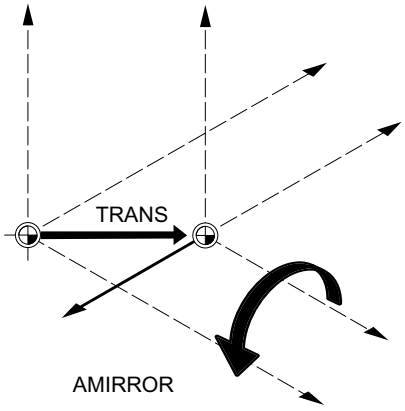


La fonction miroir est rapportée au système de coordonnées en vigueur, réglé avec G54 ... G57, G505 ... G599.

IMPORTANT
Pas de frame d'origine L'instruction `MIRROR` réinitialise toutes les composantes du frame programmable spécifié précédemment.

AMIRROR X... Y... Z...

Une fonction miroir qui vient s'ajouter à des transformations existantes est programmée avec `AMIRROR`. La référence est le système de coordonnées momentanément réglé ou le dernier système de coordonnées programmé.



Désactiver la fonction miroir

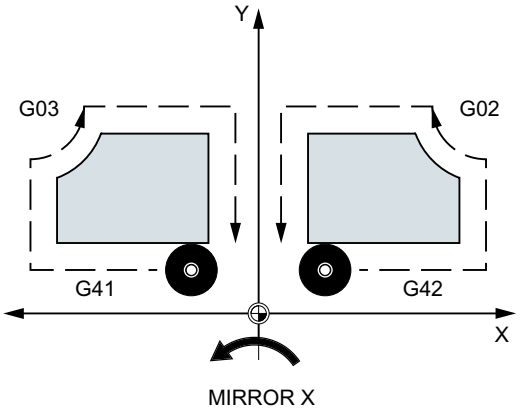
Pour tous les axes : `MIRROR` (sans indication d'axe)

Toutes les composantes du frame programmé précédemment sont effacées.

Correction du rayon d'outil

Remarque

Avec l'instruction miroir, la commande modifie automatiquement les instructions de correction de la trajectoire (G41/G42 ou G42/G41) en fonction du nouveau sens d'usinage.



Il en va de même pour le sens de l'interpolation circulaire (G2/G3 ou G3/G2).

Remarque

Si après MIRROR, une rotation additive est programmée avec AROT, il faudra, selon le cas, utiliser des sens de rotation inversés (positif/négatif ou négatif/positif). Les fonctions miroir appliquées aux axes géométriques sont converties automatiquement par la commande en rotations et, le cas échéant, en fonctions miroir par rapport à l'axe de symétrie réglable par paramètre machine. Il en va de même pour les décalages d'origine réglables.

Axe de la fonction miroir

Un paramètre machine permet de régler l'axe par rapport auquel s'exécute la fonction miroir :

PM10610 $MN_MIRROR_REF_AX = <valeur>

Valeur	Signification
0	La fonction miroir s'exécute par rapport à l'axe programmé (négation des valeurs).
1	L'axe X est l'axe de référence.
2	L'axe Y est l'axe de référence.
3	L'axe Z est l'axe de référence.

Interprétation des valeurs programmées

Un paramètre machine permet de régler la façon d'interpréter les valeurs programmées :

PM10612 $MN_MIRROR_TOGGLE = <valeur>

Valeur	Signification
0	Les valeurs d'axes programmées ne sont pas exploitées.
1	Les valeurs d'axes programmées sont exploitées : • Si la fonction miroir n'a pas encore été appliquée à l'axe, elle est exécutée lorsque les valeurs d'axe programmées sont ≠ 0. • Pour une valeur d'axe programmée = 0, la fonction miroir est désactivée.

2.12.9 Génération d'un frame selon l'orientation de l'outil (TOFRAME, TOROT, PAROT)

TOFRAME génère un système de coordonnées cartésien, dont l'axe Z coïncide avec l'orientation actuelle de l'outil. L'utilisateur a ainsi la possibilité de dégager l'outil sans collision en direction Z (p. ex. après une rupture de pièce dans le cas d'un programme à 5 axes).

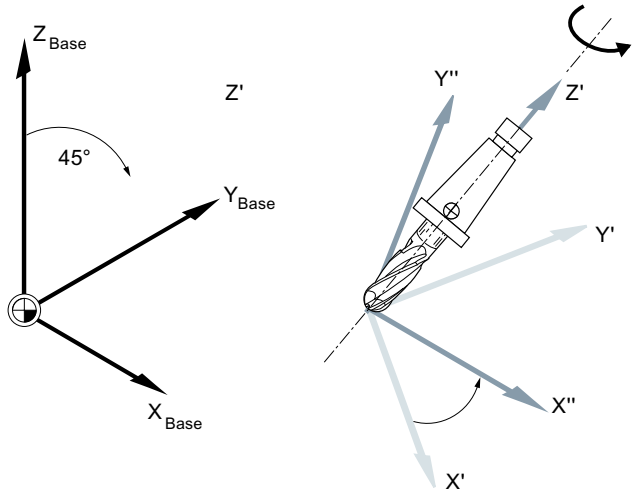
La position des deux axes X et Y dépend du réglage dans le paramètre machine PM21110 $MC_X_AXES_IN_OLD_X_Z_PLANE (système de coordonnées pour la définition automatique du frame). Le nouveau système de coordonnées est soit conservé tel qu'il résulte de la cinématique de la machine, soit une rotation supplémentaire lui est appliquée par rapport au nouvel axe Z, de sorte que le nouvel axe X se situe dans l'ancien plan Z-X (voir les indications du constructeur de machine).

Le frame résultant, décrivant l'orientation, figure dans la variable système du frame programmable ($P_PFRAME).

TOROT écrase uniquement la composante de rotation dans le frame programmé. Toutes les autres composantes restent inchangées.

TOFRAME et TOROT sont dimensionnés pour les opérations de fraisage, pour lesquelles G17 est typiquement actif (plan de travail X/Y). Pour les opérations de rotation ou, d'une façon générale, lorsque G18 ou G19 sont actifs, il faut par contre utiliser des frames, dans lesquels l'axe X ou Y coïncide avec l'orientation de la pièce. La programmation de ces frames s'effectue avec les instructions TOFRAMEX/TOROTX ou TOFRAMEY/TOROTY.

PAROT permet d'orienter le système de coordonnées pièce (SCP) sur la pièce.



Syntaxe

```
TOFRAME/TOFRAMEZ/TOFRAMEY/TOFRAMEX
...
TOROTOF

TOROT/TOROTZ/TOROTY/TOROTX
...
TOROTOF

PAROT
...
PAROTOF
```

Signification

TOFRAME :	Orienter l'axe Z du SCP parallèlement à l'orientation de l'outil par rotation du frame
TOFRAMEZ :	Identique à TOFRAME
TOFRAMEY :	Orienter l'axe Y du SCP parallèlement à l'orientation de l'outil par rotation du frame
TOFRAMEX :	Orienter l'axe X du SCP parallèlement à l'orientation de l'outil par rotation du frame
TOROT :	Orienter l'axe Z du SCP parallèlement à l'orientation de l'outil par rotation du frame La rotation définie par TOROT est la même que celle définie par TOFRAME.
TOROTZ :	Identique à TOROT
TOROTY :	Orienter l'axe Y du SCP parallèlement à l'orientation de l'outil par rotation du frame
TOROTX :	Orienter l'axe X du SCP parallèlement à l'orientation de l'outil par rotation du frame
TOROTOF :	Désactiver l'orientation parallèle à l'orientation de l'outil

PAROT :	Orienter le SCP sur la pièce par rotation du frame Les translations, les homothéties et les fonctions miroir du frame actif sont conservées.
PAROTOF :	La rotation de frame par rapport à l'outil, activée avec PAROT, est désactivée avec PAROTOF.

Remarque

L'instruction TOROT permet de réaliser une programmation cohérente pour des organes porte-outil orientables quelle que soit la cinématique.

De la même façon que pour un organe porte-outil pivotable, PAROT permet d'activer une rotation de la table de travail. Ceci permet de définir un frame qui modifie la position du système de coordonnées pièce de telle façon qu'aucun mouvement compensatoire de la machine n'intervient. L'instruction PAROT n'est pas rejetée si aucun organe porte-outil pivotable n'est actif.

Exemple

Code de programme	Commentaire
N100 G0 G53 X100 Z100 D0	
N120 TOFRAME	
N140 G91 Z20	; TOFRAME est pris en compte, tous les déplacements d'axes géométriques programmés se réfèrent au nouveau système de coordonnées.
N160 X50	
...	

Informations complémentaires

Affectation de la direction d'axe

Si l'une des instructions TOFRAMEX, TOFRAMEY, TOROTX ou TOROTY est programmée à la place de TOFRAME / TOFRAMEZ ou TOROT / TOROTZ, les affectations suivantes s'appliquent aux orientations d'axe, comme indiqué dans le tableau ci-après :

Instruction	Direction de l'outil (applications)	Axe auxiliaire (abscisse)	Axe auxiliaire (ordonnée)
TOFRAME / TOFRAMEZ / TOROT / TOROTZ	Z	X	Y
TOFRAMEY / TOROTY	Y	Z	X
TOFRAMEX / TOROTX	X	Y	Z

Frame système propre pour TOFRAME ou TOROT

Les frames générés par TOFRAME ou TOROT peuvent être écrits dans un frame système spécifique $P_TOOLFRAME. A cet effet, le bit 3 doit être mis à 1 dans le paramètre machine MD28082 $MC_MM_SYSTEM_FRAME_MASK. Le frame programmable reste inchangé. Des différences se manifestent ultérieurement quand le frame programmable est modifié.

2.12.10 Désactivation du frame (G53, G153, SUPA, G500)

Pendant l'exécution de certains processus tels que l'accostage du point de changement d'outil, différents composants de frame doivent être définis et inhibés à certains moments.

Les frames réglables peuvent être soit désactivés en mode modal, soit inhibés en mode non modal.

Les frames programmables peuvent être inhibés ou supprimés en mode non modal.

Syntaxe

```
G53
G153
SUPA
G500
TRANS
ROT
SCALE
MIRROR
```

Signification

`G53` :	Inhibition à effet non modal de tous les frames programmables et réglables
`G153` :	`G153` agit comme `G53` et inhibe de plus le frame de base global ($P_ACTBFRAME).
`SUPA` :	`SUPA` agit comme `G153` et désactive en plus les déplacements suivants : • Décalages par manivelle (DRF) • Déplacements forcés • Décalage d'origine externe • Décalage PRESET
`G500` :	Désactivation à effet modal de tous les frames réglables (G54 ... G57, G505 ... G599), si G500 ne contient aucune valeur.
`TRANS/ROT/SCALE/MIRROR` :	sans indication d'axe, entraîne la suppression des frames programmables.

2.12.11 Programmation : Désélection de corrections spécifique à un axe (CORROF)

La procédure CORROF permet de supprimer les corrections suivantes spécifiques à un axe :

- Décalages d'origine additifs réglés par manivelle (décalages DRF)
- Offsets de position programmés à l'aide de la variable système $AA_OFF

La suppression d'une valeur de correction arrête le prétraitement des blocs et la composante de position du déplacement forcé désélectionnée est reprise dans la position dans le système de coordonnées de base. Aucun axe n'est déplacé.

La valeur de position dans le système de coordonnées **machine**, disponible dans la variable système $AA_IM (consigne actuelle de l'axe dans le **SCM**), **reste inchangée**.

La valeur de position dans le système de coordonnées **pièce**, disponible dans la variable système $AA_IW (consigne actuelle de l'axe dans le **SCP**), **est changée**, car elle contient alors la composante du déplacement forcé désélectionné.

Remarque

CORROF peut être programmé dans un **programme CN**.

La programmation de CORROF dans une **action synchrone n'est pas** autorisée.

Syntaxe

```
CORROF(<Axis>,"<String>"[,<Axis>,"<String>"])
```

Signification

CORROF :	Procédure de désélection des décalages ou corrections suivantes d'un axe : • Décalage DRF • Offsets de position ($AA_OFF)	
	Prise d'effet :	modale
<Axis> :	Descripteur d'axe (descripteur d'axe de canal, d'axe géométrique ou d'axe machine)	
	Type de données :	AXIS
<String> :	Chaîne de caractères pour la définition du type de correction	
	Type de données :	BOOL
	Valeur	**Signification**
	DRF	Décalage DRF
	AA_OFF	Offset de position ($AA_OFF)

Exemples

Exemple 1 : Désélection spécifique à un axe d'un décalage DRF (1)

Un décalage DRF est généré avec la manivelle DRF dans l'axe X. Aucun décalage DRF n'est activé dans les autres axes du canal.

Code de programme	Commentaire
N10 CORROF(X,"DRF")	; CORROF agit ici comme DRFOF.
...	

Exemple 2 : Désélection spécifique à un axe d'un décalage DRF (2)

Un décalage DRF est généré avec la manivelle DRF dans l'axe X et dans l'axe Y. Aucun décalage DRF n'est activé dans les autres axes du canal.

Code de programme	Commentaire
; Seul le décalage DRF de l'axe X est désélectionné, le décalage DRF de l'axe Y étant conservé.	
; Avec DRFOF, les deux décalages auraient été désélectionnés.	
N10 CORROF(X,"DRF")	
...	

Exemple 3 : Désélection spécifique à un axe d'un offset de position $AA_OFF

Code de programme	Commentaire
; Un offset de position == 10 est interpolé pour l'axe X.	
N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5	
...	
; L'offset de position de l'axe X est désélectionné : $AA_OFF[X]=0	
; L'axe X ne se déplace pas.	
; L'offset de position est ajouté à la position actuelle de l'axe X.	
N80 CORROF(X,"AA_OFF")	
...	

Exemple 4 : Désélection spécifique à un axe d'un décalage DRF et d'un offset de position $AA_OFF (1)

Un décalage DRF est généré avec la manivelle DRF dans l'axe X. Aucun décalage DRF n'est activé dans les autres axes du canal.

Code de programme	Commentaire
; Un offset de position de 10 est interpolé pour l'axe X.	
N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5	
...	
; Seul le décalage DRF et l'offset de position de l'axe X sont désélectionnés.	
; Le décalage DRF de l'axe Y est conservé.	
N70 CORROF(X,"DRF",X,"AA_OFF")	
...	

Exemple 5 : Désélection spécifique à un axe d'un décalage DRF et d'un offset de position $AA_OFF (2)

Un décalage DRF est généré avec la manivelle DRF dans l'axe X et dans l'axe Y. Aucun décalage DRF n'est activé dans les autres axes du canal.

Code de programme	Commentaire
; Un offset de position == 10 est interpolé pour l'axe X.	
N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5	
...	

Code de programme	Commentaire
; Le décalage DRF de l'axe Y et l'offset de position de l'axe X sont désélectionnés.	
; Le décalage DRF de l'axe X est conservé.	
N70 CORROF(Y,"DRF",X,"AA_OFF")	
...	

Informations complémentaires

$AA_OFF_VAL

Après désélection de l'offset de position par $AA_OFF, la variable système $AA_OFF_VAL (trajet intégré de la superposition d'axe) de l'axe correspondant est égale à zéro.

$AA_OFF en mode JOG

Une interpolation de l'offset de position a également lieu comme déplacement superposé lors de la modification de $AA_OFF en mode JOG, si la validation de cette fonction a été réalisée avec le paramètre machine MD36750 $MA_AA_OFF_MODE.

$AA_OFF dans une action synchrone

Lors de la désélection de l'offset de position par `CORROF(<axe>,"AA_OFF")`, si une action synchrone est active activant immédiatement de nouveau $AA_OFF (`DO $AA_OFF[<axe>]=<valeur>`), alors $AA_OFF est désélectionné et n'est plus activé, puis l'alarme 21660 est émise. Si l'action synchrone est cependant activée ultérieurement, p. ex. dans le bloc suivant `CORROF`, $AA_OFF est mis à 1 et une interpolation d'offset de position est effectuée.

Remplacement automatique d'un axe de canal

Si un axe pour lequel `CORROF` a été programmé est actif dans un autre canal, il est intégré au canal par remplacement d'axe (condition : MD30552 $MA_AUTO_GET_TYPE > 0), puis l'offset de position et/ou le décalage DRF sont désélectionnés.

2.12.12 Désélection de décalages d'origine additifs (DRFOF)

Les décalages d'origine additifs, réglés par manivelle (décalages DRF), sont désélectionnés à l'aide de la procédure `DRFOF`.

La désélection déclenche un arrêt du prétraitement des blocs et la composante de position, correspondant au décalage DRF désélectionné, est reprise dans la position du système de coordonnées de base. Aucun axe n'est déplacé. La valeur de la variable système $AA_IM (consigne actuelle du SCM d'un axe) ne change pas, la valeur de la variable système $AA_IW (consigne actuelle du SCM d'un axe) change, car elle contient à présent la composante désélectionnée du déplacement forcé.

Syntaxe

```
DRFOF
```

Signification

DRFOF :	Procédure de désélection des décalages DRF pour tous les axes activés du canal	
	Prise d'effet :	modale

Exemples

Exemple 1 : Désélection spécifique à un axe d'un décalage DRF (1)

Un décalage DRF est généré avec la manivelle DRF dans l'axe X. Aucun décalage DRF n'est activé dans les autres axes du canal.

Code de programme	Commentaire
N10 CORROF(X,"DRF")	; CORROF agit ici comme DRFOF.
...	

Exemple 2 : Désélection spécifique à un axe d'un décalage DRF (2)

Un décalage DRF est généré avec la manivelle DRF dans l'axe X et dans l'axe Y. Aucun décalage DRF n'est activé dans les autres axes du canal.

Code de programme	Commentaire
N10 CORROF(X,"DRF")	; Seul le décalage DRF de l'axe X est désélectionné, le décalage DRF de l'axe Y étant conservé (avec DRFOF, les deux décalages auraient été désélectionnés).
...	

Exemple 3 : Désélection spécifique à un axe d'un décalage DRF et d'un offset de position $AA_OFF (1)

Un décalage DRF est généré avec la manivelle DRF dans l'axe X. Aucun décalage DRF n'est activé dans les autres axes du canal.

Code de programme	Commentaire
N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5	; Un offset de position == 10 est interpolé pour l'axe X.
...	
N70 CORROF(X,"DRF",X,"AA_OFF")	; Seul le décalage DRF et l'offset de position de l'axe X sont désélectionnés, le décalage DRF de l'axe Y étant conservé.
...	

Exemple 4 : Désélection spécifique à un axe d'un décalage DRF et d'un offset de position $AA_OFF (2)

Un décalage DRF est généré avec la manivelle DRF dans l'axe X et dans l'axe Y. Aucun décalage DRF n'est activé dans les autres axes du canal.

Code de programme	Commentaire
N10 WHEN TRUE DO $AA_OFF[X] = 10 G4 F5	; Un offset de position == 10 est interpolé pour l'axe X.
...	

Code de programme	Commentaire
N70 CORROF(Y,"DRF",X,"AA_OFF")	; Le décalage DRF de l'axe Y et l'offset de position de l'axe X sont désélectionnés, le décalage DRF de l'axe X étant conservé.
...	

2.12.13 Décalages d'origine spécifiques à la rectification (GFRAME0, GFRAME1 ... GFRAME100)

Instruction d'activation d'un frame de rectification dans le canal

La programmation de l'instruction GFRAME<n> active le frame de rectification correspondant de la gestion des données $P_GFR[<n>] dans le canal. A cet effet, le frame de rectification actif $P_GFRAME est réglé sur la même valeur que le frame de rectification de la gestion des données $P_GFR[<n>] :

GFRAME<n> ⇒ $P_GFRAME = $P_GFR[<n>]

Instruction	Frame de rectification activé dans le canal
GFRAME0	$P_GFR[0] (**frame neutre**)
GFRAME1	$P_GFR[1]
...	...
GFRAME100	$P_GFR[100]

Syntaxe

GFRAME<n>

Signification

GFRAME<n> :	Activation du frame de rectification <n> de la gestion des données	
	Groupe G :	64
	Position de base :	PM20150 $MC_GCODE_RESET_VALUES[**63**]
	Prise d'effet :	modale
<n> :	Numéro du frame de rectification	
	Plage de valeurs :	0, 1, 2, ... 100

2.13 Sorties de fonctions auxiliaires

Fonction

Par la sortie de fonctions auxiliaires, la CN communique en temps opportun à l'AP que le programme pièce désire faire exécuter certaines fonctions de commutation de la machine-outil par l'AP. Ceci a lieu par transfert des fonctions auxiliaires et de leurs paramètres à l'interface avec l'AP. Le traitement des valeurs et signaux transférés doit être réalisé par le programme AP utilisateur.

Fonctions auxiliaires

Les fonctions auxiliaires suivantes peuvent être transférées à l'AP :

Fonction auxiliaire	Adresse
Sélection d'outil	T
Correction d'outil	D, DL
Avance	F / FA
Vitesse de rotation de broche	S
Fonctions M	M
Fonctions H	H

Pour chaque groupe de fonctions ou fonction individuelle, les paramètres machine permettent de définir si la sortie est déclenchée **avant**, **en même temps que** ou **après** le déplacement.

L'AP acquitte les sorties de fonctions auxiliaires de différentes manières.

Propriétés

Le tableau suivant présente les principales propriétés des fonctions auxiliaires :

Fonction	Extension d'adresse		Valeur			Explications	Nombre maximal par bloc
	Signification	Zone	Zone	Type	Signification		
M	-	0 (implicite)	0 ... 99	INT	Fonction	Pour la plage de valeurs de 0 à 99, l'extension d'adresse est 0. Obligatoirement sans extension d'adresse : M0, M1, M2, M17, M30	5
	N° broche	1 - 12	1 ... 99	INT	Fonction	M3, M4, M5, M19, M70 avec extension d'adresse = n° broche (p.ex. `M2=5` ; arrêt broche pour broche 2). Sans n° de broche, la fonction s'applique à la broche maître.	
	Quelconque	0 - 99	100 ... 2147483647	INT	Fonction	Fonctions M de l'utilisateur*	

Fonction	Extension d'adresse		Valeur			Explications	Nombre maximal par bloc
	Signification	Zone	Zone	Type	Signification		
S	N° broche	1 - 12	0 ... ± 1,8*10^{308}	REAL	Vitesse de rotation	Sans n° de broche, la fonction s'applique à la broche maître.	3
H	Quelconque	0 - 99	0 ... ± 2147483647 ± 1,8*10^{308}	 INT REAL	Quelconque	Fonctions sans effet dans la CN, réalisation exclusive par l'AP*	3
T	N° broche (si GO active)	1 - 12	0 -32000 (également noms d'outil si la gestion d'outils est active)	INT	Appel d'outil	Les noms d'outil ne sont pas transférés à l'interface avec l'AP.	1
D	-	-	0 - 12	INT	Appel de correcteur d'outil	D0 : Désélection Préréglage : D1	1
DL	Correction en fonction de l'endroit	1 - 6	0 ... ± 1,8*10^{308}	REAL	Sélection de la correction d'outil fine	Se réfère au numéro D sélectionné précédemment.	1
F	-	-	0,001 - 999 999,999	REAL	Avance tangentielle		6
FA	N° axe	1 - 31	0,001 - 999 999,999	REAL	Avance paraxiale		
* La signification de ces fonctions est définie par le constructeur de machines (**voir les indications du constructeur de la machine**).							

Informations complémentaires

Nombre de sorties de fonction par bloc CN

Dans un bloc CN, il est possible de programmer 10 sorties de fonction au maximum. La génération de fonctions auxiliaires peut aussi être déclenchée dans la partie action d'**actions synchrones**.

Regroupement

Les fonctions citées peuvent être regroupées. Pour quelques fonctions M, la répartition en groupes est préréglée. Le regroupement permet de fixer le mode d'acquittement.

Sorties rapides de fonctions (QU)

Les fonctions, qui n'ont pas été configurées pour une sortie rapide, peuvent être définies à sortie rapide de façon ciblée avec le mot-clé `QU`. L'exécution du programme est poursuivie sans attente

de l'acquittement de la réalisation de la fonction supplémentaire (seulement attente de l'acquittement du transfert). Ceci permet d'éviter les interruptions inutiles des déplacements.

Remarque

Pour la fonction "Sorties rapides de fonctions", les paramètres machine correspondants doivent être activés (→ **constructeur de machines**).

Sorties de fonctions en liaison avec des déplacements

Le transfert d'informations ainsi que l'attente des réactions correspondantes sont coûteuses en temps et se répercutent donc aussi sur les déplacements.

Acquittement rapide sans répercussion sur le changement de bloc

Le mode de changement de bloc peut être influencé par la donnée machine. Avec le réglage "Changement de bloc non retardé", le comportement pour les fonctions auxiliaires rapides est le suivant :

Sortie de fonction auxiliaire	Comportement
avant le déplacement	La transition entre des blocs avec fonctions auxiliaires rapides a lieu **sans** interruption et **sans** réduction de vitesse. La sortie des fonctions auxiliaires a lieu dans le premier cycle d'interpolation du bloc. Le bloc suivant démarre sans retard dû à l'attente d'un acquittement.
pendant le déplacement	La transition entre des blocs avec fonctions auxiliaires rapides a lieu **sans** interruption et **sans** réduction de vitesse. La sortie des fonctions auxiliaires a lieu pendant le bloc. Le bloc suivant démarre sans retard dû à l'attente d'un acquittement.
après le déplacement	Le déplacement est interrompu à la fin du bloc. La sortie des fonctions auxiliaires a lieu à la fin du bloc. Le bloc suivant démarre sans retard dû à l'attente d'un acquittement.

⚠ PRUDENCE

Sorties de fonctions en contournage

Les sorties de fonctions à effectuer **avant** les déplacements interrompent le contournage (G64/G641) et génèrent un arrêt précis pour le bloc précédent.

Les sorties de fonctions à effectuer **après** les déplacements interrompent le contournage (G64/G641) et génèrent un arrêt précis pour le bloc courant.

Important : L'attente d'un signal d'acquittement manquant de l'AP peut également entraîner l'interruption du contournage, par exemple dans le cas des séquences d'instructions M dans des blocs contenant des trajets extrêmement courts.

2.13.1 Fonctions M

Les fonctions M permettent d'activer des opérations de commutation, par exemple la mise en marche/l'arrêt de l'arrosage, ainsi que d'autres fonctionnalités sur la machine.

Syntaxe

```
M<valeur>
M[<extension d'adresse>]=<valeur>
```

Signification

`M` :	Adresse pour la programmation de fonctions M	
`<extension d'adresse>` :	Pour certaines fonctions M, la notation à adresse étendue (par exemple avec indication du numéro de broche pour les fonctions de broche) s'applique à certaines fonctions M.	
`<valeur>` :	L'affectation d'une valeur (numéro de fonction M) permet une affectation à une fonction machine définie.	
	Type :	INT
	Plage de valeurs :	0 ... 2147483647 (valeur INT max.)

Fonctions M prédéfinies

Certaines fonctions M importantes pour l'exécution du programme sont déjà prédéfinies dans la commande :

Fonction M	Signification
M0*	Arrêt programmé
M1*	Arrêt facultatif
M2*	Fin du programme principal (comme M30)
M3	Rotation de la broche dans le sens horaire
M4	Rotation de la broche dans le sens antihoraire
M5	Arrêt de la broche
M6	Changement d'outil (réglage par défaut)
M17*	Fin du sous-programme
M19	Positionnement de la broche
M30*	Fin du programme principal (comme M2)
M40	Changement automatique de rapport de boîte de vitesses
M41	Rapport de boîte de vitesses 1
M42	Rapport de boîte de vitesses 2
M43	Rapport de boîte de vitesses 3
M44	Rapport de boîte de vitesses 4
M45	Rapport de boîte de vitesses 5
M70	Commutation de la broche dans le mode axe

Remarque

L'extension de l'adresse n'est pas admise pour les fonctions repérées par *.

Les fonctions M0, M1, M2, M17 et M30 sont toujours activées **après** le déplacement.

Fonctions M définies par le constructeur de machine

Tous les numéros de fonction M libres peuvent être affectés par le constructeur de machine par exemple à des fonctions de commutation destinées à la commande de dispositifs de serrage ou à l'activation/désactivation d'autres fonctions machines.

Remarque

Les fonctionnalités affectées aux numéros de fonction M libres sont spécifiques à la machine. La fonctionnalité d'une certaine fonction M peut donc varier d'une machine à l'autre.

Les fonctions M disponibles sur une machine, ainsi que les fonctionnalités correspondantes, sont indiquées par le constructeur de machine.

Exemples

Exemple 1 : nombre maximal de fonctions M dans le bloc

```
Code de programme                Commentaire
N10 S...
N20 X... M3                      ; Fonction M dans le bloc avec déplacement d'axe,
                                 ; démarrage de la broche avant le déplacement de l'axe X.
N180 M789 M1767 M100 M102 M376   ; 5 fonctions M au maximum dans un bloc.
```

Exemple 2 : fonction M en tant que sortie rapide

```
Code de programme                Commentaire
N10 H=QU(735)                    ; Sortie rapide pour H735.
N10 G1 F300 X10 Y20 G64
N20 X8 Y90 M=QU(7)               ; Sortie rapide pour M7.
```

M7 a été programmée en tant que sortie rapide de telle sorte que le contournage (G64) n'est pas interrompu.

Remarque

N'utilisez cette possibilité que dans des cas particuliers, car le déroulement temporel en sera modifié, par exemple, en cas de combinaison avec d'autres sorties de fonctions.

Informations complémentaires pour les fonctions M prédéfinies

Arrêt programmé : M0

L'usinage est arrêté dans le bloc CN contenant `M0`. Vous avez maintenant la possibilité d'enlever les copeaux, d'effectuer des mesures etc.

Arrêt programmé 1 - Arrêt facultatif : M1

M1 peut être réglé par :

- HMI / boîte de dialogue "Influence sur le programme"
  ou
- Interface CN/AP

L'exécution du programme CN est suspendue par les blocs programmés.

Arrêt programmé 2 - Une fonction auxiliaire associée à M1 avec arrêt pendant l'exécution du programme

L'arrêt programmé 2 est configurable par l'intermédiaire de HMI / boîte de dialogue "Influence sur le programme" et permet une interruption des procédures technologiques à la fin de la pièce à usiner. L'utilisateur peut ainsi intervenir dans la production en cours, par exemple pour éliminer des copeaux.

Fin de programme : M2, M17, M30

`M2`, `M17` ou `M30` permettent de terminer l'exécution d'un programme. Si le programme principal est appelé à partir d'un autre programme (comme un sous-programme), `M2`/`M30` agissent comme `M17` et inversement, autrement dit `M17` agit dans le programme principal comme `M2`/`M30`.

Fonctions de broche : M3, M4, M5, M19, M70

La notation à adresse étendue avec indication du numéro de broche s'applique à toutes les fonctions de broche.

Exemple :

Code de programme	**Commentaire**
`M2=3`	`; Rotation de la seconde broche dans le sens horaire`

Si aucune extension d'adresse n'est programmée, la fonction s'applique à la broche maître.

2.14 Instructions complémentaires

2.14.1 Transmission de messages (MSG)

L'instruction `MSG()` permet de transmettre, depuis le programme pièce, une chaîne de caractères quelconque sous forme de message destiné à l'opérateur.

Syntaxe

```
MSG("<Texte du message>"[,<Exécution>])
...
MSG( )
```

Signification

`MSG` :	Appel de sous-programme prédéfini pour la génération d'un message		
`<Texte du message>` :	Chaîne de caractères quelconque destinée à être affichée en tant que message		
	Type :	STRING	
	Longueur maximale :	124 caractères ; affichage sur deux lignes (2*62 caractères)	
	Il est également possible de transmettre des variables dans le texte du message en utilisant l'opérateur de concaténation "<<".		
`<Exécution>` :	Paramètre permettant de définir l'instant auquel l'écriture du message doit être exécutée (facultatif)		
	Type :	INT	
	Valeur :	0 (réglage par défaut)	Aucun bloc principal spécifique n'est créé pour l'écriture du message. Elle s'effectue dans le bloc CN exécutable suivant. Il ne se produit aucune interruption du contournage actif.
		1	Un bloc principal spécifique est créé pour l'écriture du message. Tout contournage actif est interrompu.
`MSG()` :	La programmation de `MSG()` sans texte de message permet de supprimer à nouveau le message actuel. Sans suppression, le message reste affiché jusqu'à l'arrivée du message suivant.		

Remarque

Si un message doit s'afficher dans la langue activée pour l'interface utilisateur, l'utilisateur doit disposer d'informations sur la langue actuellement paramétrée sur l'IHM. Cette information peut être interrogée dans le programme pièce et dans les actions synchrones, au moyen de la variable système $AN_LANGUAGE_ON_HMI (Page 1396).

Exemples

Exemple 1 : transmettre / effacer un message

Code de programme	Commentaire
N10 G91 G64 F100	; contournage
N20 X1 Y1	
N... X... Y...	
N20 MSG ("Usinage pièce 1")	; Le message n'est transmis qu'avec N30.
	; Le contournage reste inchangé.
N30 X... Y...	
N... X... Y...	
N400 X1 Y1	
N410 MSG ("Usinage pièce 2",1)	; Le message est transmis avec N410.
	; Le contournage est interrompu.
N420 X1 Y1	
N... X... Y...	
N900 MSG ()	; Suppression du message.

Exemple 2 : texte de message avec variable

Code de programme	Commentaire
N10 R12=$AA_IW [X]	; Position actuelle de l'axe X dans R12.
N20 MSG("Position de l'axe X"<<R12<<"vérifier")	; Transmettre un message avec la variable R12.
...	
N90 MSG ()	; Effacer le message dans N20.

2.14.2 Écrire la chaîne de caractères dans la variable OPI (WRTPR)

Avec la fonction `WRTPR()`, vous pouvez écrire une chaîne quelconque de caractères dans la variable OPI progProtText à partir du programme pièce.

Syntaxe

```
WRTPR(<String>[,<ExecTime>])
```

Signification

`WRTPR` :	Appel de fonction de sortie d'une chaîne de caractères.		
`<String>` :	Chaîne quelconque de caractères pouvant être écrite dans la variable OPI progProtText.		
	Type :	STRING	
	Longueur maximale :	128 caractères	
`<ExecTime>` :	Paramètre facultatif permettant de définir l'instant auquel l'écriture de la chaîne de caractères doit être exécutée.		
	Type :	INT	
	Plage de valeurs :	0, 1	
		0 (par défaut)	Aucun bloc principal spécifique n'est créé pour l'écriture de la chaîne de caractères. Elle s'effectue dans le bloc CN exécutable suivant. Il ne se produit aucune interruption du contournage actif.
		1	Un bloc principal spécifique est créé pour l'écriture de la chaîne de caractères. Tout contournage actif est interrompu.

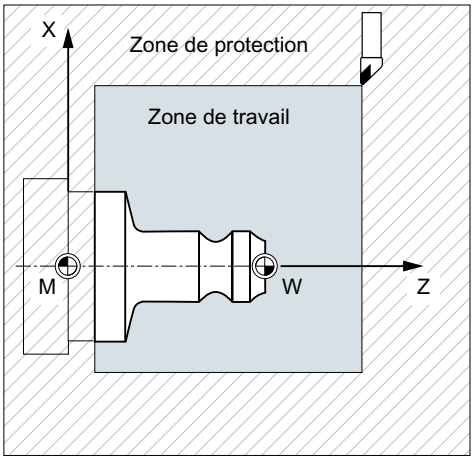
Exemples

```
Code de programme                Commentaire
N10 G91 G64 F100                 ; Contournage
N20 X1 Y1
N30 WRTPR("N30")                 ; La chaîne de caractères "N30" est tout d'abord écrite
                                 dans N40.
                                 ; Le contournage reste inchangé.
N40 X1 Y1
N50 WRTPR("N50",1)               ; La chaîne de caractères "N50" est tout d'abord écrite
                                 dans N50.
                                 ; Le contournage est interrompu.
N60 X1 Y1
```

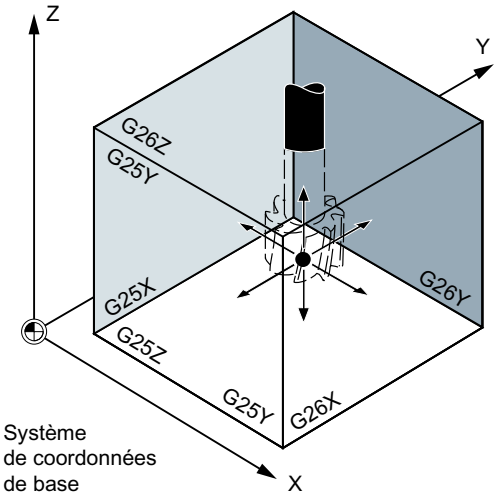
2.14.3 Limitation de la zone de travail

2.14.3.1 Limitation de la zone de travail dans le SCB (G25/G26, WALIMON, WALIMOF)

Avec G25/G26, vous pouvez limiter la zone (champ, volume) de travail dans laquelle l'outil pourra se déplacer et ce, dans chaque axe de canal. Les zones se trouvant à l'extérieur des limites de zone de travail définies par G25/G26 sont interdites pour les mouvements d'outil.



Les coordonnées pour les différents axes sont valables dans le système de coordonnées de base :



La limitation de la zone de travail pour tous les axes validés doit être programmée avec l'instruction WALMON. Une programmation de la limitation de la zone de travail avec WALIMOF est sans effet. L'instruction WALIMON est préréglée et doit donc être uniquement programmée si la limitation de la zone de travail a été auparavant désactivée.

Syntaxe

```
G25 X…Y…Z…
G26 X…Y…Z…
WALIMON
...
WALIMOF
```

Signification

G25 :	limitation **inférieure** de la zone de travail Affectation des valeurs aux axes de canal dans le système de coordonnées de base
G26 :	Limitation **supérieure** de la zone de travail Affectation des valeurs aux axes de canal dans le système de coordonnées de base
X… Y… Z… :	Limitations inférieures ou supérieures de la zone de travail des axes de canal individuels Les indications se rapportent au système de coordonnées de base.
WALIMON :	**Activer** la limitation de la zone de travail pour tous les axes
WALIMOF :	**Désactiver** la limitation de la zone de travail pour tous les axes

A part l'introduction programmable des valeurs via G25/G26, vous pouvez aussi introduire manuellement des valeurs à l'aide des données de réglage spécifiques à l'axe :

PS43420 $SA_WORKAREA_LIMIT_PLUS (limitation positive de la zone de travail)

PS43430 $SA_WORKAREA_LIMIT_MINUS (limitation négative de la zone de travail)

L'activation et la désactivation des limitations de la zone de travail, paramétrées via SD43420 et SD43430, sont effectuées dans une direction spécifique via les données de réglage immédiatement effectives et spécifiques à l'axe :

PS43400 $SA_WORKAREA_PLUS_ENABLE (limitation de la zone de travail active dans le sens positif)

PS43410 $SA_WORKAREA_MINUS_ENABLE (limitation de la zone de travail active dans le sens négatif)

L'activation/désactivation spécifique à une direction permet de limiter la zone de travail, pour un axe, dans une seule direction.

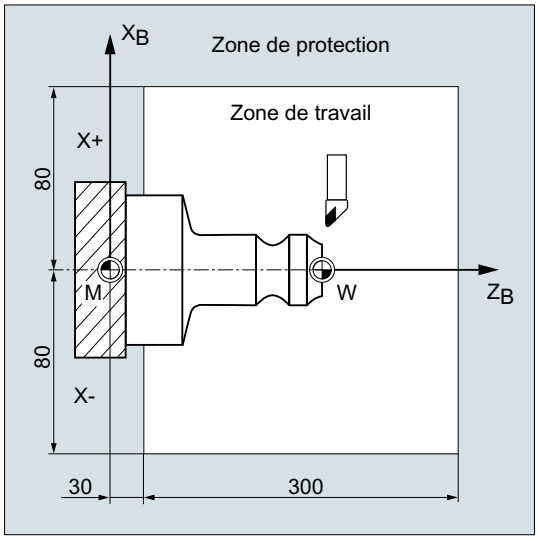
Remarque

La limitation de la zone de travail programmée via G25/G26 est prioritaire et écrase les valeurs introduites par SD43420 SD43430.

Remarque

Avec G25/G26, vous pouvez aussi programmer sous l'adresse S des valeurs limites pour la vitesse de rotation de broche. Pour plus d'informations à ce sujet, référez-vous à " Limitation programmable de la vitesse de rotation de broche (G25, G26) (Page 106) ".

Exemple



La limitation de la zone de travail via G25/26 permet de restreindre le volume de travail d'une machine rotative de manière à protéger les installations limitrophes telles que revolvers, stations de mesure, etc. contre tout endommagement.

Préréglage : WALIMON

Code de programme	Commentaire
N10 G0 G90 F0.5 T1	
N20 G25 X-80 Z30	; Définition de la limite inférieure pour les différents axes de coordonnées
N30 G26 X80 Z330	; Définition de la limite supérieure
N40 L22	; Programme d'usinage
N50 G0 G90 Z102 T2	; Aller au point de changement d'outil
N60 X0	
N70 WALIMOF	; Désactiver la limitation de la zone de travail
N80 G1 Z-2 F0.5	; Perçage des avant-trous
N90 G0 Z200	; Retour en arrière
N100 WALIMON	; Activer la limitation de la zone de travail
N110 X70 M30	; Fin du programme

Plus d'informations

Point de référence de l'outil

La pointe de l'outil est surveillée comme point de référence quand la correction de longueur d'outil est active, sinon c'est le point de référence de l'organe porte-outil.

La prise en compte du rayon d'outil doit être activée séparément. Cette activation intervient via le paramètre machine spécifique au canal :

PM21020 $MC_WORKAREA_WITH_TOOL_RADIUS

Si le point de référence de l'outil se trouve à l'extérieur du volume de travail défini par la limitation de la zone de travail ou qu'il quitte ce volume, l'exécution du programme est arrêtée.

Remarque

Lorsque des transformations sont actives, la prise en compte des données d'outil (longueur et rayon d'outil) peut différer du comportement décrit.

Limitation programmable de la zone de travail, G25/G26

Pour chaque axe, on peut définir une limite supérieure (G26) et une limite inférieure (G25) de la zone de travail. Ces valeurs sont immédiatement effectives et sont conservées, en cas de réglage adéquat du MD (→ MD10710 $MN_PROG_SD_RESET_SAVE_TAB), après remise à zéro et remise sous tension.

Remarque

La fonction prédéfinie CALCPOSI (Page 595) permet de vérifier, avant toute activation des déplacements de la machine, si la trajectoire prévue peut être exécutée à l'intérieur des limites de la zone de travail et/ou des zones de protection.

2.14.3.2 Limitation de la zone de travail dans le SCP/SCR (WALCS0 ... WALCS10)

La "Limitation de la zone de travail dans le SCP/SCR" permet une limitation flexible orientée pièce de la plage de déplacement des axes de canal dans le système de coordonnées pièce (SCP) ou le système de coordonnées réglable (SCR). Elle a été essentiellement conçue pour le domaine des tours traditionnels.

Condition

Les axes de canal doivent être référencés.

Groupe de limitations de la zone de travail

Avec les groupes de limitations de la zone de travail, il n'est pas nécessaire de réécrire les limites de la zone de travail propres à chaque axe pour tous les axes de canal en cas de commutation des affectations d'axe, par exemple en cas d'activation/désactivation des transformations ou du frame actif.

Un groupe de limitations de la zone de travail comprend les données suivantes :

- Limites de la zone de travail pour tous les axes de canal
- Système de référence de la limitation de la zone de travail

Syntaxe

```
...
$P_WORKAREA_CS_COORD_SYSTEM[<WALimNo>]=<Value>
$P_WORKAREA_CS_PLUS_ENABLE[<WALimNo>,<Ax>]=<Value>
```

```
$P_WORKAREA_CS_LIMIT_PLUS[<WALimNo>,<Ax>]=<Value>
$P_WORKAREA_CS_MINUS_ENABLE[<WALimNo>,<Ax>]=<Value>
$P_WORKAREA_CS_LIMIT_MINUS[<WALimNo>,<Ax>]=<Value>
...
WALCS<n>
...
WALCS0
```

Signification

$P_WORKAREA_CS_COORD_SYSTEM[<WALimNo>]=<Value>		
Système de coordonnées auquel se rapporte la limitation de la zone de travail d'un groupe		
<WALimNo> :	Groupe de limitations de la zone de travail	
	Type :	INT
	Plage de valeurs :	0 (groupe 1) ... 9 (groupe 10)
<Value> :	Valeur du type INT	
	1	Système de coordonnées pièce (SCP)
	3	Système de coordonnées réglable (SCR)

$$P_WORKAREA_CS_PLUS_ENABLE[<WALimNo>,<Ax>]=<Value>		
Libérer la limitation de la zone de travail dans le sens **positif** de l'axe pour l'axe de canal indiqué		
<WALimNo> :	Groupe de limitations de la zone de travail	
	Type :	INT
	Plage de valeurs :	0 (groupe 1) ... 9 (groupe 10)
<Axe> :	Nom d'axe de canal	
<Value> :	Valeur du type BOOL	
	0 (FALSE)	Aucun déblocage
	1 (TRUE)	Déblocage

$P_WORKAREA_CS_MINUS_ENABLE[<WALimNo>,<Ax>]=<Value>		
Libérer la limitation de la zone de travail dans le sens **négatif** de l'axe pour l'axe de canal indiqué		
<WALimNo> :	Groupe de limitations de la zone de travail	
	Type :	INT
	Plage de valeurs :	0 (groupe 1) ... 9 (groupe 10)
<Axe> :	Nom d'axe de canal	
<Value> :	Valeur du type BOOL	
	0 (FALSE)	Aucun déblocage
	1 (TRUE)	Déblocage

$P_WORKAREA_CS_LIMIT_PLUS[<WALimNo>,<Ax>]=<Value>
Limitation de la zone de travail dans le sens **positif** de l'axe de canal indiqué

<WALimNo> :	Groupe de limitations de la zone de travail	
	Type :	INT
	Plage de valeurs :	0 (groupe 1) ... 9 (groupe 10)
<Axe> :	Nom d'axe de canal	
<Value> :	Valeur du type REAL	

$P_WORKAREA_CS_LIMIT_MINUS[<WALimNo>,<Ax>]=<Value>		
Limitation de la zone de travail dans le sens **négatif** de l'axe de canal indiqué		
<WALimNo> :	Groupe de limitations de la zone de travail	
	Type :	INT
	Plage de valeurs :	0 (groupe 1) ... 9 (groupe 10)
<Axe> :	Nom d'axe de canal	
<Value> :	Valeur du type REAL	

WALCS<n> :	Activation des limitations de la zone de travail d'un groupe de limitations de la zone de travail	
<n> :	Numéro du groupe de limitations de la zone de travail	
	Plage de valeurs :	1 ... 10

WALCS0 :	désactivation des limitations de la zone de travail actives dans le canal

Remarque

Le nombre effectivement disponible de groupes de limitations de la zone de travail dépend de la configuration (→ voir les indications du constructeur de machines).

Exemple

3 axes sont définis dans le canal : X, Y et Z.

Un groupe de limitations de la zone de travail portant le n° 2 doit être défini, puis activé, les axes dans le groupe étant limités dans le SCP selon les spécifications suivantes :

- Axe X en sens Plus : 10 mm
- Axe X en sens Moins : aucune limitation
- Axe Y en sens Plus : 34 mm
- Axe Y en sens Moins : -25 mm
- Axe Z en sens Plus : aucune limitation
- Axe Z en sens Moins : -600 mm

Code de programme	Commentaire
...	
N51 $P_WORKAREA_CS_COORD_SYSTEM[1]=1	; La limitation de la zone de travail du groupe de limitations de la zone de travail 2 est valable dans le SCP.
N60 $P_WORKAREA_CS_PLUS_ENABLE[1,X]=TRUE	
N61 $P_WORKAREA_CS_LIMIT_PLUS[1,X]=10	
N62 $P_WORKAREA_CS_MINUS_ENABLE[1,X]=FALSE	
N70 $P_WORKAREA_CS_PLUS_ENABLE[1,Y]=TRUE	
N73 $P_WORKAREA_CS_LIMIT_PLUS[1,Y]=34	
N72 $P_WORKAREA_CS_MINUS_ENABLE[1,Y]=TRUE	
N73 $P_WORKAREA_CS_LIMIT_MINUS[1,Y]=-25	
N80 $P_WORKAREA_CS_PLUS_ENABLE[1,Z]=FALSE	
N82 $P_WORKAREA_CS_MINUS_ENABLE[1,Z]=TRUE	
N83 $P_WORKAREA_CS_LIMIT_PLUS[1,Z]=-600	
...	
N90 WALCS2	; Activation du groupe de limitations de la zone de travail 2.
...	

Informations complémentaires

Prise d'effet

La limitation de la zone de travail via `WALCS1` - `WALCS10` est effective indépendamment de la limitation de la zone de travail via `WALIMON`. Si les deux fonctions sont actives, c'est la limitation rencontrée en premier par la déplacement de l'axe qui sera effective.

Point de référence de l'outil

La prise en compte des données d'outil (longueur et rayon de l'outil) et, par conséquent, du point de référence de l'outil lors de la surveillance des limitations de la zone de travail, correspond à la limitation de la zone de travail avec `WALIMON`.

2.14.4 Accostage du point de référence (G74)

Après la mise en marche de la machine, tous les chariots de déplacement axial doivent accoster leur point de référence (lorsque des systèmes de mesure relatifs sont utilisés). Ce n'est qu'après cette opération que des déplacements peuvent être programmés.

Le code `G74` permet d'exécuter l'accostage du point de référence dans le programme CN.

Syntaxe

```
G74 X1=0 Y1=0 Z1=0 A1=0 ... ; Programmation dans un bloc CN spécifique
```

Signification

`G74` :	Appel d'instruction G Accostage du point de référence
`X1=0 Y1=0 Z1=0 …` :	L'adresse d'axe machine `X1`, `Y1`, `Z1`... spécifiée pour les **axes linéaires** est accostée au point de référence
`A1=0 B1=0 C1=0 …` :	L'adresse d'axe machine `A1`, `B1`, `C1`... spécifiée pour les **axes rotatifs** est accostée au point de référence

Remarque

Avant l'accostage du point de référence, aucune transformation ne doit être programmée pour un axe qui doit accoster le point de référence avec `G74`.

La transformation est désactivée avec l'instruction `TRAFOOF`.

Exemple

Dès que le système de mesure change, le point de référence est accosté et l'origine pièce déterminée.

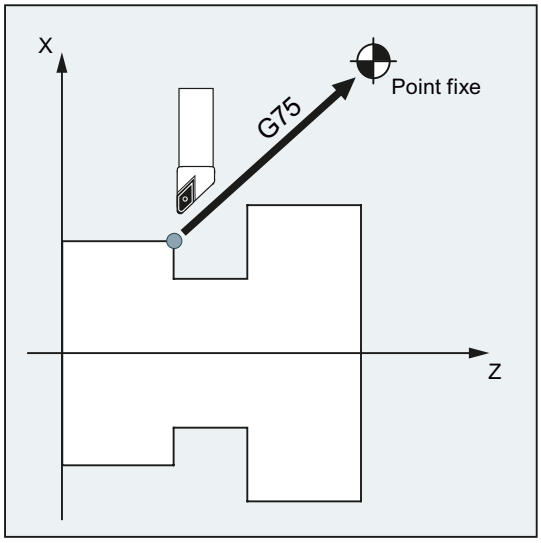
Code de programme	**Commentaire**
`N10 SPOS=0`	`; broche en asservissement de position`
`N20 G74 X1=0 Y1=0 Z1=0 C1=0`	`; Accostage du point de référence pour les axes linéaires et les axes rotatifs`
`N30 G54`	`; Décalage d'origine`
`N40 L47`	`; Programme d'usinage`
`N50 M30`	`; Fin du programme`

2.14.5 Accostage d'un point fixe (G75)

L'instruction `G75` à effet non modal permet à des axes individuels et indépendants les uns des autres d'accoster des points fixes dans la zone de la machine, p. ex. des points de changement d'outil, points de chargement, points de changement de palette, etc.

Les points fixes correspondent à des positions dans le système de coordonnées machine, qui sont enregistrées dans les paramètres machine (PM30600 $MA_FIX_POINT_POS[n]). 4 points fixes peuvent être définis au maximum par axe.

Les points fixes peuvent être accostés dans n'importe quel programme CN, indépendamment des positions d'outil et de pièce courantes. Un arrêt interne du prétraitement des blocs est réalisé avant le déplacement des axes.



Conditions

Pour l'accostage de points fixes avec G75, les conditions suivantes doivent être remplies :

- Les coordonnées du point fixe doivent être déterminées de façon précise et enregistrées dans les paramètres machine.
- Les points fixes doivent se situer dans la plage de déplacement valide (→ Tenir compte des limites logicielles des fins de course !)
- La prise de référence des axes à déplacer doit avoir été effectuée.
- Aucune correction du rayon d'outil ne doit être active.
- Aucune transformation cinématique ne doit être active.
- Les axes à déplacer ne doivent prendre part à aucune transformation active.
- Aucun des axes à déplacer ne doit être un axe asservi d'un couplage actif.
- Aucun des axes à déplacer ne doit être un axe d'une association Gantry.
- Les cycles de compilation ne doivent concerner aucune composante de déplacement.

Syntaxe

```
G75 <nom d'axe><position de l'axe> ... FP=<n>
```

Signification

G75 :	Accostage d'un point fixe
<nom d'axe> :	Nom de l'axe machine à déplacer vers le point fixe Tous les descripteurs d'axe sont autorisés.
<position de l'axe> :	La valeur de position n'a aucune signification. De ce fait, on indique en général la valeur "0".

FP= :	Point fixe à accoster		
	<n> :	Numéro du point fixe	
		Plage de valeurs :	1, 2, 3, 4
	Remarque : Si aucun FP=<n> ou aucun numéro de point fixe ne sont programmés ou si FP=0, cela est interprété comme FP=1 et le point fixe 1 est accosté.		

Remarque

Il est également possible de programmer plusieurs axes dans un bloc G75. Ces axes sont alors déplacés simultanément sur le point fixe indiqué.

Remarque

La valeur de l'adresse FP ne doit pas être supérieure au nombre de points fixes définis pour chaque axe programmé (PM30610 $MA_NUM_FIX_POINT_POS).

Exemple

Pour un changement d'outil, il s'agit de déplacer les axes X (= AX1) et Z (= AX3) à la position fixe de l'axe machine 1 avec X = 151,6 et Z = -17,3.

Paramètres machine :

- PM30600 $MA_FIX_POINT_POS[AX1,0] = 151.6
- PM30600 $MA_FIX_POINT[AX3,0] = 17.3

Programme CN :

```
Code de programme          Commentaire
…
N100 G55                   ; Activer le décalage d'origine réglable.
N110 X10 Y30 Z40           ; Accoster les positions dans le SCP.
N120 G75 X0 Z0 FP=1 M0     ; L'axe X se déplace sur 151,6
                           ; et l'axe Z se déplace sur 17,3 (dans le SCM).
                           ; Chaque axe se déplace à sa vitesse maximale.
                           ; Aucun déplacement supplémentaire ne doit être actif
                           dans ce bloc.
                           ; Afin qu'un déplacement supplémentaire ne puisse pas
                           ; non plus être réalisé une fois les positions finales
                           atteintes,
                           ; un arrêt est inséré à cet endroit.
N130 X10 Y30 Z40           ; La position de N110 est de nouveau accostée.
                           ; Le décalage d'origine est de nouveau actif.
…
```

Remarque

Lorsque la fonction "Gestion d'outils avec des magasins" est active, les fonctions auxiliaires `T...` ou `M...` (typiquement `M6`) ne suffisent pas pour déclencher le verrouillage du changement de bloc à la fin du déplacement `G75`.

Raison : Avec le paramétrage "Gestion d'outils avec des magasins active", les fonctions auxiliaires pour le changement d'outil ne sont pas transmises à l'AP.

Informations complémentaires

G75

Les axes sont déplacés comme axes machine en rapide. Le déplacement est représenté en interne par les fonctions "SUPA" (inhibition de tous les frames) et "G0 RTLIOF" (déplacement en rapide avec interpolation axiale individuelle).

Si les conditions pour "RTLIOF" (interpolation axiale individuelle) ne sont pas remplies, le point est accosté en tant que trajectoire.

Lorsqu'ils atteignent le point fixe, les axes s'arrêtent dans la fenêtre de tolérance "Arrêt précis fin".

Dynamique paramétrable pour G75

Pour les déplacements de positionnement aux positions de points fixes (G75), le paramètre machine suivant permet de régler le mode dynamique souhaité :

MD18960 $MN_POS_DYN_MODE (type de dynamique de l'axe de positionnement)

Déplacements axiaux supplémentaires

Les déplacements axiaux supplémentaires suivants sont pris en compte au moment de l'interprétation du bloc G75 :

- Décalage d'origine externe
- DRF
- Offset de synchronisation ($AA_OFF)

Ensuite, les déplacements axiaux supplémentaires ne doivent plus être modifiés jusqu'à ce que la fin du déplacement soit atteinte par le bloc G75.

Des déplacements supplémentaires après l'interprétation du bloc G75 entraînent un décalage correspondant du point fixe accosté.

Les déplacements supplémentaires suivants ne sont pas pris en compte quel que soit le moment d'interprétation et entraînent un décalage correspondant de la position de destination :

- Correction d'outil en ligne
- Déplacements supplémentaires provenant de cycles de compilation dans le SCB comme le SCM

Frames actifs

Tous les frames actifs sont ignorés Les déplacements s'effectuent dans le système de coordonnées machine.

Limitation de la zone de travail dans le SCP/SRC

La limitation de la zone de travail spécifique au système de coordonnées (WALCS0 ... WALCS10) n'a aucun effet dans le bloc avec G75. La destination est surveillée comme point de départ du bloc suivant.

Déplacements d'axes/de broches avec POSA/SPOSA

Si des axes/broches programmés ont précédemment été déplacés avec POSA ou SPOSA, ces déplacements sont d'abord menés à terme avant l'accostage du point fixe.

Fonctions de broche dans le bloc G75

Si la broche est exclue de la fonction "Accostage d'un point fixe", il est possible de programmer des fonctions de broche supplémentaires (par exemple un positionnement avec SPOS/SPOSA) dans le bloc G75.

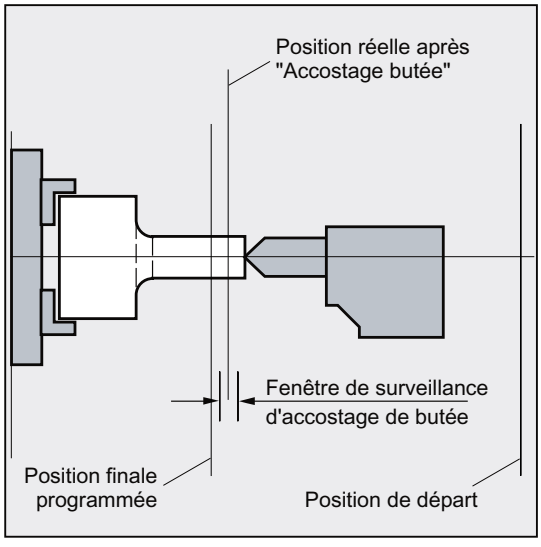
Axes modulo

Avec les axes modulo, le point fixe est accosté par le trajet le plus court.

2.14.6 Accostage d'une butée (FXS, FXST, FXSW)

Fonction

Grâce à la fonction "Accostage d'une butée", il est possible de générer des forces définies pour le blocage de pièces, telles que celles requises pour les poupées mobiles, les fourreaux de poupée mobile et les préhenseurs par exemple. De plus, cette fonction permet d'accoster des points de référence mécaniques.



En cas de couple suffisamment réduit, des mesures simples sont aussi possibles sans qu'un palpeur doive être raccordé. La fonction "Accostage d'une butée" peut être utilisée pour les axes et pour les broches susceptibles d'être traitées comme des axes.

Syntaxe

```
FXS[<axe>]=...
FXST[<axe>]=...
FXSW[<axe>]=...
FXS[<axe>]=... FXST[<axe>]=...
FXS[<axe>]=... FXST[<axe>]=... FXSW[<axe>]=...
```

Signification

`FXS` :	Instruction d'activation et de désactivation de la fonction "Accostage d'une butée"
	`FXS[<axe>]=1` : Activation de la fonction
	`FXS=[<axe>]=0` : Désactivation de la fonction
`FXST` :	Instruction optionnelle pour le réglage du couple de blocage
	Indication en % du couple maximal de l'entraînement
`FXSW` :	Instruction optionnelle pour le réglage de la largeur de la fenêtre de surveillance d'immobilisation en butée
	Indication en mm, inch ou degrés
`<axe>` :	Noms d'axe machine
	On programme des axes machine (X1, Y1, Z1 etc.).

Remarque

Les instructions `FXS`, `FXST` et `FXSW` sont modales.

La programmation de `FXST` et `FXSW` est facultative : s'il n'y a pas de donnée introduite, la commande numérique utilise la dernière valeur programmée ou la valeur figurant dans le paramètre machine correspondant.

Activer l'accostage de la butée FXS[<axe>] = 1

Le déplacement vers le point de destination peut être décrit comme étant un déplacement d'axe à interpolation ou d'axe de positionnement. Dans le cas d'un axe de positionnement, la fonction peut être active au-delà des limites du bloc dans lequel elle a été programmée.

L'accostage d'une butée peut aussi avoir lieu dans plusieurs axes simultanément et en parallèle au déplacement d'autres axes. La butée doit se trouver entre la position de départ et la position de destination.

IMPORTANT

Risque de collision

Dès que la fonction "Accostage d'une butée" a été activée pour un axe/une broche, vous ne devez programmer aucune nouvelle position pour cet axe.

Avant d'activer la fonction, il faut commuter les broches en mode asservissement de position.

Exemple :

Code de programme	**Commentaire**
`X250 Y100 F100 FXS[X1]=1 FXST[X1]=12.3 FXSW[X1]=2`	`; L'axe X1 est déplacé vers la position de destination X= 250 mm avec l'avance F100 (indication facultative).` `Le couple de blocage est égal à 12,3 % du couple maximum d'entraînement ; la surveillance a lieu dans une fenêtre de 2 mm de large.`
`...`	

Désactivation de l'accostage de butée FXS[<axe>] = 0

Le fait de désactiver la fonction déclenche un arrêt du prétraitement des blocs.

Le bloc avec `FXS[<axe>]=0` doit contenir des déplacements.

IMPORTANT
Risque de collision Le déplacement vers la position de retrait doit éloigner l'outil de la butée, pour éviter un endommagement de la butée ou de la machine. Le changement de bloc a lieu dès que la position de retrait est atteinte. En l'absence de position de retrait, le changement de bloc se fait dès la coupure de la limitation de couple

Exemple :

Code de programme	**Commentaire**
`X200 Y400 G01 G94 F2000 FXS[X1] = 0`	`; L'axe X1 opère un retrait de la butée pour atteindre la position X=200 mm. Toutes les autres introductions sont optionnelles.`
`...`	

Couple de blocage (FXST) et fenêtre de surveillance (FXSW)

Si vous avez programmé une limitation du couple FXST, celle-ci est active dès le début du bloc, autrement dit l'accostage de la butée se fera aussi avec un couple réduit. FXST et FXSW sont programmables et modifiables à n'importe quel moment dans le programme pièce : Les modifications seront activées avant l'exécution des déplacements programmés dans le même bloc.

IMPORTANT
Risque de collision Si une nouvelle fenêtre de surveillance de butée est programmée, alors non seulement la largeur de la fenêtre se trouvera modifiée, mais aussi le point de référence pour le centre de la fenêtre, si l'axe a été déplacé auparavant. La position réelle de l'axe machine après modification de la fenêtre est le nouveau centre de la fenêtre. La fenêtre est à choisir de sorte que seule une rupture de la butée puisse activer la surveillance d'immobilisation.

Informations complémentaires

Rampe de montée

Avec un paramètre machine, vous pouvez définir une rampe de montée pour la nouvelle limite de couple afin d'éviter une modification brutale (dans le cas d'une pression sur le fourreau de contre-poupée par exemple).

Suppression des alarmes

En pratique, vous pouvez inhiber l'alarme de butée du programme pièce en la masquant dans un paramètre machine et en activant le réglage du PM avec NEW_CONF.

Activation

Les instructions d'accostage de la butée peuvent être activées dans des actions synchrones / des cycles technologiques. L'activation peut se faire sans déplacement, la limitation du couple agit immédiatement. Dès que l'axe est déplacé suivant les consignes, la surveillance de la butée est active.

Activation à partir d'actions synchrones

Exemple :

Quand l'événement attendu ($R1) se produit alors que l'accostage de la butée n'est pas encore amorcé, FXS doit être activé pour l'axe Y. Le couple doit être égal à 10 % du couple nominal. La largeur de surveillance d'immobilisation en butée est égale à la valeur préréglée.

Code de programme

```
N10 IDS=1 WHENEVER (($R1=1) AND ($AA_FXS[Y]==0)) DO $R1=0 FXS[Y]=1
FXST[Y]=10
```

Le programme pièce doit contenir les instructions pour que $R1 soit mis à 1 au moment voulu.

Désactivation à partir d'actions synchrones

Exemple :

Quand un événement attendu ($R3) s'est produit et que l'état "Butée atteinte" (variable système $AA_FXS) est réalisé, FXS doit être désactivé.

Code de programme

```
IDS=4 WHENEVER (($R3==1) AND ($AA_FXS[Y]==1)) DO FXS[Y]=0
FA[Y]=1000 POS[Y]=0
```

Butée atteinte

Lorsque la butée a été accostée :

- La distance restant à parcourir est supprimée et la consigne de position est alignée.
- Le couple de l'entraînement croît jusqu'à la valeur limite `FXSW`, puis reste constant à cette valeur.
- La surveillance d'immobilisation en butée est activée à l'intérieur de la largeur de fenêtre attribuée.

Autres conditions à prendre en compte

- Mesure avec effacement de la distance restant à parcourir
  "Mesure avec effacement de la distance restant à parcourir" (instruction `MEAS`) et "Accostage d'une butée" ne peuvent pas être programmés dans un même bloc.
  Exception : Une des fonctions agit sur un axe à interpolation et l'autre sur un axe de positionnement ou les deux agissent sur des axes de positionnement.
- Contrôle du contour
  Aussi longtemps que la fonction "Accostage d'une butée" est active, il n'y a aucune surveillance du contour.
- Axes de positionnement
  Lorsque la fonction "Accostage d'une butée" est exécutée avec des axes de positionnement, le changement de bloc intervient indépendamment du déplacement vers la butée.
- Axes Link et axes de conteneurs
  L'accostage d'une butée est également autorisé pour les axes Link et les axes conteneurs. L'état de l'axe machine affecté est conservé au-delà des rotations conteneur. Ceci est également valable pour la limitation de couple à effet modal avec FOCON.
- L'accostage d'une butée est **impossible** dans les cas suivants :
  - Dans le cas des axes Gantry
  - Pour des axes concurrents de positionnement exclusivement pilotés par l'AP (l'activation de `FXS` doit avoir lieu dans le programme CN).
- Quand la limite de couple est trop basse, l'axe peut ne plus suivre les valeurs de consigne prescrites, le régulateur de position atteint la limite et l'écart par rapport au contour programmé augmente. Des déplacements par à-coup peuvent alors survenir lorsque la limite de couple augmente. Pour s'assurer que l'axe peut encore suivre les valeurs de consigne, il convient de vérifier que l'écart par rapport au contour programmé n'est pas supérieur à ce qu'il serait si le couple n'était pas limité.

2.14.7 Arrêt temporisé (G4)

L'instruction `G4` permet de programmer dans un bloc un délai (arrêt temporisé) qui court à partir du moment où le bloc est exécuté. Le passage au bloc suivant s'effectue dès que le délai est entièrement écoulé.

Remarque

`G4` interrompt le contournage.

Syntaxe

```
G4 F<Time>
G4 S<NumSpi>
G4 S<n> = <NumSpi>
```

Signification

`G4` :	Activation de l'arrêt temporisé	
	Uniquement dans le bloc :	oui
`F<Time>` :	L'indication de l'arrêt temporisé `<Time>` s'effectue en secondes sous l'adresse `F`.	
`S<NumSpi>`:	L'indication de l'arrêt temporisé en tours de broche `<NumSpi>` par rapport à la broche principale actuelle s'effectue sous l'adresse S.	
`S<n>=NumSpi>` :	L'indication de l'arrêt temporisé en tours de broche `<NumSpi>` par rapport à la broche adressée avec l'extension d'adresse `<n>` s'effectue sous l'adresse S.	

Remarque

Les adresses `F` et `S` utilisées pour l'horodatage dans le bloc d'arrêt temporisé `G4` n'ont pas d'incidence sur les avances `F...` et les vitesses de rotation de broche `S...` du programme.

Autres conditions à prendre en compte

Actions synchrones

Dans un programme, deux actions synchrones sont programmées de manière que le bloc suivant avec arrêt temporisé devienne le bloc d'action dans lequel les actions synchrones sont exécutées. La première action synchrone est une action synchrone modale. L'autre action synchrone est une action synchrone non modale. Si l'action synchrone non modale doit influencer l'action synchrone modale, par ex. déblocage de l'usinage via UNLOCK, **au moins deux temps de cycle d'interpolation**, par ex. `G4 F<temps de cycle d'interpolation * 2>`, doivent être disponibles comme arrêt temporisé effectif.

L'arrêt temporisé effectif dépend du réglage du paramètre machine PM10280 $MN_PROG_FUNCTION_MASK, bit 4 = <valeur>

Valeur	Signification
0	L'arrêt temporisé effectif est égal à l'arrêt temporisé programmé
1	L'arrêt temporisé effectif est égal à l'arrêt temporisé programmé, arrondi au multiple immédiatement supérieur du temps de cycle d'interpolation (PM10071 $MN_IPO_CYCLE_TIME)

Exemple de programme :

- PM10071 $MN_IPO_CYCLE_TIME == 8 ms
- PM10280 $MN_PROG_FUNCTION_MASK, bit 4 = 1

```
Code de programme                          Commentaire
N10 WHEN TRUE DO LOCK(1)                   ; SynAct non modale : LOCK de la
                                           ; SynAct modale ID=1
N20 G4 F2                                  ; Bloc d'action pour SynAct de N10
N30 WHEN TRUE DO UNLOCK(1)                 ; SynAct non modale : UNLOCK
                                           ; de la SynAct modale ID=1
N40 ID=1 WHENEVER TRUE DO $R0=1 RDISABLE   ; SynAct modale ID=1
                                           ; Paramètre R R0=1
                                           ; Activation du blocage de la lecture
N50 G4 F0.012                              ; Bloc d'action pour SynAct de N40 et N50
                                           ; Voir ci-dessous la section "Description"
N60 G4 F10
```

Description

Le comportement souhaité est que l'action synchrone non modale de N30 annule le blocage actif (LOCK) de l'action synchrone modale avec ID=1 de N40 et donc que le paramètre R soit écrit dans N50 et que le blocage de la lecture devienne effectif. Cependant, ce comportement n'est obtenu que lorsque l'arrêt temporisé effectif correspond au moins à deux temps de cycle d'interpolation.

L'arrêt temporisé effectif est déterminé par l'arrêt temporisé programmé, le temps de cycle d'interpolation et le réglage dans PM10280 $MN_PROG_FUNCTION_MASK, bit 4. Pour que l'arrêt temporisé effectif corresponde au moins à deux temps de cycle d'interpolation, l'arrêt temporisé suivant doit être programmé :

- Bit 4 == 0 : Arrêt temporisé programmé ≥ 2 * temps de cycle d'interpolation
- Bit 4 == 1 : Arrêt temporisé programmé ≥ 1,5 * temps de cycle d'interpolation

Si l'arrêt temporisé effectif est inférieur à deux temps de cycle d'interpolation, l'écriture du paramètre R et le blocage de la lecture ne sont exécutés que dans le bloc N60.

Exemple

```
Code de programme          Commentaire
N10 G1 F200 Z-5 S300 M3    ; Avance F, vitesse de rotation de la broche S
N20 G4 F3                  ; Arrêt temporisé : 3s
```

Code de programme	Commentaire
N30 X40 Y10	
N40 G4 S30	; Attendre 30 rotations de la broche (pour S = 300 tr/min et une correction de la vitesse de rotation de 100 %, cela correspond à : t = 0,1 min).
N50 X...	; L'avance et la vitesse de rotation de la broche programmées dans N10 continuent à être actives.

2.14.8 Arrêt interne du prétraitement des blocs

Fonction

Lors de l'accès à des données d'état de la machine ($A...), la commande déclenche un arrêt du prétraitement des blocs. Le bloc suivant n'est exécuté que lorsque tous les blocs prétraités et mémorisés auparavant ont été exécutés en entier. Un arrêt précis (comme G9) a lieu dans le bloc précédent.

Exemple

Code de programme	Commentaire
...	
N40 POSA[X]=100	
N50 IF $AA_IM[X]==R100 GOTOF MARKE1	; Accès à des données d'état de la machine ($A...), la commande déclenche un arrêt interne du prétraitement des blocs.
N60 G0 Y100	
N70 WAITP(X)	
N80 MARQUE1:	
...	

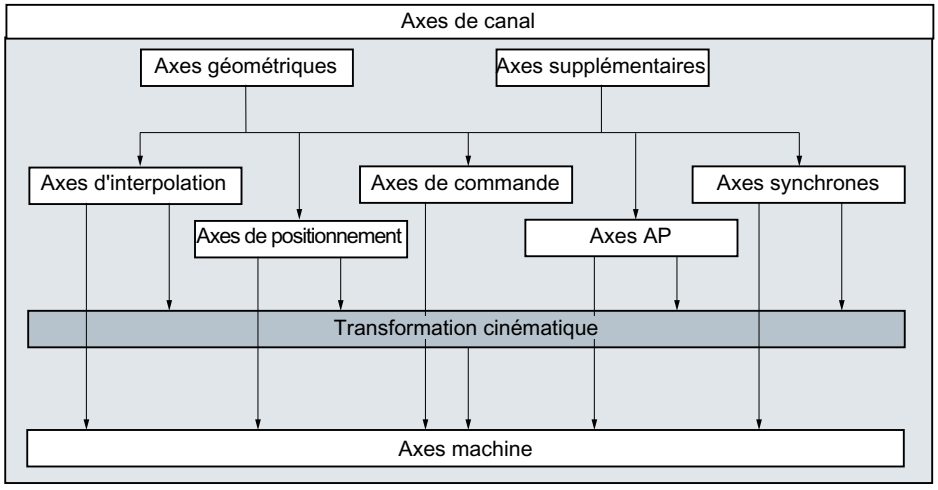
2.15 Informations diverses

2.15.1 Axes

Types d'axes

Dans le cadre de la programmation, on fait la distinction entre les types d'axe suivants :

- Axes principaux / axes géométriques
- Axes supplémentaires
- Broche principale, broche maître
- Axes machine
- Axes de canal
- Axes à interpolation
- Axes de positionnement
- Axes synchrones
- Axes de commande
- Axes AP / axes de positionnement concurrents
- Axes Link
- Axes Lead-Link



2.15.1.1 Axes principaux / Axes géométriques

Les axes principaux définissent un système de coordonnées orthogonal direct. C'est dans ce système de coordonnées que les déplacements d'outil seront programmés.

En commande numérique, les axes principaux sont appelés axes géométriques. Cette expression est également utilisée dans ce manuel.

Axes géométriques permutables

La fonction "Axes géométriques permutables" (voir Description fonctionnelle Notions complémentaires) permet de modifier, dans le programme pièce, le groupe d'axes géométriques configuré par paramètre machine. Un axe de canal défini comme axe supplémentaire synchrone peut remplacer un axe géométrique quelconque.

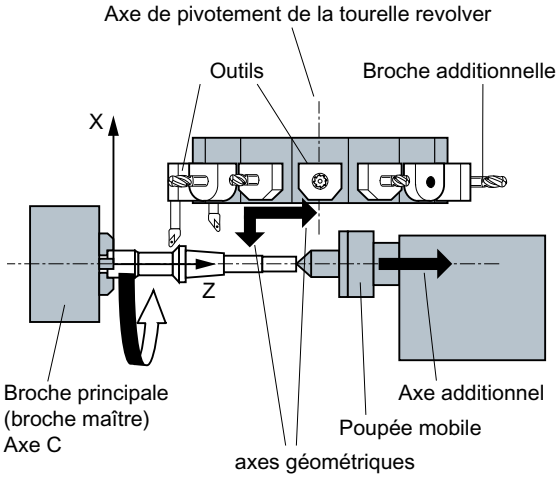
Descripteur d'axe

Le nom/descripteur d'un axe géométrique peut être défini par le paramètre machine suivant :

PM20060 $MC_AXCONF_GEOAX_NAME_TAB (nom d'axe géométrique dans le canal)

Désignation standard sur les tours :

1. Axe géométrique : X
2. Axe géométrique : Z



Désignation standard sur les fraiseuses :

1. Axe géométrique : X
2. Axe géométrique : Y
3. Axe géométrique : Z

Informations complémentaires

On utilise au maximum trois axes géométriques pour programmer les frames et la géométrie de la pièce (contour).

Les descripteurs des axes géométriques et des axes de canal peuvent être identiques, dans la mesure où une transformation cinématique est possible.

Les noms des axes géométriques et des axes de canal peuvent être identiques dans tous les canaux. Ainsi, un programme peut être exécuté dans n'importe quel canal.

2.15.1.2 Axes supplémentaires

Contrairement aux axes géométriques, aucun lien géométrique n'existe entre les axes supplémentaires.

Axes supplémentaires typiques :

- Axes pour tourelles revolver d'outil
- Axes pour tables pivotantes
- Axes pour têtes pivotantes
- Axes de chargeurs

Descripteur d'axe

Dans le cas d'un tour à tourelle revolver, p .ex. :

- Position de revolver U
- Contre-pointe V

Exemple de programmation

Code de programme	**Commentaire**
`N10 G1 X100 Y20 Z30 A40 F300`	`; Déplacements d'axes en interpolation.`
`N20 POS[U]=10POS[X]=20 FA[U]=200 FA[X]=350`	`; Déplacements d'axes de positionnement.`
`N30 G1 X500 Y80 POS[U]=150FA[U]=300 F550`	`; Axe en interpolation et axe de positionnement.`
`N40 G74 X1=0 Z1=0`	`; Accostage du point de référence.`

2.15.1.3 Broche principale, broche maître

La broche considérée comme broche principale est déterminée par la cinématique de la machine. Cette broche est généralement déclarée broche maître par un paramètre machine.

Cette affectation peut être modifiée par de l'instruction de programme `SETMS(<numéro de broche>)`. Avec `SETMS` sans indication du numéro de broche, il est possible de retourner à la broche maître définie dans le paramètre machine.

Des fonctions spéciales telles que le filetage à l'outil s'appliquent à la broche maître.

Descripteur de broche

S ou S0

2.15.1.4 Axes machine

Les axes machines sont les axes présents physiquement sur la machine.

Le déplacement programmé d'un axe à interpolation ou supplémentaire peut agir sur plusieurs axes machine par l'intermédiaire d'une transformation active dans le canal (TRANSMIT, TRACYL ou TRAORI).

Les axes machine ne sont adressés directement dans le programme que dans des cas spéciaux (par exemple pour l'accostage du point de référence ou d'un point fixe).

Descripteur d'axe

Le nom/descripteur d'un axe machine peut être défini par le paramètre machine spécifique à la CN suivant :

PM10000 $MN_AXCONF_MACHAX_NAME_TAB (nom d'axe machine)

Réglage par défaut : X1, Y1, Z1, A1, B1, C1, U1, V1

Les axes machine possèdent en outre des descripteurs fixes qui peuvent toujours être utilisés indépendamment du nom réglé dans le paramètre machine :

AX1, AX2, ..., AX<n>

2.15.1.5 Axes de canal

Tous les axes géométriques, supplémentaires et machine affectés à un canal sont appelés axes de canal.

Descripteur d'axe

Le nom/descripteur spécifique au canal d'un axe géométrique ou supplémentaire peut être défini par le paramètre machine suivant :

PM20080 $MC_AXCONF_CHANAX_NAME_TAB (nom de l'axe de canal)

Réglage par défaut : X, Y, Z, A, B, C, U, V

L'affectation de l'axe machine auquel correspond un axe géométrique ou supplémentaire dans le canal est défini dans le paramètre machine suivant :

PM20070 $MC_AXCONF_MACHAX_USED (axes machine utilisés)

2.15.1.6 Axes à interpolation

Les axes à interpolation décrivent la trajectoire et, par conséquent, le déplacement de l'outil dans l'espace.

L'avance programmée est appliquée sur cette trajectoire. Les axes qui participent à la trajectoire atteignent leur position finale simultanément. En règle générale, il s'agit des axes géométriques.

Cependant, les axes qui deviendront axes à interpolation et qui détermineront ainsi la vitesse, sont définis lors du préréglage de la machine.

Dans le programme CN, les axes à interpolation peuvent être spécifiés par `FGROUP`.

Pour plus d'informations sur `FGROUP`, référez-vous à "Avance (G93, G94, G95, F, FGROUP, FL, FGREF) (Page 108)".

2.15.1.7 Axes de positionnement

Les axes de positionnement sont interpolés séparément, c'est à dire que chaque axe de positionnement possède son propre interpolateur d'axe et sa propre avance. Les axes de positionnement ne sont pas en interpolation avec les axes à interpolation.

Les axes de positionnement sont pilotés par le programme CN ou par l'AP. Quand un axe est appelé à être piloté simultanément par la CN et l'AP, un message d'erreur apparaît.

Axes typiques de positionnement :

- Dispositif de chargement de pièces
- Dispositif de déchargement de pièces
- Magasin d'outils/Tourelle revolver

Types

Il y a lieu de faire la distinction entre les axes de positionnement avec synchronisation en fin de bloc et ceux avec synchronisation après plusieurs blocs.

Axes POS

le changement de bloc a lieu en fin de bloc, lorsque tous les axes à interpolation et de positionnement programmés dans ce bloc ont atteint le point final programmé.

Axes POSA

les déplacements des axes de positionnement peuvent se dérouler sur plusieurs blocs.

Axes POSP

le déplacement de ces axes de positionnement pour accoster la position finale se fait en plusieurs parties.

Remarque

Les axes de positionnement deviennent des axes synchrones quand ils sont déplacés sans l'identificateur spécifique POS/POSA.

Le contournage (G64) n'est possible pour les axes à interpolation que si les axes de positionnement (POS) atteignent leur position finale avant les axes à interpolation.

Les axes à interpolation programmés avec `POS`/`POSA` sont extraits du groupe d'axes à interpolation pour ce bloc.

Pour plus d'informations sur `POS`, `POSA` et `POSP`, référez-vous à "Déplacement des axes de positionnement (POS, POSA, POSP, FA, WAITP, WAITMC) (Page 116)".

2.15.1.8 axes synchrones

Les axes synchrones se déplacent de façon synchrone avec les axes à interpolation depuis la position de départ jusqu'à la position finale programmée.

L'avance programmée sous`F` est valable pour tous les axes à interpolation programmés dans le bloc, cependant pas pour les axes synchrones. Pour effectuer leur course, les axes synchrones ont besoin du même temps que les axes à interpolation.

Un axe synchrone peut être, par exemple, un axe rotatif qui est déplacé en synchronisme avec l'interpolation d'un contour.

2.15.1.9 Axes de commande

Les axes de commande sont démarrés par des actions synchrones à la suite d'un événement (commandes). Ils peuvent être positionnés, activés et arrêtés de façon totalement asynchrone avec le programme pièce. Il n'est pas possible de déplacer un axe simultanément à partir du programme pièce et d'une action synchrone.

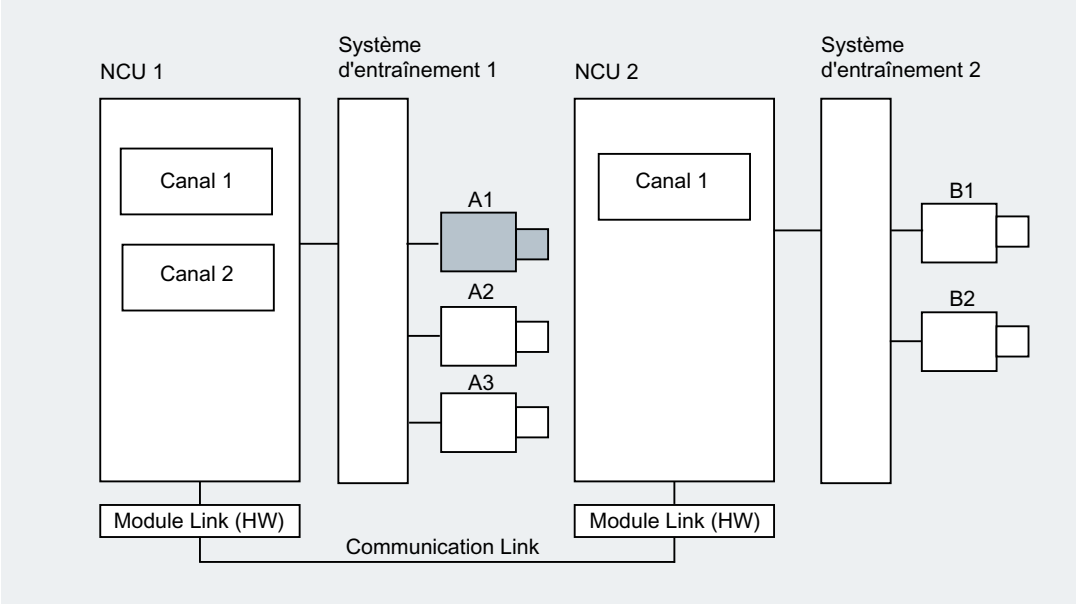
Les axes de commande sont interpolés séparément, c'est à dire que chaque axe de commande possède son propre interpolateur d'axe et sa propre avance.

2.15.1.10 Axes AP

Le déplacement des axes AP est piloté par l'AP, par le biais de blocs fonctionnels spécifiques dans le programme de base et peut être exécuté de façon asynchrone avec tous les autres axes. Les déplacements sont exécutés indépendamment des axes synchrones et des axes à interpolation.

2.15.1.11 les axes Link

Les axes Link sont des axes raccordés physiquement à une autre NCU qui assure leur asservissement de position. Ils peuvent être affectés dynamiquement à des canaux d'une **autre** NCU. Du point de vue d'une NCU déterminée, les axes Link ne sont pas des axes locaux.



Le concept de **conteneur d'axes** sert à la modification dynamique de l'affectation à une NCU. La permutation d'axe avec les instructions `GET` et `RELEASE` du programme pièce n'est **pas** disponible pour les axes Link.

Plus d'informations

Conditions

- Les NCU concernées, NCU1 et NCU2 doivent être reliées par l'intermédiaire de cartes Link à communication Link rapide.
- L'axe doit être configuré de façon adéquate par paramètre machine.
- L'option "Axe link" doit être disponible.

Description

L'asservissement de position a lieu dans la NCU sur laquelle l'axe est raccordé physiquement à l'entraînement. Dans cette NCU se trouve l'interface VDI d'axe correspondante. Dans le cas des axes Link, les consignes de position sont générées sur une autre NCU puis transférées par l'intermédiaire de cartes NCU-Link.

La communication Link doit assurer la coordination entre les interpolateurs et le régulateur de position ou l'interface AP. Les consignes calculées par les interpolateurs doivent être transmises à la boucle d'asservissement de position sur la NCU hôte ou les valeurs réelles doivent être transmises en retour.

Conteneur d'axes

Un conteneur d'axes est un tampon FIFO dans lequel a lieu l'affectation d'axes locaux et/ou d'axes link à des canaux. Les inscriptions figurant dans ce tampon FIFO peuvent faire l'objet d'un **décalage cyclique**.

La configuration des axes link autorise, outre le renvoi direct à des axes locaux ou link, le renvoi à des conteneurs d'axes dans la table d'affectation des axes machine. Un tel renvoi comprend :

- le numéro du conteneur **et**
- l'emplacement (le positionnement dans le tampon FIFO à l'intérieur du conteneur)

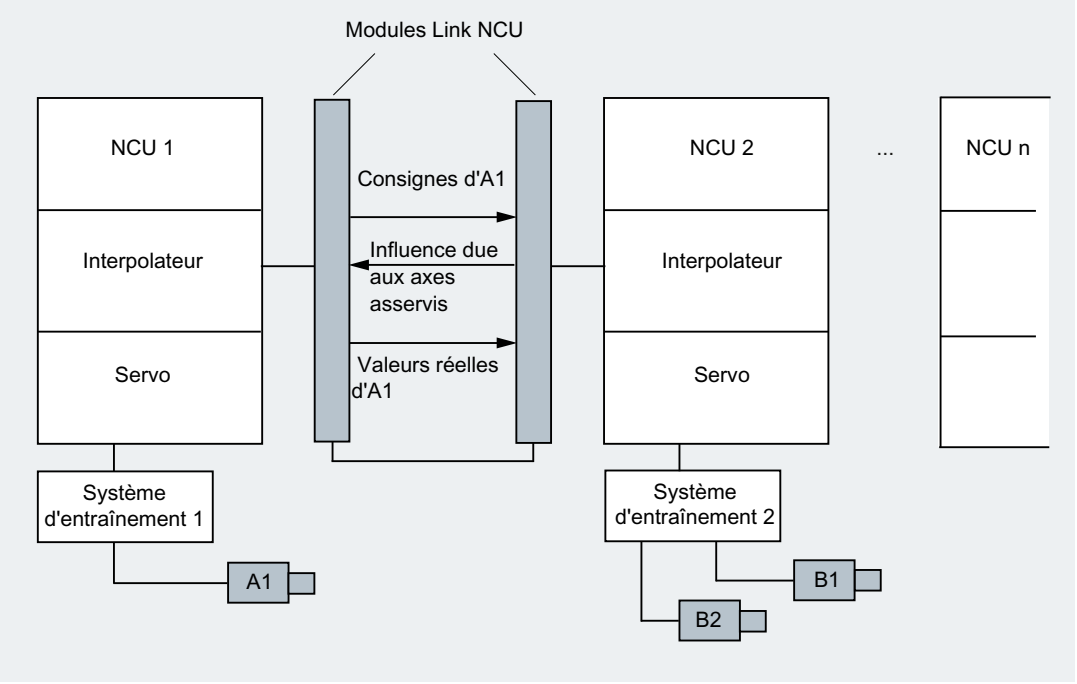
Une entrée du tampon FIFO peut être :

- un axe local **ou**
- un axe link

Les entrées de conteneur d'axes sont des axes machine locaux ou des axes link du point de vue d'une NCU donnée. Les entrées dans la table d'affectation des axes machine (MD10002 $MN_AXCONF_LOGIC_MACHAX_TAB) d'une NCU donnée sont fixes.

2.15.1.12 Axes Lead-Link

Un axe Link-Lead est un axe qui est interpolé par une NCU, mais qui est utilisé par une ou plusieurs autres NCU comme axe pilote pour la commande d'axes asservis.



Un signal d'alarme axial du régulateur de position est envoyé à toutes les autres NCU qui ont un rapport avec l'axe concerné au travers d'un axe Link-Lead.

Les NCU qui utilisent l'axe Lead-Link peuvent lui être couplées de la manière suivante :

- Valeur pilote (de consigne, réelle ou simulée)
- Déplacements conjugués
- Asservissement tangentiel
- Réducteur électronique (ELG)
- Broche synchrone

Programmation

NCU pilote :

Le déplacement de l'axe pilote ne peut être programmé que par la seule NCU qui lui est raccordée physiquement. Aucune mesure particulière n'est à observer pour la programmation.

NCU des axes asservis :

La programmation dans la NCU des axes asservis ne doit contenir aucune instruction de déplacement destinée à l'axe Lead-Link (axe pilote). Toute manquement à cette règle entraîne l'émission d'une alarme.

L'axe Lead-Link est interpellé de la manière habituelle par le biais du descripteur d'axe de canal. Les états de l'axe Lead-Link sont fournis par des variables système bien définies.

Plus d'informations

Conditions

- Les NCU concernées, NCU1 à NCU<n> (<n> = 8 au maximum) doivent être reliées par l'intermédiaire de cartes link à communication link rapide.
  Bibliographie :
  Manuel Configuration NCU
- L'axe doit être configuré de façon adéquate par paramètre machine.
- L'option "Axe link" doit être disponible.
- Les NCU concernées doivent toutes être configurées avec la même période d'appel de l'interpolateur.

Restrictions

- Un axe pilote faisant office d'axe Lead-Link ne peut pas être un axe link, en d'autres termes, il ne peut pas être déplacé par d'autres NCU que sa NCU d'origine.
- Un axe pilote faisant office d'axe Lead-Link ne peut pas être un axe conteneur, en d'autres termes il ne peut pas être sollicité alternativement par différentes NCU.
- Un axe Lead-Link ne peut pas être un axe directeur programmé appartenant à une association d'axes Gantry.
- Les couplages avec des axes Lead-Link ne peuvent pas être cascadés.
- La permutation d'axes n'est possible qu'au sein de la NCU d'affectation d'origine de l'axe Lead-Link.

Variables système

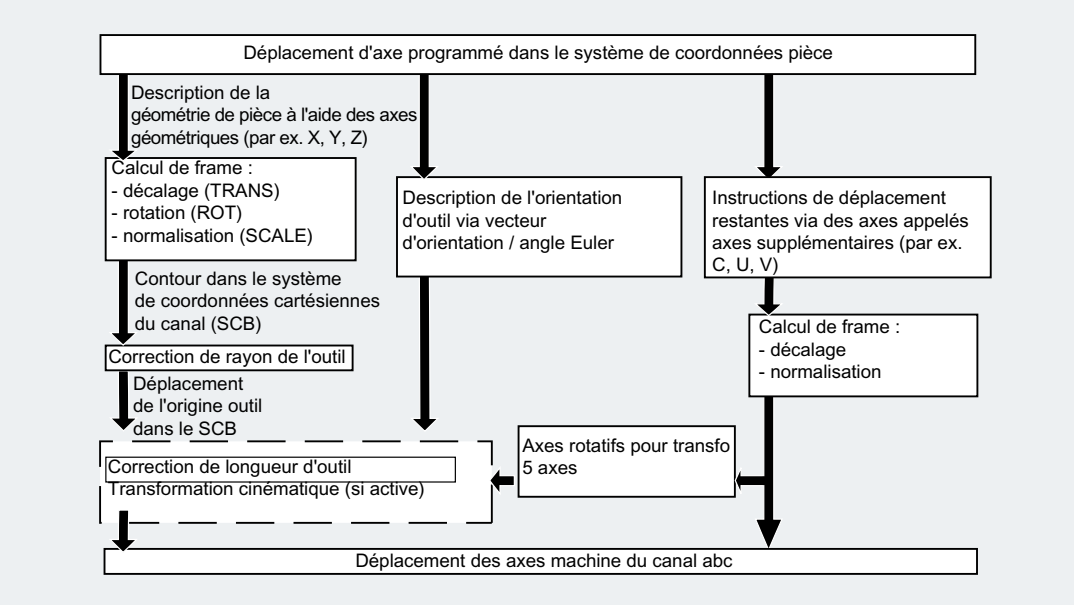
Variables système pouvant être utilisées avec le descripteur d'axe de canal de l'axe Lead-Link :

Variable système	Signification
$AA_LEAD_SP	Valeur pilote simulée - Position
$AA_LEAD_SV	Valeur pilote simulée - vitesse

Quand ces variables système sont actualisées par la NCU de l'axe pilote, les nouvelles valeurs sont transmises aux NCU qui sont appelées à déplacer les axes asservis à cet axe pilote.

2.15.2 De l'instruction de déplacement au déplacement de la machine

La figure suivante illustre le rapport entre les déplacements programmés d'axes (instructions de déplacement) et le déplacement résultant de la machine.

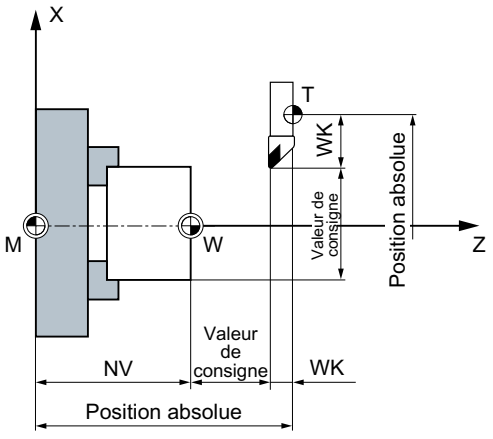


2.15.3 Calcul de la distance

Le calcul de la distance détermine le trajet à effectuer dans un bloc en tenant compte de tous les décalages et de toutes les corrections.

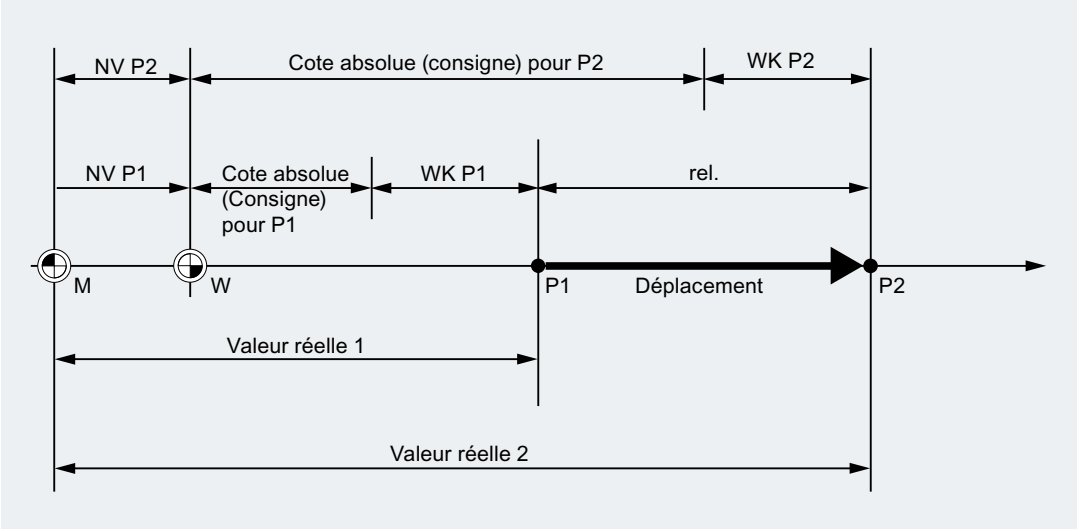
Règle générale :

Distance =
Valeur de consigne - Valeur réelle + Décalage d'origine (DO) + Correction d'outil (CO)



Si on programme, dans un nouveau bloc, un nouveau décalage d'origine et un nouveau correcteur d'outil, on a :

- dans le cas d'une saisie de cotes absolues :
  Distance = (Cote absolue P2 - Cote absolue P1) + (DO P2 - DO P1) + (CO P2 - CO P1).
- dans le cas d'une saisie de cotes relatives :
  Distance = Cote relative + (DO P2 - DO P1) + (CO P2 - CO P1).



2.15.4 Adresses

Adresses fixes

La configuration de ces adresses est fixe, ce qui signifie que les symboles d'adresse ne peuvent pas être modifiés.

Une liste figure dans le tableau "Adresses fixes (Page 1324)".

Adresses réglables

Par le biais d'un paramètre machine, le constructeur de machines peut affecter un autre nom à ces adresses.

Remarque

Au sein de la commande, les adresses réglables doivent être définies de manière univoque, autrement dit, un même nom d'adresse ne peut pas être utilisé pour différents types d'adresses (valeurs d'axe et points finaux, orientation de l'outil, paramètres d'interpolation, etc.).

Une liste figure dans le tableau "Adresses réglables (Page 1329)".

modales / non modales

Une adresse modale avec sa valeur programmée reste valide au-delà du bloc dans lequel elle est programmée, jusqu'à ce qu'une nouvelle valeur soit programmée sous la même adresse.

Une adresse non modale est valide dans un seul bloc, le bloc dans lequel elle a été programmée.

Exemple :

```
Code de programme          Commentaire
N10 G01 F500 X10
N20 X10                    ; L'avance F de N10 reste valide jusqu'à
                            la programmation d'une nouvelle
                            avance.
```

Adresses avec extension axiale

Dans une adresse avec extension axiale, l'adresse est suivie d'un nom d'axe entre crochets qui précise l'affectation à des axes.

Exemple :

```
Code de programme          Commentaire
FA[U]=400                  ; Avance spécifique à l'axe U
```

Voir aussi le tableau "Adresses fixes (Page 1324)".

Notation à adresse étendue

L'utilisation des adresses avec extension offre la possibilité d'intégrer un plus grand nombre d'axes et de broches dans une systématique.

Une adresse étendue se compose d'une extension numérique et d'une expression arithmétique séparée par le signe "=". L'extension numérique compte un ou deux chiffres et est toujours positive.

La notation à adresse étendue est admise uniquement pour les adresses simples suivantes :

Adresse	Signification
X, Y, Z, ...	Adresses d'axe
I, J, K	Paramètres d'interpolation
S	Vitesse de rotation de broche
SPOS, SPOSA	Position de broche
M	Fonctions supplémentaires
H	Fonctions auxiliaires
T	numéro d'outil
F	Avance

Exemples :

Code de programme	Commentaire
X7	; Le signe "=" n'est pas requis, mais possible également ici ; 7 est la valeur.
X4=20	; Axe X4, "=" requis
CR=7.3	; 2 lettres, "=" requis
S1=470	; Vitesse de rotation pour la 1ère broche : 470 tr/min
M3=5	; Arrêt de broche pour la 3è broche

Dans le cas des adresses M, H, S, SPOS et SPOSA, l'extension numérique est remplaçable par une variable. Le descripteur de variable est alors à écrire entre crochets.

Exemples :

Code de programme	Commentaire
S[SPINU]=470	; Vitesse de rotation pour la broche dont le numéro est précisé dans la variable SPINU
M[SPINU]=3	; Rotation en sens horaire pour la broche dont le numéro est précisé dans la variable SPINU
T[SPINU]=7	; Présélection de l'outil pour la broche dont le numéro est précisé dans la variable SPINU

2.15.5 Noms

Les instructions selon DIN 66025 sont complétées dans le langage évolué CN entre autres par les objets indiqués.

Exemples d'objets désignés :

- Variables système
- Variables définies par l'utilisateur
- Axes/broches
- Sous-programmes
- Mots-clés
- Marques de saut
- Macros

Remarque

Les descripteurs doivent être univoques. Un même descripteur ne doit **pas** être utilisé pour désigner plusieurs objets.

Règles de dénomination

Un nom peut être choisi librement dans la mesure où les règles suivantes sont respectées :

- Caractères autorisés :
  - Caractères alphabétiques : A ... Z, a ... z
  - Chiffres : 0 ... 9
  - Tiret bas : _
- Les deux premiers caractères doivent être obligatoirement des lettres ou des tirets bas.
- Longueur maximale :
  - Noms de programme : (Page 43) 24 caractères
  - Noms d'axe : 8 caractères
  - Noms de variable : 31 caractères

Remarque

Les mots-clés réservés ne doivent pas être utilisés comme descripteurs.

Cycles

Pour éviter les conflits de noms, il est recommandé de tenir compte de la définition suivante lors de l'attribution de noms aux cycles utilisateur :

Chaîne de caractères	réservé pour les noms de
• CYCLE • CUST_ • GROUP_ • _ • S_ • E_ • F_	Cycles SIEMENS
• CCS_	Cycles de compilation SIEMENS
• CC_	Cycles de compilation utilisateur

Cycles utilisateur

Concernant les noms des cycles utilisateur, il est recommandé qu'ils commencent par "U_".

Variables

Pour plus d'informations sur l'attribution de noms aux variables, voir les chapitres suivants :

- Données système (Page 396)
- Définition de variables utilisateur (DEF) (Page 403)

2.15.6 Constante

Constante (généralités)

Une constante est un élément de donnée dont la valeur ne change pas lors de l'exécution d'un programme, par ex. une affectation de valeur à une adresse.

Constante décimale

La valeur numérique d'une constante décimale est représentée dans le système décimal.

Constante INTEGER

Une constante INTEGER est une valeur entière, c.-à-d. une série de chiffres avec ou sans signe et sans virgule.

Exemples :

`X10`	Affectation de la valeur +10 à l'adresse X
`X-35`	Affectation de la valeur -35 à l'adresse X
`X0`	Affectation de la valeur 0 à l'adresse X **Remarque :** X0 ne peut pas être remplacé par X.

Constante REAL

Une constante REAL est une série de chiffres avec ou sans signe, avec une virgule et avec ou sans exposant.

Exemples :

`X10.25`	Affectation de la valeur +10,25 à l'adresse X
`X-10.25`	Affectation de la valeur -10,25 à l'adresse X
`X0.25`	Affectation de la valeur +0,25 à l'adresse X
`X.25`	Affectation de la valeur +0,25 à l'adresse X, sans "0" en tête
`X=-.1EX-3`	Affectation de la valeur $-0{,}1*10^{-3}$ à l'adresse X

Remarque

Si, dans le cas d'une adresse avec notation décimale admise, vous écrivez un nombre de décimales supérieur au nombre prévu pour cette adresse, la valeur sera arrondie au nombre de décimales initialement fixé.

Constante hexadécimale

Les constantes interprétées comme des valeurs hexadécimales, c.-à-d. en base 16, sont également admises. Dans ce cas, les lettres "A" à "F" sont représentatives des chiffres hexadécimaux avec les valeurs décimales 10 à 15.

Les constantes hexadécimales sont écrites entre apostrophes et sont précédées de la lettre "H" suivie de la valeur hexadécimale. Les séparateurs entre les lettres et les chiffres sont admis.

Exemple :

Code de programme	Commentaire
`$MC_TOOL_MANAGEMENT_MASK='H7F'`	`; L'affectation de la constante hexadécimale active les bits 0 à 7 dans le paramètre machine.`

Remarque

Le nombre maximum de caractères est limité par la plage de valeurs des données de type entier.

Constante binaire

Les constantes interprétées comme des valeurs binaires sont également admises. Dans ce cas, seuls les chiffres "0" et "1" sont utilisés.

Les constantes binaires sont écrites entre apostrophes et sont précédées de la lettre "B" suivie de la valeur binaire. Les séparateurs entre les chiffres sont admis.

Exemple :

Code de programme	Commentaire
`$MN_AUXFU_GROUP_SPEC='B10000001'`	`; L'affectation de la constante binaire active les bits 0 et 7 dans le paramètre machine.`

Remarque

Le nombre maximum de caractères est limité par la plage de valeurs des données de type entier.

2.15.7 Opérateurs et fonctions de calcul

Opérateurs

Opérateurs arithmétiques

Les variables systèmes de type RAL ou INT peuvent être traitées avec les opérateurs suivants :

Opérateur	Signification
+	Addition
-	Soustraction
*	Multiplication

Opérateur	Signification
/	• Division en actions synchrones : INT / INT ⇒ **INT** • Divisions en actions synchrones avec résultat de type REAL à l'aide de la fonction ITOR() : ITOR(INT) / ITOR(INT) ⇒ **REAL** • Divisions en programmes CN : INT / INT ⇒ **REAL**
DIV	Divisions d'entiers : INT / INT ⇒ **INT**
MOD	La division modulo, (uniquement type INT) livre le reste d'une division INT Exemple : 3 MOD 4 = 3

Remarque

Seules des variables de même type peuvent être combinées.

Opérateurs relationnels

Opérateur	Signification
==	égal à
>	différent de
<	inférieur à
>	supérieur à
<=	inférieur ou égal à
>=	supérieur ou égal à

Opérateurs booléens

Opérateur	Signification
NOT	NON
AND	ET
OR	OU
XOR	OU exclusif

Opérateurs logiques sur bits

Opérateur	Signification
B_OR	OU sur bits
B_AND	ET sur bits
B_XOR	OU exclusif sur bits
B_NOT	Négation sur bits

Priorité des opérateurs

Les opérateurs ont les priorités suivantes lors de l'exécution dans l'action synchrone (priorité la plus élevée : 1) :

Prio	Opérateurs	Signification
1	NOT, B_NOT	Négation, négation sur bits
2	*, /, DIV, MOD	Multiplication, division
3	+, -	Addition, soustraction
4	B_AND	ET sur bits
5	B_XOR	OU exclusif sur bits
6	B_OR	OU sur bits
7	AND	ET
8	XOR	OU exclusif
9	OR	OU
10	<<	Concaténation de chaînes de caractères, type de résultat STRING
11	==, <>, <, >, >=, <=	Opérateurs relationnels

Remarque

Dans les expressions à plusieurs opérateurs, il est fortement recommandé de grouper les opérateurs par des parenthèses "(...)" pour s'assurer qu'ils soient traités dans l'ordre souhaité.

Exemple de condition avec une expression à plusieurs opérateurs

Code de programme

```
... WHEN ($AA_IM[X] > VALUE) AND ($AA_IM[Y] > VALUE1) DO ...
```

Fonctions de calcul

Opérateur	Signification
SIN()	Sinus
COS()	Cosinus
TAN()	Tangente
ASIN()	Arc sinus
ACOS()	Arc cosinus
ATAN2(,)	Arc tangente 2
SQRT( )	racine carrée
ABS( )	Valeur absolue
POT( )	Puissance 2 (carré)
TRUNC( )	Partie entière Réglage de la précision des instructions de comparaison avec TRUNC
ROUND()	Arrondissement à un nombre entier
LN()	Logarithme naturel
EXP( )	Fonction exponentielle

Indexation

L'indice d'une variable système de type "Champ de ..." peut comporter une autre variable système. L'indice est alors pris en compte dans le traitement de bloc dans la période d'appel de l'interpolateur.

Exemple

```
Code de programme
... WHEN ... DO $AC_PARAM[ $AC_MARKER[1] ] = 3
```

Limitations

- Une imbrication de l'indexage avec d'autres variables système n'est pas autorisée.
- L'indice ne doit pas comporter de variable de prétraitement. L'exemple suivant n'est donc **pas** autorisé car $P_EP est une variable de prétraitement :
  $AC_PARAM[1] = $P_EP[$AC_MARKER[0]]

Voir aussi

Fonctions de calcul (Page 450)

Opérations (Page 450)

Notions complémentaires

3

3.1 Programmation CN flexible

3.1.1 Variables

L'utilisation de variables des domaines Données système et Données utilisateur, en particulier en association avec des fonctions de calcul et des structures de contrôle, permet de réaliser des programmes CN et des cycles de façon flexible.

⚠ ATTENTION

Dommages matériels et corporels dus à des variables modifiées

Lors de l'utilisation de variables dans le programme CN, tenir compte du fait que l'opérateur de la machine ou toute personne non autorisée avec un droit d'accès correspondant peut modifier les variables et par conséquent influencer l'exécution du programme. Il peut en résulter des dommages matériels et corporels.

- Afin d'éviter toute conséquence négative sur l'exécution du programme due à la modification des variables, prévoir des contrôles des données correspondants ("Input validation") dans le programme CN.

- Données système
  Les données système contiennent les variables prédéfinies dans le système. Ces variables ont une signification définie. Elles sont utilisées avant tout par le logiciel système. L'utilisateur peut lire et écrire ces variables dans les programmes CN et les cycles. Exemple : paramètres machine, données de réglage, variables système.
  Bien que la signification d'une donnée système soit prédéfinie et fixe, les propriétés peuvent être modifiées dans une certaine mesure par l'utilisateur via une redéfinition.
  Voir "Redéfinition de données système, données utilisateur et instructions de langage CN (REDEF) (Page 409)"
- Données utilisateur
  Les données utilisateur contiennent les variables définies par l'utilisateur, dont la signification est établie exclusivement par l'utilisateur. Elles ne sont pas exploitées par le système.
  Les données utilisateur sont réparties en :
  - Variables utilisateur prédéfinies
    Il s'agit de variables déjà définies dans le système, dont le nombre est paramétré au moyen des paramètres machine. L'utilisateur a la possibilité d'adapter les propriétés de ces variables.
    Voir "Redéfinition de données système, données utilisateur et instructions de langage CN (REDEF) (Page 409)".
  - Variables définies par l'utilisateur
    Il s'agit de variables définies par l'utilisateur et que le système crée lors de l'exécution. Leur nombre, type de données, visibilité ainsi que toutes les autres propriétés sont définies exclusivement par l'utilisateur.
    Voir "Définition de variables utilisateur (DEF) (Page 403)"

3.1.1.1 Données système

Les données système contiennent les variables prédéfinies dans le système, qui permettent d'accéder, dans les programmes CN et les cycles, au paramétrage actuel de la commande, ainsi qu'aux états de la machine, de la commande et du processus.

Variables de prétraitement

Il s'agit de données système qui sont lues et écrites dans le contexte du prétraitement, autrement dit au moment de l'interprétation du bloc dans lequel la variable est programmée. Les variables de prétraitement ne déclenchent pas d'arrêt de prétraitement.

Variables d'exécution

Il s'agit de données système qui sont lues ou écrites dans le contexte de l'exécution, autrement dit au moment de l'exécution du bloc dans lequel la variable est programmée. Les variables d'exécution sont des :

- Variables pouvant être programmées dans des actions synchrones (lecture/écriture)
- Variables pouvant être programmées dans le programme CN et capables de déclencher un arrêt de prétraitement (lecture/écriture)
- Variables pouvant être programmées dans le programme CN, et dont la valeur est déterminée durant le prétraitement mais n'est écrite que lors de l'exécution (synchrones à l'exécution : écriture seule).

Systématique de préfixe

Pour l'identification spécifique des données système, le nom est normalement précédé d'un préfixe formé par le caractère $, suivi d'une ou de deux lettres et d'un caractère de soulignement :

$ + 1ère Lettre	**Signification : type de donnée**
Données de prétraitement (Données système qui sont lues / écrites durant le prétraitement des blocs)	
$M	Paramètres machine[1)]
$S	Données de réglage, zones de protection [1)]
$T	Paramètres de gestion des outils
$P	Valeurs programmées
$C	Variables des cycles enveloppes ISO
$O	Données optionnelles
R	Paramètres R (paramètres de calcul)[2)]
Données d'exécution (données système qui sont lues / écrites durant l'exécution des blocs)	
$$M	Paramètres machine[1)]
$$S	Données de réglage [1)]
$A	Données d'exécution courantes
$V	Données du régulateur de position
$R	Paramètres R (paramètres de calcul)[2)]
[1)] Selon que les paramètres machine et les données de réglage sont précédés d'un ou de deux caractères $, ils sont traités comme des variables de prétraitement ou d'exécution. Le choix de la notation est libre et spécifique à l'application. [2)] Si un paramètre R est utilisé en tant que variable d'exécution dans le programme pièce / le cycle, aucun préfixe n'est spécifié, p. ex. R10. Lors de l'utilisation en tant que variable d'exécution dans une action synchrone, le préfixe est formé d'un caractère $, p. ex. $R10.	

2ème caractère	**Signification : Visibilité**
N	Variable globale CN (**C**N)
C	Variable spécifique au canal (**C**hannel)
A	Variable spécifique à l'axe (**A**xis)

Autres conditions à prendre en compte

Exceptions dans la systématique de préfixe

Les variables système ci-après s'écartent de la systématique de préfixe indiquée plus haut :

- $TC_... : Le deuxième caractère C ne désigne pas une variable spécifique au canal, mais une variable système spécifique au porte-outil (TC = Tool Carrier)
- $P_ ... : Variables système spécifiques à un canal

Utilisation de paramètres machine et de données de réglage dans les actions synchrones

Lors de l'utilisation de paramètres machine et de données de réglage dans les actions synchrones, il est possible de déterminer, au moyen du préfixe, si le paramètre machine ou la donnée de réglage sont écrits/lus de façon synchrone au prétraitement ou à l'exécution.

Si le paramètre reste inchangé pendant l'exécution, la lecture peut être effectuée de façon synchrone au prétraitement. Le préfixe du paramètre machine ou de la donnée de réglage s'écrit de ce fait avec un caractère $:

```
ID=1 WHENEVER $AA_IM[z] < $SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Si le paramètre est modifié pendant l'exécution, la lecture /écriture doit être effectuée de façon synchrone à l'exécution. Le préfixe du paramètre machine ou de la donnée de réglage s'écrit de ce fait avec deux caractères $:

```
ID=1 WHENEVER $AA_IM[z] < $$SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Remarque

Ecriture des paramètres machine et des données de réglage

Lors de l'écriture d'un paramètre machine ou d'une donnée de réglage, il faut s'assurer que le niveau d'accès actif pour l'exécution du programme pièce / cycle autorise la protection en écriture et que la prise d'effet du paramètre soit "IMMEDIATE".

Voir aussi

Variables (Page 395)

3.1.1.2 Variables utilisateur prédéfinies : Paramètres de calcul spécifiques à un canal (R)

Les paramètres de calcul ou paramètres R spécifiques au canal sont des variables utilisateur prédéfinies désignées par R et définies comme tableau de type de données REAL. Pour des raisons historiques, outre la notation des paramètres R avec un indice de tableau, p. ex. `R[10]`, la notation sans indice de tableau, p. ex. `R10`, est également autorisée.

En cas d'utilisation dans des actions synchrones, la lettre $ doit figurer en tête, p. ex. `$R10`.

Syntaxe

En cas d'utilisation comme variable de prétraitement :
`R<n>`

`R[<expression>]`

En cas d'utilisation comme variable d'exécution :
`$R<n>`
`$R[<expression>]`

Signification

`R` :	Descripteur en cas d'utilisation comme variable de prétraitement, p. ex. dans le programme pièce	
`$R` :	Descripteur en cas d'utilisation comme variable d'exécution, p. ex. dans des actions synchrones	
	Type :	REAL
	Plage de valeurs :	En notation non exponentielle : ± (0,000 0001 ... 9999 9999) **Remarque :** Le nombre maximum de décimales autorisées est de 8
		En notation exponentielle : ± ($1*10^{-300}$ ... $1*10^{+300}$) **Remarque :** • Notation : <Mantisse>EX<Exposant> p. ex. 8.2EX-3 • Le nombre maximum de caractères autorisés, y compris le signe et le point décimal, est de 10.
`<n>` :	Numéro du paramètre R	
	Type :	INT
	Plage de valeurs :	0 - MAX_INDEX **Remarque** MAX_INDEX résulte du nombre de paramètres R paramétrés : MAX_INDEX = (MD28050 $MN_MM_NUM_R_PARAM) - 1
`<expression>` :	Indice de tableau L'indice de tableau peut être une expression quelconque, à condition que le résultat de cette expression puisse être converti dans le type de données INT (INT, REAL, BOOL, CHAR)	

Exemple

Affectations de paramètres R et utilisation de paramètres R dans des fonctions mathématiques :

```
Code de programme          Commentaire
R0=3.5678                  ; Affectation dans le prétraitement des blocs
R[1]=-37.3                 ; Affectation dans le prétraitement des blocs
R3=-7                      ; Affectation dans le prétraitement des blocs
$R4=-0.1EX-5               ; Affectation dans l'exécution des blocs : R4 = -0.1 * 10^-5
$R[6]=1.874EX8             ; Affectation dans l'exécution des blocs : R6 = 1.874 * 10^8
R7=SIN(25.3)               ; Affectation dans le prétraitement des blocs

R[R2]=R10                  ; Adressage indirect au moyen d'un paramètre R
```

Code de programme	Commentaire
`R[(R1+R2)*R3]=5`	`; Adressage indirect au moyen d'une expression mathématique`
`X=(R1+R2)`	`; Déplacer l'axe X à la position résultant de la somme de R1 et de R2`
`Z=SQRT(R1*R1+R2*R2)`	`; Déplacer l'axe Z à la position racine carrée (R1^2 + R2^2)`

Voir aussi

Variables (Page 395)

3.1.1.3 Variables utilisateur prédéfinies : Paramètres de calcul globaux (RG)

Fonction

En plus des paramètres R spécifiques de canal, l'utilisateur disposent également de paramètres R globaux. Ils existent une fois au sein de la commande et peuvent être lus/écrits depuis tous les canaux.

Les paramètres globaux R sont, par exemple, utilisés pour transférer des informations d'un canal dans un autre. Les réglages globaux qui doivent être exploités pour tous les canaux en sont un autre exemple, comme la saillie de la pièce brute hors de la broche.

La lecture et l'écriture des paramètres R globaux s'effectuent via l'interface utilisateur ou dans le programme CN lors du prétraitement. Une utilisation dans des actions synchrones ou des cycles technologiques n'est pas possible.

Remarque

Lors de la lecture et de l'écriture des paramètres R globaux, **aucune** synchronisation n'intervient entre les canaux.

La lecture et l'écriture intervenant lors du prétraitement, l'instant auquel une valeur écrite issue d'un canal prend effet dans un autre canal n'est pas défini.

Exemple :

dans le canal 1, une boucle tourne avec un paramètre R global comme compteur de cycles. Le canal 2 écrit, dans ce paramètre R global, une valeur qui entraîne une interruption de la boucle dans le canal 1. Toutes les boucles interprétées jusqu'alors dans le canal 1 lors du prétraitement sont cependant toujours exécutées. Le nombre de boucles que cela représente n'est pas défini et dépend, entre autres, de l'utilisation du canal.

La synchronisation entre les canaux doit être réalisée de manière applicative par l'utilisateur lui-même, par exemple à l'aide de repères WAIT !

Syntaxe

Ecriture dans le programme CN

```
RG[<n>]=<valeur>
RG[<expression>]=<valeur>
```

Lecture dans le programme CN
`R...=RG[<n>]`
`R...=RG[<expression>]`

Signification

`RG` :	Nom par défaut de l'adresse CN pour paramètres R **globaux** **Remarque :** Le nom de l'adresse CN peut être réglé par le biais de PM15800 $MN_R_PARAM_NCK_NAME	
`<n>` :	Numéro du paramètre R global	
	Type :	INT
	Plage de valeurs :	0 ... MAX_INDEX **Remarque** MAX_INDEX résulte du nombre de paramètres R globaux paramétrés : MAX_INDEX = (PM18156 $MN_MM_NUM_R_PARAM_NCK) - 1
`<expression>` :	L'indice de tableau peut être une expression quelconque, à condition que le résultat de cette expression puisse être converti dans le type de données INT (INT, REAL, BOOL, CHAR)	
`<valeur>` :	Valeur du paramètre R global	
	Type :	REAL
	Plage de valeurs :	En notation non exponentielle : ± (0,000 0001 ... 9999 9999) **Remarque :** Le nombre maximum de décimales autorisées est de 8
		En notation exponentielle : ± ($1*10^{-300}$ ... $1*10^{+300}$) **Remarque :** • Notation : <Mantisse>EX<exponent> par ex. 8.2EX-3 • Le nombre maximum de caractères autorisés, y compris le signe et le point décimal, est de 10.

3.1.1.4 Variables utilisateur prédéfinies : variables Link

Dans le cadre de la fonction "Link NCU", les variables Link permettent de réaliser l'échange cyclique de données entre des NCU mises en réseau. Elle autorisent un accès spécifique au format de données à la mémoire de variables Link. La taille et la structure de données de la mémoire de variables Link sont définies de manière spécifique à l'installation par l'utilisateur / le constructeur de machine.

Les variables Link sont des variables utilisateur globales du système, dont l'écriture et la lecture peuvent être effectuées dans des programmes pièce et des cycles à partir de toutes les NCU du groupe Link, à condition que la communication Link soit configurée. Contrairement aux variables utilisateur globales (GUD), l'utilisation des variables Link est également possible dans des actions synchrones.

Pour des installations ne disposant pas de Link NCU actif, les variables Link peuvent être utilisées localement dans la commande en plus des variables utilisateur globales (GUD), comme variables utilisateur globales supplémentaires.

Syntaxe

```
$A_DLB[<indice>]
$A_DLW[<indice>]
$A_DLD[<indice>]
$A_DLR[<indice>]
```

Signification

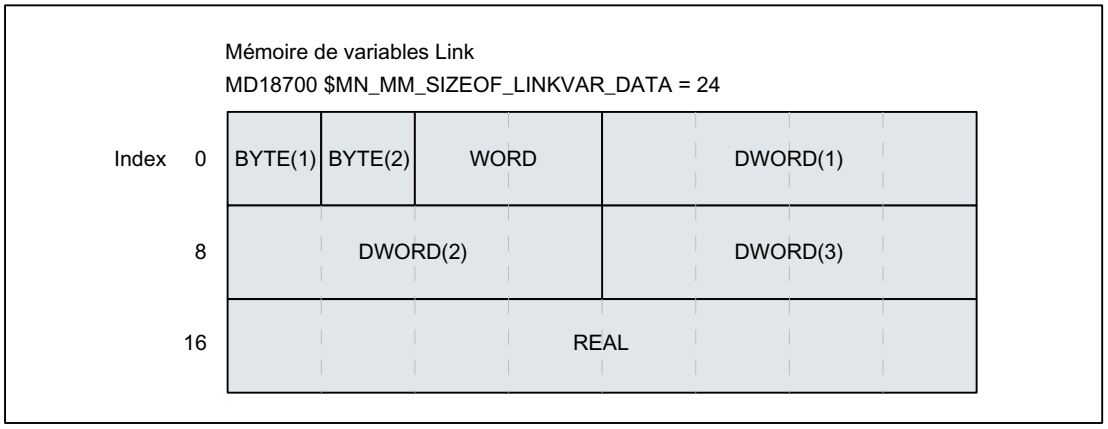
`$A_DLB` :	Variable Link pour le format de données BYTE (1 octet)	
	Type de données :	UINT
	Plage de valeurs :	0 ... 255
`$A_DLW` :	Variable Link pour le format de données WORD (2 octets)	
	Type de données :	INT
	Plage de valeurs :	-32768 ... 32767
`$A_DLD` :	Variable Link pour le format de données DWORD (4 octets)	
	Type de données :	INT
	Plage de valeurs :	-2147483648 ... 2147483647
`$A_DLR` :	Variable Link pour le format de données REAL (8 octets)	
	Type de données :	REAL
	Plage de valeurs :	$\pm(2{,}2*10^{-308}$ ... $1{,}8*10^{+308})$
`<indice>` :	Indice d'adresse en octets, calculé à partir du début de la mémoire de variables Link	
	Type de données :	INT
	Plage de valeurs :	0 - MAX_INDEX **Remarque** • MAX_INDEX résulte de la taille paramétrée pour la mémoire de variables Link : MAX_INDEX = (PM18700 $MN_MM_SIZEOF_LINKVAR_DATA) - 1 • Seule est autorisée la programmation d'indices pour lesquels les octets adressés dans la mémoire de variables Link correspondent à une limite de format de données ⇒ Indice = n * octets, avec n = 0, 1, 2, ... – `$A_DLB[i]` : i = 0, 1, 2, ... – `$A_DLW[i]` : i = 0, 2, 4, ... – `$A_DLD[i]` : i = 0, 4, 8, ... – `$A_DLR[i]` : i = 0, 8, 16, ...

Exemple

L'installation d'automatisation ne contient que 2 NCU (NCU1 et NCU2). L'axe machine AX2 déplacé comme axe Link par NCU2 est raccordé à NCU1.

NCU1 inscrit de façon cyclique la valeur réelle du courant ($VA_CURR) de l'axe AX2 dans la mémoire de variables Link. NCU2 lit de façon cyclique la valeur réelle du courant transmise par la communication Link et affiche l'alarme 61000 en cas de dépassement de la valeur limite.

La structure de données dans la mémoire des variables Link est représentée dans la figure suivante. La valeur réelle du courant est transmise au moyen de la valeur REAL.



NCU1

Durant la période d'appel de l'interpolateur pendant une action synchrone statique, NCU1 écrit de façon cyclique la valeur réelle du courant de l'axe AX2 dans la mémoire des variables Link, au moyen de la variable Link $A_DLR[16].

Code de programme

```
N111 IDS=1 WHENEVER TRUE DO $A_DLR[16]=$VA_CURR[AX2]
```

NCU2

Durant la période d'appel de l'interpolateur pendant une action synchrone statique, NCU2 lit de façon cyclique la valeur réelle du courant de l'axe AX2 à partir de la mémoire des variables Link, au moyen de la variable Link $A_DLR[16]. Si la valeur réelle du courant est supérieure à 23,0 A, l'alarme 61000 s'affiche.

Code de programme

```
N222 IDS=1 WHEN $A_DLR[16] > 23.0 DO SETAL(61000)
```

3.1.1.5 Définition de variables utilisateur (DEF)

L'instruction `DEF` permet de définir des variables spécifiques à l'utilisateur ou des variables utilisateur (User Data) et de leur affecter des valeurs.

Les catégories suivantes de variables utilisateur sont disponibles en fonction du domaine de validité, autrement dit du domaine dans lequel la variable est visible :

- Variables utilisateur locales (LUD)
  Les variables utilisateur locales (LUD) sont des variables définies dans un programme CN qui n'est pas le programme principal au moment de l'exécution. Elles sont déclarées à l'appel du programme CN et supprimées au reset de fin de programme ou au démarrage suivant de la commande. L'accès aux variables LUD peut uniquement être réalisé dans le programme CN dans lequel ces variables sont définies.
- Variables utilisateur globales d'un programme (PUD)
  Les variables utilisateur globales d'un programme (PUD) sont des variables définies dans un programme CN utilisé comme programme principal. Elles sont déclarées à l'appel du programme CN et supprimées au reset de fin de programme ou au démarrage suivant de la commande. L'accès aux variables PUD peut être réalisé dans le programme principal et dans tous ses sous-programmes.

  Remarque

  Disponibilité des variables utilisateur globales d'un programme (PUD)

  Les variables utilisateur globales définies dans le programme principal (PUD) ne sont également disponibles dans les sous-programmes que si le paramètre machine suivant est mis à 1 :

  MD11120 $MN_LUD_EXTENDED_SCOPE = 1

  Lorsque PM11120 = 0, les variables utilisateur globales définies dans le programme principal sont uniquement disponibles dans le programme principal.

- Variables utilisateur globales (GUD)
  Les variables utilisateur globales (GUD) sont des variables globales d'une CN ou d'un canal, définies dans un bloc de données (SGUD, MGUD, UGUD, GUD4 ... GUD9) et conservées même après un reset de fin de programme ou au démarrage suivant de la commande. Les variables GUD sont accessibles dans tous les programmes CN.

Les variables utilisateur doivent avoir été définies avant leur utilisation (lecture / écriture). Il convient de respecter les règles suivantes :

- Les variables GUD doivent être définies dans un fichier de définition, p. ex. _N_DEF_DIR/ _N_UGUD_DEF.
- Les variables PUD et LUD doivent être définies dans la section de définition d'un programme CN.
- La définition des données doit être effectuée dans un bloc spécifique.
- Un seul type de données peut être utilisé par définition de données.
- Plusieurs variables du même type de données peuvent être définies dans chaque définition de données.

Syntaxe

LUD et PUD

```
DEF <type> <unité physique> <valeurs limites> <nom>[<valeur 1>,
<valeur 2>, <valeur 3>]=<valeur d'initialisation>
```

GUD

```
DEF <domaine> <arrêt prétraitement blocs> <droits d'accès> <classe
de données> <type> <unité physique> <valeurs limites>
<nom>[<valeur_1>, <valeur_2>, <valeur_3>]=<valeur d'initialisation>
```

Signification

DEF :	Instruction de définition de variables utilisateur GUD, PUD, LUD	
<domaine> :	Domaine de validité, uniquement significatif pour GUD :	
	CN :	Variable utilisateur globale dans la CN
	CHAN :	Variable utilisateur globale dans le canal
<arrêt prétraitement blocs> :	Arrêt du prétraitement des blocs, uniquement significatif pour GUD (facultatif)	
	SYNR :	arrêt du prétraitement lors de la lecture
	SYNW :	arrêt du prétraitement lors de l'écriture
	SYNRW :	arrêt du prétraitement lors de la lecture / l'écriture
<droits d'accès> :	Niveau de protection pour la lecture / l'écriture de variables GUD par programme CN ou OPI (facultatif)	
	APRP <niveau de protection> :	Lecture : Programme CN
	APWP <niveau de protection> :	Ecriture : Programme CN
	APRB <niveau de protection> :	Lecture : OPI
	APWB <niveau de protection> :	Ecriture : OPI
	<niveau de protection> :	Plage de valeurs : 0 ... 7
	Voir "Attribut : droits d'accès (APR, APW, APRP, APWP, APRB, APWB) (Page 420)"	
<classe de données> :	Attribution d'une classe de données (SINUMERIK 828D uniquement !)	
	DCM :	Classe de données M (= Manufacturer)
	DCI :	Classe de données I (= Individual)
	DCU :	Classe de données U (= User)
<type> :	Type de données :	
	INT :	valeur entière avec signe
	REAL :	Nombre réel (LONG REAL selon IEEE)
	BOOL :	Valeur booléenne TRUE (1) / FALSE (0)
	CHAR :	Caractère ASCII
	STRING[<Longueur maximale>] :	Chaîne de caractères de longueur définie
	AXIS :	Descripteur d'axe/de broche
	FRAME :	Indications géométriques pour une transformation statique de coordonnées
	Voir "Types de données (Page 432)"	

<unité physique> :	Unité physique (facultative)	
	PHU <unité> :	Unité physique
	Voir "Attribut : unité physique (PHU) (Page 417)"	
<valeurs limites> :	Valeur limite inférieure/supérieure (facultative)	
	LLI <valeur limite> :	Valeur limite inférieure (lower limit)
	ULI <valeur limite> :	Valeur limite supérieure (upper limit)
	Voir "Attribut : valeurs limites (LLI, ULI) (Page 416)"	
<nom> :	Nom de la variable **Remarque** • 31 caractères au maximum • Les deux premiers caractères doivent être une lettre et/ou un caractère de soulignement. • Le caractère "$" est réservé aux variables système et ne doit pas être utilisé.	
[<valeur_1>, <valeur_2>, <valeur_3>] :	Indication de la dimension des tableaux pour des variables de tableau comportant 1 à 3 dimensions au maximum (facultative) Pour l'initialisation de variables de tableau, voir "Définition et initialisation de variables de tableau (DEF, SET, REP) (Page 426)"	
<valeur d'initialisation> :	Valeur d'initialisation (facultative) voir "Attribut : Valeur d'initialisation (Page 413)" Pour l'initialisation de variables de tableau, voir "Définition et initialisation de variables de tableau (DEF, SET, REP) (Page 426)"	

Exemples

Exemple 1 : définitions de variables utilisateur dans le bloc de données pour constructeur de machines

```
Code de programme                                                    Commentaire
%_N_MGUD_DEF                                                         ; Bloc GUD : constructeur de machines
$PATH=/_N_DEF_DIR
DEF CHAN REAL PHU 24 LLI 0 ULI 10 STROM_1, STROM_2
; Description
; Définition de deux variables GUD : STROM_1, STROM_2
; Domaine de validité : pour l'ensemble du canal
; Type de données : REAL
; Arrêt du prétraitement des blocs : non programmé => valeur par défaut = pas d'arrêt du prétraitement
des blocs
; Unité phys. : 24 = [A]
; Valeurs limites : inférieure = 0,0, supérieure = 10,0
; Droits d'accès : non programmés => valeur par défaut = 7 = position 0 du commutateur à clé
; Valeur d'initialisation : non programmée => valeur par défaut = 0,0

DEF NCK REAL PHU 13 LLI 10 APWP 3 APRP 3 APWB 0 APRB 2 ZEIT_1=12, ZEIT_2=45
```

Code de programme	Commentaire

```
; Description
; Définition de deux variables GUD : ZEIT_1, ZEIT_2
; Domaine de validité : global dans la CN
; Type de données : REAL
; Arrêt du prétraitement des blocs : non programmé => valeur par défaut = pas d'arrêt du prétraitement
des blocs
; Unité phys. : 13 = [s]
; Valeurs limites : inférieure = 10,0, supérieure = non programmée => limite du domaine de définition
; Droits d'accès :
; Programme CN : écriture/lecture = 3 = utilisateur
; BTSS: écriture = 0 = Siemens, lecture = 3 = utilisateur
; Valeur d'initialisation : ZEIT_1 = 12,0, ZEIT_2 = 45,0

DEF NCK APWP 3 APRP 3 APWB 0 APRB 3 STRING[5] GUD5_NAME = "COUNTER"
; Description
; Définition d'une variable GUD : GUD5_NAME
; Domaine de validité : global dans la CN
; Type de données : STRING, 5 caractères au maximum
; Arrêt du prétraitement des blocs : non programmé => valeur par défaut = pas d'arrêt du prétraitement
des blocs
; Unité phys. : non programmée => valeur par défaut = 0 = pas d'unité physique
; Valeurs limites : non programmées => limites du domaine de définition : inférieure = 0, supérieure =
255
; Droits d'accès :
; Programme CN : écriture/lecture = 3 = utilisateur
; BTSS: écriture = 0 = Siemens, lecture = 3 = utilisateur
; Valeur d'initialisation : "COUNTER"
M30
```

Exemple 2 : variables utilisateur globales et locales du programme (PUD/LUD)

Code de programme	Commentaire
PROC MAIN	; Programme principal
DEF INT VAR1	; Définition PUD
...	
SUB2	; Appel de sous-programme
...	
M30	

Code de programme	Commentaire
PROC SUB2	; Sous-programme SUB2
DEF INT VAR2	; Définition LUD
...	
IF (VAR1==1)	; Lecture PUD
VAR1=VAR1+1	; Lecture et écriture PUD

Code de programme	Commentaire
VAR2=1	; Ecriture LUD
ENDIF	
SUB3	; Appel de sous-programme
...	
M17	

Code de programme	Commentaire
PROC SUB3	; Sous-programme SUB3
...	
IF (VAR1==1)	; Lecture PUD
VAR1=VAR1+1	; Lecture et écriture PUD
VAR2=1	; Erreur : LUD de SUB2 non connu
ENDIF	
...	
M17	

Exemple 3 : définition et utilisation de variables utilisateur du type de données AXIS

Code de programme	Commentaire
DEF AXIS ABSCISSE	; 1er axe géométrique
DEF AXIS BROCHE	; Broche
...	
IF ISAXIS(1) == FALSE GOTOF SUITE	
ABSCISSE = $P_AXN1	
SUITE :	
...	
BROCHE=(S1)	; 1ère Broche
OVRA[BROCHE]=80	; Correction de vitesse de rotation de broche = 80 %
BROCHE=(S3)	; 3ème Broche

Autres conditions à prendre en compte

Variables utilisateur globales (GUD)

Les paramètres machines suivants doivent être pris en compte dans le cadre de la définition de variables utilisateur globales (GUD) :

N°	Descripteur : $MN_	Signification
11140	GUD_AREA_ SAVE_TAB	Sauvegarde supplémentaire pour blocs GUD
18118 [1)]	MM_NUM_GUD_MODULES	Nombre de fichiers GUD dans le système de fichiers actif
18120 [1)]	MM_NUM_GUD_NAMES_NCK	Nombre de noms GUD globaux
18130 [1)]	MM_NUM_GUD_NAMES_CHAN	Nombre de noms GUD spécifiques à un canal
18150 [1)]	MM_GUD_VALUES_MEM	Espace mémoire pour valeurs GUD globales

N°	Descripteur : $MN_	Signification
18660 [1)]	MM_NUM_SYNACT_GUD_REAL	Nombre de variables GUD configurables du type de données REAL
18661 [1)]	MM_NUM_SYNACT_GUD_INT	Nombre de variables GUD configurables du type de données INT
18662 [1)]	MM_NUM_SYNACT_GUD_BOOL	Nombre de variables GUD configurables du type de données BOOL
18663 [1)]	MM_NUM_SYNACT_GUD_AXIS	Nombre de variables GUD configurables du type de données AXIS
18664 [1)]	MM_NUM_SYNACT_GUD_CHAR	Nombre de variables GUD configurables du type de données CHAR
18665 [1)]	MM_NUM_SYNACT_GUD_STRING	Nombre de variables GUD configurables du type de données STRING

[1)] PM lisible pour SINUMERIK 828D uniquement !

Utilisation dans tous les canaux d'une variable utilisateur globale dans la CN du type de données AXIS

Une variable utilisateur globale de la NC du type de données `AXIS`, initialisée au moyen d'un descripteur lors de la définition dans le bloc de données, peut uniquement être utilisée dans différents canaux de la CN, si l'axe possède le même numéro de canal dans l'ensemble des canaux.

Si tel n'est pas le cas, la variable doit être chargée au début du programme CN, ou la fonction AXNAME(...) (voir "Fonctions d'axe (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Page 905)") doit être utilisée, comme dans l'exemple suivant.

```
Code de programme                          Commentaire
DEF NCK STRING[5] ACHSE="X"                ; Définition dans le bloc de données
...
N100 AX[AXNAME(AXE)]=111 G00               ; Utilisation dans le programme CN
```

3.1.1.6 Redéfinition de données système, données utilisateur et instructions de langage CN (REDEF)

L'instruction `REDEF` permet de modifier les attributs de données système, données utilisateur et instructions de langage CN. La condition de base nécessaire pour une redéfinition est qu'elle soit exécutée après la définition correspondante.

Lors d'une redéfinition, il n'est pas possible de modifier simultanément plusieurs attributs. Une instruction `REDEF` spécifique doit être programmée pour chaque attribut à modifier.

Dans le cas où plusieurs modifications d'attributs concurrentes sont programmées, la modification active est toujours la dernière.

Réinitialisation des valeurs d'attribut

Les attributs pour les droits d'accès et l'instant d'initialisation modifiés avec `REDEF` peuvent être réinitialisés à leur valeur par défaut par une nouvelle programmation de `REDEF`, suivie du nom de la variable ou de l'instruction de langage CN :

- Droits d'accès : Niveau de protection 7
- Instant d'initialisation : pas d'initialisation ou conservation de la valeur actuelle

Attributs redéfinissables

Voir "Vue d'ensemble des attributs définissables et redéfinissables (Page 424)".

Variables utilisateur locales (PUD / LUD)

Aucune redéfinition n'est autorisée pour des variables utilisateur locales (PUD / LUD).

Syntaxe

```
REDEF <nom> <arrêt prétraitement blocs>
REDEF <nom> <unité physique>
REDEF <nom> <valeurs limites>
REDEF <nom> <droits d'accès>
REDEF <nom> <instant d'initialisation>
REDEF <nom> <instant d'initialisation> <valeur d'initialisation>
REDEF <nom> <classe de données>
REDEF <nom>
```

Signification

`REDEF` :	Instruction pour la redéfinition d'un attribut précis ou pour la réinitialisation des attributs "Droits d'accès" et/ou "Instant d'initialisation" de variables système, de variables utilisateur et d'instructions de langage CN	
`<Nom>` :	Nom d'une variable déjà définie ou d'une instruction de langage CN	
`<arrêt prétraitement blocs>` :	Arrêt du prétraitement des blocs	
	`SYNR` :	arrêt du prétraitement lors de la lecture
	`SYNW` :	arrêt du prétraitement lors de l'écriture
	`SYNRW` :	arrêt du prétraitement lors de la lecture / l'écriture

<unité physique> :	Unité physique	
	PHU <unité> :	Unité physique
	Voir "Attribut : unité physique (PHU) (Page 417)". **Remarque** Non redéfinissable pour : • Variables système • Données utilisateur globales (GUD) des types de données : BOOL, AXIS, STRING, FRAME	
<valeurs limites> :	Valeur limite inférieure/supérieure	
	LLI <valeur limite> :	Valeur limite inférieure (lower limit)
	ULI <valeur limite> :	Valeur limite supérieure (upper limit)
	Voir "Attribut : valeurs limites (LLI, ULI) (Page 416)". **Remarque** Non redéfinissable pour : • Variables système • Données utilisateur globales (GUD) des types de données : BOOL, AXIS, STRING, FRAME	
<droits d'accès> :	Droits d'accès en lecture / écriture au moyen du programme pièce ou de BTSS	
	APX <niveau de protection> :	Exécuter : Elément de langage CN
	APRP <niveau de protection> :	Lecture : programme pièce
	APWP <niveau de protection> :	Ecriture : programme pièce
	APRB <niveau de protection> :	Lecture : OPI
	APWB <niveau de protection> :	Ecriture : OPI
	<niveau de protection> :	Plage de valeurs : 0 ... 7
	Voir "Attribut : droits d'accès (APR, APW, APRP, APWP, APRB, APWB) (Page 420)".	
<instant d'initialisation> :	Instant de réinitialisation de la variable	
	INIPO :	Power On
	INIRE :	Fin du programme principal, Reset CN ou Power On
	INICF :	NEWCONF ou fin du programme principal, Reset CN ou Power On
	PRLOC :	Fin du programme principal, Reset CN après une modification locale ou Power On
	Voir "Attribut : Valeur d'initialisation (Page 413)".	

<valeur d'initialisation> :	Valeur d'initialisation Pour redéfinir la valeur d'initialisation, il convient également de toujours spécifier un instant d'initialisation (voir <instant d'initialisation>). Voir "Attribut : Valeur d'initialisation (Page 413)". Pour l'initialisation de variables de tableau, voir "Définition et initialisation de variables de tableau (DEF, SET, REP) (Page 426)". **Remarque** Non redéfinissable pour les variables système, sauf les données de réglage.	
<classe de données> :	Attribution d'une classe de données (SINUMERIK 828D uniquement !)	
	DCM :	Classe de données M (= Manufacturer)
	DCI :	Classe de données I (= Individual)
	DCU :	Classe de données U (= User)

Exemple

Redéfinitions de la variable système $TC_DPCx dans le bloc de données pour le constructeur de machines

Code de programme

```
%_N_MGUD_DEF                                ; Bloc GUD : constructeur de machines
N100 REDEF $TC_DPC1 APWB 2 APWP 3
N200 REDEF $TC_DPC2 PHU 21
N300 REDEF $TC_DPC3 LLI 0 ULI 200
N400 REDEF $TC_DPC4 INIPO (100, 101, 102, 103)
N800 REDEF $TC_DPC1
N900 REDEF $TC_DPC4
M30
```

Concernant N100 : Droit d'accès en écriture : BTSS = niveau de protection 2, programme pièce = niveau de protection 3

Concernant N200 : Unité physique [%]

Concernant N300 : inférieure = 0, supérieure = 200

Concernant N400 : La variable de tableau est initialisée avec les quatre valeurs lors d'un PowerOn.

Concernant N800 / N900 Réinitialisation des valeurs d'attribut "Droits d'accès" et/ou "Instant d'initialisation"

Remarque

Utilisation des fichiers ACCESS

Lorsque des fichiers ACCESS sont utilisés, la redéfinition des droits d'accès doit être transposée de _N_MGUD_DEF en _N_MACCESS_DEF.

Autres conditions à prendre en compte

Granularité

Une redéfinition concerne toujours la variable complète, caractérisée de façon univoque par son nom. Il n'est pas possible, p. ex. pour des variables de tableau, d'attribuer différentes valeurs d'attribut à des éléments individuels de tableau.

3.1.1.7 Attribut : Valeur d'initialisation

Définition (DEF) de variables utilisateur

Lors de la définition, une valeur d'initialisation peut être prédéfinie pour les variables utilisateur suivantes :

- Variables utilisateur globales (GUD)
- Variables utilisateur globales d'un programme (PUD)
- Variables utilisateur locales (LUD)

Redéfinition (REDEF) de variables système et de variables utilisateur

Lors de la redéfinition, une valeur d'initialisation peut être prédéfinie pour les variables suivantes :

- Données système
  - Données de réglage
- Données utilisateur
  - Paramètres R
  - Variable à action synchrone ($AC_MARKER, $AC_PARAM, $AC_TIMER)
  - Variable GUD à action synchrone (SYG_xy[ ], avec x=R, I, B, A, C, S et y=S, M, U, 4, ..., 9)
  - Paramètre EPS
  - Données d'outil OEM
  - Données de magasin OEM
  - Variables utilisateur globales (GUD)

Instant de réinitialisation

Lors de la redéfinition, il est possible de spécifier l'instant auquel la variable est réinitialisée, c'est-à-dire doit reprendre sa valeur d'initialisation :

- `INIPO` (Power On)
  La variable est réinitialisée lors d'un Power On.
- `INIRE` (Reset)
  La variable est réinitialisée lors d'un Reset CN, d'un Reset GMFC, à la fin du programme pièce (`M02`/`M30`) ou lors d'un Power On.
- `INICF` (NEWCONF)
  La variable est réinitialisée lors de la fonction "Activation des paramètres machine" via IHM, via une instruction du programme pièce `NEWCONF` ou via un Reset CN, un Reset GMFC, ou à la fin du programme pièce (M02/M30) ou lors d'un Power On.
- `PRLOC` (modification locale du programme)
  La variable est uniquement réinitialisée en cas de Reset CN, Reset GMFC, à la fin du programme pièce (M02 / M30) si elle a été modifiée dans le cadre du programme pièce actuel. L'attribut `PRLOC` ne doit être utilisé que dans le contexte de données de réglage programmables (voir le tableau suivant).

Tableau 3-1 Données de réglage programmables

Numéro	Descripteur	Fonction G [1)]
42000	$SC_THREAD_START_ANGLE	SF
42010	$SC_THREAD_RAMP_DISP	DITS / DITE
42400	$SA_PUNCH_DWELLTIME	PDELAYON
42800	$SA_SPIND_ASSIGN_TAB	SETMS
43210	$SA_SPIND_MIN_VELO_G25	G25
43220	$SA_SPIND_MAX_VELO_G26	G26
43230	$SA_SPIND_MAX_VELO_LIMS	LIMS
43300	$SA_ASSIGN_FEED_PER_REV_SOURCE	FPRAON
43420	$SA_WORKAREA_LIMIT_PLUS	G26
43430	$SA_WORKAREA_LIMIT_MINUS	G25
43510	$SA_FIXED_STOP_TORQUE	FXST
43520	$SA_FIXED_STOP_WINDOW	FXSW
43700	$SA_OSCILL_REVERSE_POS1	OSP1
43710	$SA_OSCILL_REVERSE_POS2	OSP2
43720	$SA_OSCILL_DWELL_TIME1	OST1
43730	$SA_OSCILL_DWELL_TIME2	OST2
43740	$SA_OSCILL_VELO	FA
43750	$SA_OSCILL_NUM_SPARK_CYCLES	OSNSC
43760	$SA_OSCILL_END_POS	OSE
43770	$SA_OSCILL_CTRL_MASK	OSCTRL
43780	$SA_OSCILL_IS_ACTIVE	OS
43790	$SA_OSCILL_START_POS	OSB
1) Cette fonction G sollicite la donnée de réglage		

Autres conditions à prendre en compte

Valeur d'initialisation : Variables utilisateur globales (GUD)

- Dans le cas de la variable utilisateur globale (GUD) avec le domaine de validité `CN`, seul l'instant d'initialisation `INIPO` (Power On) peut être spécifié.
- Pour les variables utilisateur globales (GUD) avec le domaine de validité `CHAN`, il est possible de spécifier comme instant d'initialisation, outre `INIPO` (Power On), également `INIRE` (Reset) ou `INICF` (NEWCONF).
- Pour les variables utilisateur globales (GUD) avec le domaine de validité `CHAN` et l'instant d'initialisation `INIRE` (Reset) ou `INICF` (NEWCONF), les variables sont seulement réinitialisées pour Reset CN, Reset GMFC et "Activation des paramètres machine" dans les canaux dans lesquels les événements spécifiés ont été déclenchés.

Valeur d'initialisation : type de données FRAME

Pour les variables du type de données `FRAME`, la spécification d'une valeur d'initialisation n'est pas autorisée. Les variables du type de données `FRAME` sont implicitement toujours initialisées avec le frame par défaut.

Valeur d'initialisation : type de données CHAR

Pour les variables du type de données `CHAR`, il est possible de programmer à la place du code ASCII (0...255), le caractère ASCII correspondant représenté entre guillemets, p. ex. "A"

Valeur d'initialisation : type de données STRING

Pour les variables du type de données `STRING`, la chaîne de caractères doit figurer entre guillemets, p. ex. : ...= "MACHINE_1"

Valeur d'initialisation : type de données AXIS

Pour les variables du type de données `AXIS`, le descripteur d'axe doit figurer entre guillemets pour la notation étendue d'adresse, p. ex. : ...=(X3)

Valeur d'initialisation : Variable système

Pour les variables système, il n'est pas possible de prédéfinir par redéfinition des valeurs d'initialisation spécifiques à l'utilisateur. Les valeurs d'initialisation des variables système sont prédéfinies par le système. Il est par contre possible de modifier par redéfinition l'instant (`INIRE`, `INICF`) auquel la variable système est réinitialisée.

Valeur d'initialisation implicite : type de données AXIS

Pour les variables du type de données `AXIS`, la valeur d'initialisation implicite suivante est utilisée :

- Données système : "premier axe géométrique"
- GUD d'action synchrone (désignation : SYG_A*), PUD, LUD :
  descripteur d'axe à partir du paramètre machine : PM20082
  $MC_AXCONF_CHANAX_DEFAULT_NAME

Valeur d'initialisation implicite : données d'outil et de magasin

Pour les données d'outil et de magasin, il est possible de spécifier des valeurs d'initialisation à partir du paramètre machine suivant : PM17520 $MN_TOOL_DEFAULT_DATA_MASK

Remarque

Synchronisation

La synchronisation d'événements déclenchant la réinitialisation d'une variable globale avec lecture de cette variable à un autre endroit relève totalement de la responsabilité de l'utilisateur / du constructeur de machine.

Voir aussi

Variables (Page 395)

3.1.1.8 Attribut : valeurs limites (LLI, ULI)

Une valeur limite supérieure et inférieure de la plage de définition peut uniquement être prédéfinie pour les types de données suivants :

- INT
- REAL
- CHAR

Définition (DEF) de variables utilisateur : valeurs limites et valeurs d'initialisation implicites

Si aucune valeur d'initialisation explicite n'est définie lors de la déclaration d'une variable utilisateur de l'un des types de données précités, la valeur d'initialisation implicite du type de données est affectée à la variable :

- INT : 0
- REAL : 0.0
- CHAR : 0

Si la valeur d'initialisation implicite se situe en dehors de la plage de définition déterminée par les valeurs limites programmées, la variable est initialisée avec la valeur limite la plus proche de la valeur d'initialisation implicite :

- Valeur d'initialisation implicite < valeur limite inférieure (LLI) ⇒
  valeur d'initialisation = valeur limite inférieure
- Valeur d'initialisation implicite > valeur limite supérieure (ULI) ⇒
  valeur d'initialisation = valeur limite supérieure

Exemples :

Code de programme	**Commentaire**

`DEF REAL GUD1`	`; Valeur limite inférieure = limite de la plage de définition`
	`; Valeur limite supérieure = limite de la plage de définition`
	`; Aucune valeur d'initialisation programmée`
	`; => valeur d'initialisation implicite = 0.0`

`DEF REAL LLI 5.0 GUD2`	`; Valeur limite inférieure = 5.0`
	`; Valeur limite supérieure = limite de la plage de définition`
	`; => Valeur d'initialisation = 5.0`

`DEF REAL ULI –5 GUD3`	`; Valeur limite inférieure = limite de la plage de définition`
	`; Valeur limite supérieure = -5.0`
	`; => Valeur d'initialisation = -5,0`

Redéfinition (REDEF) de variables utilisateur : valeurs limites et valeurs réelles actuelles

Si lors de la redéfinition des valeurs limites d'une variable utilisateur, celles-ci sont modifiées de telle sorte que la valeur réelle actuelle se situe en dehors de la nouvelle plage de définition, une alarme est générée et les valeurs limites ne sont pas appliquées.

Remarque

Redéfinition (REDEF) de variables utilisateur

Lors de la redéfinition des valeurs limites d'une variable utilisateur, il faut veiller à la modification cohérente des valeurs suivantes :

- Valeurs limites
- Valeur réelle
- Valeur d'initialisation lors de la redéfinition et de la réinitialisation automatique sur la base de INIPO, INIRE ou INICF

Voir aussi

Variables (Page 395)

3.1.1.9 Attribut : unité physique (PHU)

Une unité physique peut uniquement être prédéfinie pour des variables possédant l'un des types de données suivants :

- INT
- REAL

Unités physiques programmables (PHU)

L'unité physique est spécifiée en tant que nombre à virgule fixe : `PHU <unité>`

Les unités physiques suivantes peuvent être programmées :

<Unité>	Signification	Unité physique
0	Pas d'unité physique	-
1	Position linéaire ou angulaire[1)2)]	[mm], [inch], [degré]
2	Position linéaire[2)]	[mm], [inch]
3	Position angulaire	[degré]
4	Vitesse linéaire ou angulaire[1)2)]	[mm/min], [inch/min], [tr/min]
5	Vitesse linéaire[2)]	[mm/min]
6	Vitesse angulaire	[tr/min]
7	Accélération linéaire ou angulaire[1)2)]	[m/s^2], [$inch/s^2$], [tr/s^2]
8	Accélération linéaire[2)]	[m/s^2], [$inch/s^2$]
9	Accélération angulaire	[tr/s^2]
10	A-coup linéaire ou angulaire[1)2)]	[m/s^3], [$inch/s^3$], [tr/s^3]
11	A-coup linéaire[2)]	[m/s^3], [$inch/s^3$]
12	A-coup angulaire	[tr/s^3]
13	Temporisation	[s]
14	Gain du régulateur de position	[16.667/s]
15	Avance par tour [2)]	[mm/tr], [inch/tr]
16	Compensation de température [1)2)]	[mm], [inch]
18	Force	[N]
19	Masse	[kg]
20	Moment d'inertie[3)]	[kgm^2]
21	Pour cent	[%]
22	Fréquence	[Hz]
23	Tension	[V]
24	Courant	[A]
25	Température	[°C]
26	Angle	[degré]
27	KV	[1000/min]
28	Position linéaire ou angulaire[3)]	[mm], [inch], [degré]
29	Vitesse de coupe [2)]	[m/min], [pied/min]
30	Vitesse circonférentielle[2)]	[m/s], [pied/s]
31	Résistance	[Ohm]
32	Inductance	[mH]
33	Couple [3)]	[Nm]
34	Constante du couple [3)]	[Nm/A]
35	Gain du régulateur de courant	[V/A]
36	Gain de vitesse de rotation [3)]	[Nm/(rad*s)]
37	Vitesse de rotation	[tr/min]
42	Puissance	[kW]

<Unité>	Signification	Unité physique
43	Courant faible	[µA]
46	Couple faible 3)	[µNm]
48	Pour mille	-
49	-	[Hz/s]
65	Débit	[l/min]
66	pression	[bar]
67	Volumes 3)	[cm^3]
68	Gain du système3)	[mm/(V*min)]
69	Gain du système pour régulateur de couple	[N/V]
155	Pas de vis 3)	[mm/tr], [inch/tr]
156	Modification du pas de vis 3)	[mm/tr / tr], [inch/tr / tr]
1) L'unité physique dépend du type d'axe : axe linéaire ou rotatif		
2) Commutation du système de coordonnées G70/G71(inch/métrique) Après une commutation du système de base (PM10240 $MN_SCALING_SYSTEM_IS_METRIC) avec G70/G71, **aucune** conversion des valeurs (valeur réelle, valeur par défaut et valeurs limites) n'est effectuée lors d'accès d'écriture/de lecture à des variables système et utilisateur relatives à des longueurs G700/G710(inch/métrique) Après une commutation du système de base (PM10240 $MN_SCALING_SYSTEM_IS_METRIC) avec G700/G710, **une** conversion des valeurs (valeur réelle, valeur par défaut et valeurs limites) est effectuée lors d'accès d'écriture/de lecture à des variables système et utilisateur relatives à des longueurs		
3) La variable n'est **pas** convertie automatiquement dans le système de coordonnées actuel de la CN (en pouces / métrique). La conversion relève de l'entière responsabilité de l'utilisateur / constructeur de machine.		

Remarque

Débordement de niveau dû aux conversions de format

Le format de stockage interne de l'ensemble des variables utilisateur (GUD / PUD / LUD) exprimées en unités physiques de longueur est métrique. Une utilisation excessive de telles variables lors de l'exécution de la NCK, par ex. durant des actions synchrones, risquerait de provoquer un dépassement de la durée de calcul du niveau de l'interpolateur lors d'un changement de système de coordonnées, conduisant ainsi à la génération de l'alarme 4240.

Remarque

Compatibilité d'unités

La compatibilité des unités concernées n'est pas vérifiée lors de l'utilisation de variables (affectation, comparaison, calcul, etc.). Une éventuelle conversion requise relève exclusivement de la responsabilité de l'utilisateur / constructeur de machine.

Voir aussi

Variables (Page 395)

3.1.1.10 Attribut : droits d'accès (APR, APW, APRP, APWP, APRB, APWB)

Désignation

La désignation des attributs d'accès AP... se compose comme suit :

1. A : **A**ccess
2. P : **P**rotection
3. R / W : **R**ead / **W**rite
4. P / B : **P**rogram / **B**TSS

Droits d'accès / niveaux de protection

Les niveaux de protection suivants, à indiquer lors de la programmation, correspondent aux différents droits d'accès :

Droit d'accès	Niveau de protection
Mot de passe système	0
Mot de passe constructeur de machines	1
Mot de passe maintenance	2
Mot de passe utilisateur	3
Commutateur à clé, position 3	4
Commutateur à clé, position 2	5
Commutateur à clé, position 1	6
Commutateur à clé, position 0	7

Définition (DEF) des données utilisateur

Il est possible de définir des droits d'accès (APR... / APW...) pour les données suivantes :

- Données utilisateur globales (GUD)

Redéfinition (REDEF) des données système et utilisateur

Il est possible de redéfinir des droits d'accès (APR... / APW...) pour les données suivantes :

- Données système
  - Paramètres machine

    Remarque

    Redéfinition des droits de lecture des paramètres machine

    Le niveau de protection pour la lecture des paramètres machine ne peut être réglé simultanément pour le programme pièce et BTSS qu'à l'aide du mot clé APR.

    Les mots clé APRP et APRB ne sont pas pris en charge pour la redéfinition des droits de lecture et génèrent l'alarme 12490 "Le droit d'accès APRP/APRB <niveau de protection> n'a pas été paramétré".

  - Données de réglage
  - Variable système
  - Données process
  - Données de magasin
  - Données d'outil
- Données utilisateur
  - Paramètres R
  - Variable à action synchrone ($AC_MARKER, $AC_PARAM, $AC_TIMER)
  - Variable GUD à action synchrone (SYG_xy[ ], avec x=R, I, B, A, C, S et y=S, M, U, 4, ..., 9)
  - Paramètre EPS
  - Données d'outil OEM
  - Données de magasin OEM
  - Variables utilisateur globales (GUD)

    Remarque

    Lors de la redéfinition, il est possible d'attribuer le droit d'accès à une variable quelconque comprise entre le niveau de protection le plus faible, à savoir 7, et son propre niveau de protection, par ex. 1 (constructeur de machines).

Redéfinition (REDEF) d'instructions de langage CN

Il est possible de redéfinir le droit d'accès ou d'exécution (APX) pour les instructions de langage CN suivantes :

- Fonctions G / fonctions préparatoires (Page 174)
- Fonctions prédéfinies (Page 1382)
- Appels de sous-programmes prédéfinis

- Instruction DO pour les actions synchrones
- Descripteurs de cycles dans un programme
  Le cycle doit être enregistré dans un répertoire de cycles et contenir une instruction PROC.

Droits d'accès concernant les programmes CN et les cycles (APRP, APWP)

Les différents droits d'accès à partir d'un programme CN ou d'un cycle ont les incidences suivantes :

- APRP 0 / APWP 0
  - L'exécution du programme CN requiert le mot de passe système.
  - Le cycle doit être enregistré dans le répertoire _N_CST_DIR (système).
  - Pour le répertoire _N_CST_DIR, le droit d'exécution doit être défini sur Système dans le paramètre machine MD11160 $MN_ACCESS_EXEC_CST.
- APRP 1 / APWP 1 ou APRP 2 / APWP 2
  - L'exécution du programme CN requiert le mot de passe du constructeur de machines ou du service de maintenance.
  - Le cycle doit être enregistré dans le répertoire _N_CMA_DIR (constructeur de machines) ou _N_CST_DIR.
  - Pour les répertoires _N_CMA_DIR ou _N_CST_DIR, les droits d'exécution doivent être définis au moins pour le constructeur de machines dans les paramètres machine MD11161 $MN_ACCESS_EXEC_CMA ou MD11160 $MN_ACCESS_EXEC_CST.
- APRP 3 / APWP 3
  - L'exécution du programme CN requiert le mot de passe de l'utilisateur.
  - Le cycle doit être enregistré dans le répertoire _N_CUS_DIR (utilisateur), _N_CMA_DIR ou _N_CST_DIR.
  - Pour les répertoires _N_CUS_DIR, _N_CMA_DIR ou _N_CST_DIR, les droits d'exécution doivent être définis au moins pour l'utilisateur dans les paramètres machine MD11162 $MN_ACCESS_EXEC_CUS, MD11161 $MN_ACCESS_EXEC_CMA ou MD11160 $MN_ACCESS_EXEC_CST.
- APRP 4...7 / APWP 4...7
  - L'exécution du programme CN requiert la position 3 ... 0 du commutateur à clé.
  - Le cycle doit être enregistré dans le répertoire _N_CUS_DIR, _N_CMA_DIR ou _N_CST_DIR.
  - Pour les répertoires _N_CUS_DIR, _N_CMA_DIR ou _N_CST_DIR, les droits d'exécution doivent être définis au moins pour la position correspondante du commutateur à clé dans les paramètres machine MD11162 $MN_ACCESS_EXEC_CUS, MD11161 $MN_ACCESS_EXEC_CMA ou MD11160 $MN_ACCESS_EXEC_CST.

Droits d'accès concernant BTSS (APRB, APWB)

Les droits d'accès (APRB, APWB) restreignent l'accès via BTSS aux variables système et variables utilisateur de la même manière pour tous les composants système (IHM, AP, ordinateurs externes, services EPS, etc.).

Remarque

Droits d'accès IHM locaux

Lors de la modification des droits d'accès de données système, il est nécessaire de veiller à la cohérence avec les droits d'accès définis par les mécanismes de l'IHM.

Attributs d'accès APR / APW

Pour des raisons de compatibilité, les attributs APR et APW sont représentés de manière implicite sur les attributs APRP / APRB et APWP / APWB :

- APR x ⇒ APRP x APRB x
- APW y ⇒ APWP y APWB y

Droits d'accès au moyen de fichiers ACCESS

Lorsque des fichiers ACCESS sont utilisés pour l'attribution de droits d'accès, les redéfinitions de ces droits pour les données système, données utilisateur et instructions de langage CN doivent être effectuées exclusivement dans les fichiers ACCESS. Les données utilisateur globales (GUD) constituent une exception. Pour celles-ci, la redéfinition doit toujours être effectuée dans les fichiers de définition *_DEF correspondants.

Pour une protection d'accès continue, les paramètres machine doivent être adaptés de façon cohérente pour les droits d'exécution et le droit d'accès des répertoires correspondants.

Il en résulte la marche à suivre de principe suivante :

1. Création des fichiers de définition requis :
   - _N_DEF_DIR/_N_SACCESS_DEF
   - _N_DEF_DIR/_N_MACCESS_DEF
   - _N_DEF_DIR/_N_UACCESS_DEF
2. Paramétrage du droit d'accès pour les fichiers de définition à la valeur requise pour la redéfinition :
   - MD11170 $MN_ACCESS_WRITE_SACCESS = <niveau de protection>
   - MD11171 $MN_ACCESS_WRITE_MACCESS = <niveau de protection>
   - MD11172 $MN_ACCESS_WRITE_UACCESS = <niveau de protection>

3. L'accès depuis des cycles à des éléments protégés nécessite l'adaptation des droits d'exécution et des droits d'écriture aux répertoires _N_CST_DIR, _N_CMA_DIR et _N_CST_DIR de ces cycles :
   Droits d'exécution
   - MD11160 $MN_ACCESS_EXEC_CST = <niveau de protection>
   - MD11161 $MN_ACCESS_EXEC_CMA = <niveau de protection>
   - MD11162 $MN_ACCESS_EXEC_CUS = <niveau de protection>

   Droits d'écriture
   - MD11165 $MN_ACCESS_WRITE_CST = <niveau de protection>
   - MD11166 $MN_ACCESS_WRITE_CMA = <niveau de protection>
   - MD11167 MN_ACCESS_WRITE_CUS = <niveau de protection>

   Le droit d'exécution doit être défini au moins au même niveau de protection que le niveau de protection le plus élevé de l'élément utilisé.
   Le droit d'écriture doit être défini au moins au même niveau de protection que le droit d'exécution.
4. Les droits d'écriture des répertoires des cycles pour l'IHM locale doivent être définis au même niveau de protection que les répertoires des cycles de la CN locale.

Appels de sous-programmes dans des fichiers ACCESS

Pour une meilleure structuration de la protection d'accès, il est possible d'appeler également des sous-programmes (extension SPF ou MPF) dans les fichiers ACCESS. Les droits d'accès du fichier ACCESS appelant sont alors attribués à ces sous-programmes.

Remarque

Seuls les droits d'accès peuvent être redéfinis dans les fichiers ACCESS. La programmation ou la redéfinition de tous les autres attributs doit toujours encore s'effectuer dans les fichiers de définition correspondants.

Voir aussi

Variables (Page 395)

3.1.1.11 Vue d'ensemble des attributs définissables et redéfinissables

Les tableaux suivants indiquent pour quels types de données il est possible de définir (`DEF`) et/ou redéfinir (`REDEF`) les différents attributs.

Données système

Type de donnée	Valeur initiale	Valeurs limites	Unité physique	Droits d'accès	Classe de données (828D uniquement)
Paramètres machine	---	---	---	REDEF	REDEF
Données de réglage	REDEF	---	---	REDEF	---
Données FRAME	---	---	---	REDEF	---
Données process	---	---	---	REDEF	---
Compensation d'erreur de pas de vis de transmission (EEC)	---	---	---	REDEF	---
Compensation de la flexion (CEC)	---	---	---	REDEF	---
Compensation des défauts aux transitions entre quadrants (QEC)	---	---	---	REDEF	---
Données de magasin	---	---	---	REDEF	---
Données d'outil	---	---	---	REDEF	---
Zones de protection	---	---	---	REDEF	---
Organes porte-outil orientables	---	---	---	REDEF	---
Chaînes cinématiques	---	---	---	REDEF	---
Zones de protection 3D	---	---	---	REDEF	---
Limitation de la zone de travail	---	---	---	REDEF	---

Données utilisateur

Type de donnée	Valeur initiale	Valeurs limites	Unité physique	Droits d'accès	Classe de données
Paramètre R	REDEF	REDEF	REDEF	REDEF	---
Variable à action synchrone ($AC_...)	REDEF	REDEF	REDEF	REDEF	---
GUD d'action synchrone (SYG_...)	REDEF	REDEF	REDEF	REDEF	---
Paramètre EPS	REDEF	REDEF	REDEF	REDEF	---
Données outil OEM	REDEF	REDEF	REDEF	REDEF	---
Données magasin OEM	REDEF	REDEF	REDEF	REDEF	---
Variables utilisateur globales (GUD)	DEF / REDEF	DEF	DEF	DEF / REDEF	DEF / REDEF
Variables utilisateur locales (PUD / LUD)	DEF	DEF	DEF	---	---

Voir aussi

Variables (Page 395)

3.1.1.12 Définition et initialisation de variables de tableau (DEF, SET, REP)

Une variable utilisateur peut être définie comme tableau (Array) de 1 à 3 dimensions au maximum :

- 1 dimension : `DEF <type de données> <nom de variable>[<n>]`
- 2 dimensions : `DEF <type de données> <nom de variable>[<n>,<m>]`
- 3 dimensions : `DEF <type de données> <nom de variable>[<n>,<m>,<o>]`

Remarque

Les variables utilisateur du type de données STRING peuvent être définies comme tableau à 2 dimensions au maximum.

Types de données

Les variables utilisateur peuvent être définie comme tableaux pour les types de données suivants : BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME

Affectation de valeur à des éléments de tableau

Les affectations de valeurs à des éléments de tableau peuvent être réalisées aux instants suivants :

- Lors de la définition du tableau (valeurs d'initialisation)
- Pendant l'exécution du programme

Les affectations de valeurs peuvent être réalisées par :

- Indication explicite d'un élément de tableau
- Indication explicite d'un élément de tableau comme élément de départ et indication d'une liste de valeurs (`SET`)
- Indication explicite d'un élément de tableau comme élément de départ et indication d'une valeur ainsi que de sa fréquence d'itération (`REP`)

Remarque

Il n'est pas possible d'affecter des valeurs d'initialisation à des variables utilisateur du type de données FRAME.

Syntaxe (`DEF`)

```
DEF <type de données> <nom de variable>[<n>,<m>,<o>]
DEF  STRING[<longueur de chaîne>] <nom de variable>[<n>,<m>]
```

Syntaxe (DEF...=SET...)

Utilisation d'une liste de valeurs :

- Lors de la définition :
  ```
  DEF <type de données> <nom de variable>[<n>,<m>,<o>] =
  SET(<valeur1>,<valeur2>,...)
  ```
  Equivaut à :
  ```
  DEF <type de données> <nom de variable>[<n>,<m>,<o>]=
  (<valeur1>,<valeur2>,...)
  ```

  Remarque

  Lors de l'initialisation au moyen d'une liste de valeurs, l'indication de `SET` est facultative.

- Lors d'une affectation de valeur :
  ```
  <Nom de variable>[<n>,<m>,<o>]=SET(<VALEUR1>,<valeur2>,...)
  ```

Syntaxe (DEF...=REP...)

Utilisation d'une valeur avec itération

- Lors de la définition :
  ```
  DEF <type de données> <nom de variable>[<n>,<m>,<o>]=REP(<Valeur>)
  DEF <type de données> <nom de variable>[<n>,<m>,<o>] =
  REP(<valeur>,<nombre d'éléments de tableau>)
  ```
- Lors d'une affectation de valeur :
  ```
  <Nom de variable>[<n>,<m>,<o>]=REP(<Valeur>)
  <Nom de variable>[<n>,<m>,<o>] = REP(<valeur>,<nombre d'éléments
  de tableau>)
  ```

Signification

`DEF` :	Instruction de définition des variables	
`<type de données>` :	Type de données de la variable	
	Plage de valeurs : • Pour les variables système : BOOL, CHAR, INT, REAL, STRING, AXIS • Pour les variables GUD et LUD : BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME	
`<longueur de chaîne>` :	Nombre maximal de caractères du type de données STRING	
`<nom de variable>` :	Nom de variable	
`[<n>,<m>,<o>]` :	Dimensions de tableaux ou indices de tableaux	
`<n>` :	Dimension ou indice pour un tableau à 1 dimension	
	Type :	INT (également AXIS pour les variables système)
	Plage de valeurs :	Dimension de tableau max. : 65535 Indice de tableau : 0 ≤ n ≤ 65534

<m> :	Dimension ou indice pour un tableau à 2 dimension	
	Type :	INT (également AXIS pour les variables système)
	Plage de valeurs :	Dimension de tableau max. : 65535 Indice de tableau : 0 ≤ m ≤ 65534
<o> :	Dimension ou indice pour un tableau à 3 dimension	
	Type :	INT (également AXIS pour les variables système)
	Plage de valeurs :	Dimension de tableau max. : 65535 Indice de tableau : 0 ≤ o ≤ 65534
SET :	Affectation de valeur au moyen de la liste de valeurs indiquée	
(<valeur1>,<valeur2>,...) :	Liste de valeurs	
REP :	Affectation de valeur au moyen de la <valeur> indiquée	
<valeur> :	Valeur à écrire dans les éléments de tableau lors de l'initialisation avec REP	
<nombre d'éléments de tableau> :	Nombre d'éléments de tableau à décrire avec la <valeur> indiquée. Pour les autres éléments de tableau, s'applique selon l'instant : • Initialisation lors d la définition du tableau : → Les éléments de tableau restant sont décrits avec des zéros • Affectation pendant l'exécution du programme : → Les valeurs courantes des éléments de tableau demeurent inchangées. Si le paramètre n'est pas programmé, tous les éléments de tableau sont décrits avec <valeur>. Si le paramètre est égal à zéro, s'applique selon l'instant : • Initialisation lors d la définition du tableau : → Tous les éléments sont mis à zéro par le système. • Affectation pendant l'exécution du programme : → Les valeurs courantes des éléments de tableau demeurent inchangées.	

Indice de tableau

L'ordre implicite des éléments de tableau, p. ex. lors d'une affectation de valeur via SET ou REP, est déterminé par itération de l'indice de tableau de droite à gauche.

Exemple : Initialisation d'un tableau à 3 dimensions comportant 24 éléments de tableau :

```
DEF INT FELD[2,3,4] = REP(1,24)
  FELD[0,0,0] = 1      1. élément de tableau
  FELD[0,0,1] = 1      2. élément de tableau
  FELD[0,0,2] = 1      3. élément de tableau
  FELD[0,0,3] = 1      4. élément de tableau
  ...
  FELD[0,1,0] = 1      5. élément de tableau
  FELD[0,1,1] = 1      6. élément de tableau
  ...
```

```
  FELD[0,2,3] = 1      12. élément de tableau
  FELD[1,0,0] = 1      13. élément de tableau
  FELD[1,0,1] = 1      14. élément de tableau
  ...
  FELD[1,2,3] = 1      24. élément de tableau
```

Correspond à :

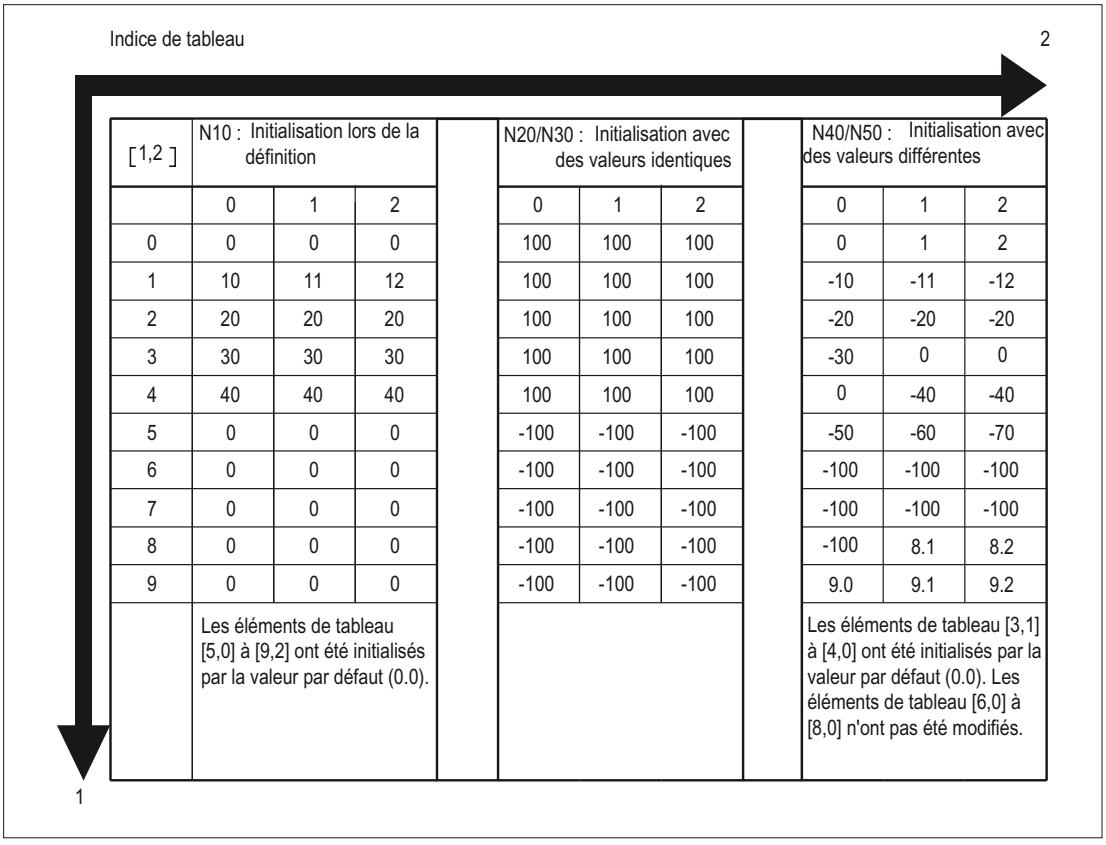
```
FOR n=0 TO 1
  FOR m=0 TO 2
     FOR o=0 TO 3
       FELD[n,m,o] = 1
     ENDFOR
  ENDFOR
ENDFOR
```

Exemple : Initialisation de tableaux de variables complets

Pour l'affectation courante, voir la figure.

Code de programme

```
N10 DEF REAL TABLEAU1[10,3]=SET(0,0,0,10,11,12,20,20,20,30,30,30,40,40,40,)
N20 TABLEAU1[0,0]=REP(100)
N30 TABLEAU1[5,0]=REP(-100)
N40 TABLEAU1[0,0]=SET(0,1,2,-10,-11,-12,-20,-20,-20,-30, , , ,-40,-40,-50,-60,-70)
N50 TABLEAU1[8,1]=SET(8.1,8.2,9.0,9.1,9.2)
```



[1,2]	N10 : Initialisation lors de la définition		
	0	1	2
0	0	0	0
1	10	11	12
2	20	20	20
3	30	30	30
4	40	40	40
5	0	0	0
6	0	0	0
7	0	0	0
8	0	0	0
9	0	0	0
	Les éléments de tableau [5,0] à [9,2] ont été initialisés par la valeur par défaut (0.0).		

N20/N30 : Initialisation avec des valeurs identiques		
0	1	2
100	100	100
100	100	100
100	100	100
100	100	100
100	100	100
-100	-100	-100
-100	-100	-100
-100	-100	-100
-100	-100	-100
-100	-100	-100

N40/N50 : Initialisation avec des valeurs différentes		
0	1	2
0	1	2
-10	-11	-12
-20	-20	-20
-30	0	0
0	-40	-40
-50	-60	-70
-100	-100	-100
-100	-100	-100
-100	8.1	8.2
9.0	9.1	9.2
Les éléments de tableau [3,1] à [4,0] ont été initialisés par la valeur par défaut (0.0). Les éléments de tableau [6,0] à [8,0] n'ont pas été modifiés.		

Voir aussi

Définition et initialisation de variables de tableau (DEF, SET, REP) (Page 426)

Variables (Page 395)

Informations complémentaires (SET)

Initialisation lors de la définition

- Un nombre d'éléments de tableau égal au nombre d'éléments contenus dans la liste de valeurs est initialisé avec les valeurs de la liste, en commençant par le 1er élément de tableau.
- La valeur 0 est affectée aux éléments de tableau sans valeur explicite indiquée dans la liste de valeurs (lacunes dans la liste de valeurs).
- La liste de valeurs ne doit pas comporter de lacunes pour les variables relevant du type de données AXIS.
- Si la liste contient davantage de valeurs que ne sont définis d'éléments de tableau, une alarme est émise.

Affectation de valeur pendant l'exécution du programme

Les règles décrites ci-dessus dans le cadre de la définition s'appliquent à l'affectation de valeurs pendant l'exécution du programme. Il existe en outre les possibilités suivantes :

- Il est également possible d'utiliser des expressions comme éléments dans la liste de valeurs.
- L'affectation de valeurs débute par l'indice de tableau programmé. De cette façon, on peut affecter des valeurs à des tableaux partiels.

Exemple :

Code de programme	Commentaire
`DEF INT ARRAY[5,5]`	`; Définition de tableau`
`ARRAY[0,0]=SET(1,2,3,4,5)`	`; Affectation de valeurs aux 5 premiers éléments de tableau [0,0] - [0,4]`
`ARRAY[0,0]=SET(1,2, , ,5)`	`; Affectation de valeurs avec lacune sur les 5 premiers éléments de tableau [0,0] - [0,4], éléments de tableau [0,2] et [0,3] = 0`

`ARRAY[2,3]=SET(VARIABLE,4*5.6)`	`; Affectation de valeurs avec une variable et une expression à partir de l'indice de tableau [2,3] :`
	`[2,3] = VARIABLE`
	`[2,4] = 4 * 5.6 = 22.4`

Informations complémentaires (REP)

Initialisation lors de la définition

- L'ensemble des éléments de tableau ou le nombre d'entre eux spécifié de manière facultative sont initialisés avec la valeur indiquée (constante).
- Les variables relevant du type de données FRAME ne peuvent pas être initialisées.

Exemple :

Code de programme	Commentaire
`DEF REAL varName[10]=REP(3.5,4)`	`; Initialiser la définition de tableau et les éléments de tableau [0] à [3] avec la valeur 3,5`

Affectation de valeur pendant l'exécution du programme

Les règles décrites ci-dessus dans le cadre de la définition s'appliquent à l'affectation de valeurs pendant l'exécution du programme. Il existe en outre les possibilités suivantes :

- Il est également possible d'utiliser des expressions comme éléments dans la liste de valeurs.
- L'affectation de valeurs débute par l'indice de tableau programmé. De cette façon, on peut affecter des valeurs à des tableaux partiels.

Exemples :

Code de programme	Commentaire
`DEF REAL varName[10]`	`; Définition de tableau`
`varName[5]=REP(4.5,3)`	`; Éléments de tableau [5] à [7] = 4,5`
`R10=REP(2.4,3)`	`; Paramètres R, R10 à R12 = 2,4`

Code de programme	**Commentaire**
`DEF FRAME FRM[10]`	`; Définition de tableau`
`FRM[5]=REP(CTRANS(X,5))`	`; Éléments de tableau [5] à [9] = CTRANS(X,5)`

Voir aussi

Définition et initialisation de variables de tableau (DEF, SET, REP) (Page 426)

3.1.1.13 Types de données

Les types de données suivants sont disponibles dans la CN :

type de données	**Signification**	**Plage de valeurs**
INT	Valeur entière avec signe	-2147483648 ... +2147483647
REAL	Nombre réel (LONG REAL selon IEEE)	±(~2,2*10^{-308} ... ~1,8*10^{+308})
BOOL	Valeur booléenne TRUE (1) et FALSE (0)	1, 0
CHAR	Caractère ASCII	Code ASCII 0 ... 255
STRING	Chaîne de caractères de longueur définie	au maximum 200 caractères (caractères spéciaux exclus)
AXIS	Descripteur d'axe/de broche	Descripteur d'axe de canal
FRAME	Indications géométriques pour une transformation statique de coordonnées (décalage, rotation, mise à l'échelle, fonction miroir)	---

Conversion implicite du type de données

Les conversions suivantes de types de données sont possibles et réalisées de façon implicite pour les affectations et transmissions de paramètres :

de ↓/ vers →	**REAL**	**INT**	**BOOL**
REAL	x	o	&
INT	x	x	&
BOOL	x	x	x

x : Réalisable sans restrictions

o : Risque de perte de données par écrasement de la plage de valeurs ⇒ alarme ;
arrondissement : valeur après la virgule ≥ 0,5 ⇒ arrondissement à la valeur supérieure, valeur après la virgule < 0,5 ⇒ arrondissement à la valeur inférieure

&: valeur ≠ 0 ⇒ TRUE, valeur == 0 ⇒ FALSE

Voir aussi

Variables (Page 395)

3.1.1.14 Minimum, maximum et plage de variables (MINVAL, MAXVAL, BOUND)

Les instructions MINVAL et MAXVAL permettent de comparer entre elles les valeurs de deux variables. Le résultat fourni en retour la valeur la plus petite (pour MINVAL) ou la valeur la plus grande (pour MAXVAL).

L'instruction BOUND permet de vérifier si la valeur d'une variable à tester se trouve dans une plage de valeurs définie.

Syntaxe

```
<valeur la plus petite>=MINVAL(<variable1>,<variable2>)
<valeur la plus grande>=MAXVAL(<variable1>,<variable2>)
<valeur de retour>=<BOUND>(<minimum>,<maximum>,<variable à tester>)
```

Signification

MINVAL :	Détermine la **plus petite** parmi deux variables (<variable1>, <variable2>)
<valeur la plus petite> :	Variable résultante pour l'instruction MINVAL Est réglée à la valeur la plus petite de la variable.
MAXVAL :	Détermine la **plus grande** parmi deux variables (<variable1>, <variable2>)
<valeur la plus grande> :	Variable résultante pour l'instruction MAXVAL Est réglée à la valeur la plus grande de la variable.
BOUND :	Vérifie si une variable (<variable à tester>) se trouve dans la plage de valeurs définie.
<minimum> :	Variable définissant la valeur minimale de la plage de valeurs
<maximum> :	Variable définissant la valeur maximale de la plage de valeurs
<valeur de retour> :	Variable résultante pour l'instruction BOUND Si la valeur de la variable à tester se trouve dans la plage de valeurs définie, la variable résultante est réglée à la valeur de la variable à tester. Si la valeur de la variable à tester est supérieure à la valeur maximale, la variable résultante est réglée à la valeur maximale de la plage de définition. Si la valeur de la variable à tester est inférieure à la valeur minimale, la variable résultante est réglée à la valeur minimale de la plage de définition.

Remarque

MINVAL, MAXVAL et BOUND peuvent également être programmées dans actions synchrones.

Remarque

Comportement en cas d'égalité

En cas d'égalité de MINVAL/MAXVAL, cette valeur identique est fournie. Avec BOUND, la valeur de la variable à tester est de nouveau fournie.

Exemple

Code de programme	Commentaire
DEF REAL rVar1=10.5, rVar2=33.7, rVar3, rVar4,	rVar5, rValMin, rValMax, rRetVar
rValMin=MINVAL(rVar1,rVar2)	; rValMin est réglée à la valeur 10.5.
rValMax=MAXVAL(rVar1,rVar2)	; rValMax est réglée à la valeur 33.7.
rVar3=19.7	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 se trouve dans les limites, rRetVar est réglée à 19.7.
rVar3=1.8	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 se situe en-deçà de la limite minimale, rRetVar est réglée à 10.5.
rVar3=45.2	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 se situe au-dessus de la limite maximale, rRetVar est réglée à 33.7.

3.1.1.15 Vérification de l'existence d'une variable (ISVAR)

La fonction prédéfinie ISVAR permet de vérifier si une variable système ou une variable utilisateur (par ex. paramètre machine, donnée de réglage, variable système, variable générale telle que variable GUD) est connue dans la CN.

Variables

La variable à interroger a la structure suivante :

Variable sans dimension :	<Variable>
Variable à une dimension sans indice de tableau :	<Variable>[ ]
Variable à une dimension avec indice de tableau n :	<Variable>[<n>]
Variable à deux dimensions sans indice de tableau :	<Variable>[,]
Variable à deux dimensions avec indices de tableau n et m :	<Variable>[<n>,<m>]

Syntaxe

```
<Ergebnis>=ISVAR(<Variable>[<n>,<m>])
```

Signification

<Ergebnis> :	Valeur de retour		
	Type de données :	BOOL	
	Plage de valeurs :	1	Variable présente
		0	Variable inconnue
ISVAR :	Vérifie que la variable système/utilisateur est connue dans la CN.		

<Variable> :	Nom de la variable système/utilisateur	
	Type de données :	STRING
<n> :	Indice de tableau de la première dimension (**facultatif**)	
	Type de données :	INT
<m> :	Indice de tableau de la deuxième dimension (**facultatif**)	
	Type de données :	INT

Conformément au paramètre de transfert, les points suivants sont vérifiés :

- Le nom est-il connu ?
- La variable est-elle un tableau ?
- S'agit-il d'un tableau à une ou deux dimensions ?
- L'indice de tableau se trouve-t-il dans la plage autorisée ?

Si tous ces points sont vérifiés, l'état TRUE (1) est signalé en retour.

Si l'un des point n'est pas vérifié ou en présence d'une erreur de syntaxe, FALSE (0) est signalé.

Exemples

Code de programme	**Commentaire**
DEF INT VAR1	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VAR1")	; Dans ce cas, IS_VAR est TRUE.

Code de programme	**Commentaire**
DEF REAL VARARRAY[10,10]	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VARARRAY[,]")	; Dans ce cas, IS_VAR est TRUE ; c'est un tableau à deux dimensions.
N20 IS_VAR=ISVAR("VARARRAY")	; IS_VAR est TRUE ; la variable existe.
N30 IS_VAR=ISVAR("VARARRAY[8,11]")	; IS_VAR est FALSE ; l'indice de tableau est illicite.
N40 IS_VAR=ISVAR("VARARRAY[8,8")	; IS_VAR est FALSE ; il manque "]" (erreur de syntaxe).
N50 IS_VAR=ISVAR("VARARRAY[,8]")	; IS_VAR est TRUE ; l'indice de tableau est autorisé.
N60 IS_VAR=ISVAR("VARARRAY[8,]")	; IS_VAR est TRUE ; l'indice de tableau est autorisé.

Code de programme	**Commentaire**
DEF BOOL IS_VAR=FALSE	
N100 IS_VAR=ISVAR("$MC_GCODE_RESET_VALUES[1]"	; Le paramètre de transfert est un paramètre machine ; IS_VAR est TRUE.

Code de programme	Commentaire
`DEF BOOL IS_VAR=FALSE`	
`N10 IS_VAR=ISVAR("$P_EP")`	`; Dans ce cas, IS_VAR est TRUE.`
`N20 IS_VAR=ISVAR("$P_EP[X]")`	`; Dans ce cas, IS_VAR est TRUE.`

3.1.1.16 Lecture des valeurs d'attribut / du type de données (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP)

Les fonctions GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM et GETVARDFT permettent de lire les valeurs d'attribut des variables système ou utilisateur, et GETVARTYP permet d'en lire le type de données.

Lecture de l'unité physique

Syntaxe :
`<Résultat>=GETVARPHU(<Nom>)`

Signification :

`<Résultat>` :	Valeur numérique de l'unité physique		
	Type de données :	INT	
	Plage de valeurs :	voir le tableau dans "Attribut : unité physique (PHU) (Page 417)"	
		En cas d'erreur :	
		- 2	Le nom de variable spécifié n'est pas associé à un paramètre système ni à une variable utilisateur.
`GETVARPHU` :	Lecture de l'unité physique d'une variable système ou utilisateur		
`<Nom>` :	Nom de la variable système ou utilisateur		
	Type de données :	STRING	

Exemple :

Soit la variable GUD suivante définie dans la NC :

DEF CHAN REAL PHU 42 LLI 0 ULI 10000 electric

Code de programme	Commentaire
`DEF INT result=0`	
`result=GETVARPHU("electric")`	`; Détermine l'unité physique correspondant à la variable GUD`
`IF (result < 0) GOTOF error`	

Le résultat signalé en retour est la valeur 42 qui correspond à l'unité physique [kW].

Remarque

GETVARPHU permet par exemple de vérifier si, pour l'affectation de variables a = b, les deux variables possèdent l'unité physique attendue.

Lecture du droit d'accès :

Syntaxe :
`<Résultat>=GETVARAP(<Nom>,<Accès>)`

Signification :

`<Résultat>` :	Niveau de protection pour l'<accès> spécifié		
	Type de données :	INT	
	Plage de valeurs :	0 ... 7	Voir "Attribut : droits d'accès (APR, APW, APRP, APWP, APRB, APWB) (Page 420)".
		En cas d'erreur :	
		- 1	Pas d'accès en écriture (pertinent uniquement pour les types d'accès "WP" et "WB")
		- 2	Le nom de variable spécifié n'est pas associé à un paramètre système ni à une variable utilisateur.
		- 3	Valeur incorrecte pour le paramètre <Accès>
`GETVARAP` :	Lecture du droit d'accès à une variable système ou utilisateur		
`<Nom>` :	Nom de la variable système ou utilisateur		
	Type de données :	STRING	
`<Zugriff>` :	Type d'accès		
	Type de données :	STRING	
	Plage de valeurs :	`"RP"`	Lecture par le programme pièce
		`"WP"`	Ecriture par le programme pièce
		`"RB"`	Lecture par OPI
		`"WB"`	Ecriture par OPI

Exemple :

Code de programme	**Commentaire**
`DEF INT result=0`	
`result=GETVARAP("$TC_MAP8","WB")`	`; Détermine la protection de l'accès pour le paramètre système "position de magasin" pour la lecture par OPI.`
`IF (result < 0) GOTOF error`	

Le résultat signalé en retour est la valeur 7 qui correspond à la position 0 du commutateur à clé (= accès non protégé).

Remarque

GETVARAP permet par exemple de réaliser un programme de vérification qui contrôle les droits d'accès attendus par l'application.

Lecture des valeurs limites

Syntaxe :
`<État>=GETVARLIM(<Nom>,<Valeur limite>,<Résultat>)`

Signification :

`<État>` :	État de la fonction		
	Type de données :	INT	
	Plage de valeurs :	1	OK
		-1	pas de valeur limite définie (pour les variables de type AXIS, STRING, FRAME)
		-2	Le nom de variable spécifié n'est pas associé à un paramètre système ni à une variable utilisateur.
		-3	valeur incorrecte pour le paramètre <Valeur limite>
`GETVARLIM` :	Lecture de la valeur limite inférieure ou supérieure d'une variable système ou utilisateur		
`<Nom>` :	Nom de la variable système ou utilisateur		
	Type de données :	STRING	
`<Valeur limite>` :	Indique la valeur limite à lire		
	Type de données :	CHAR	
	Plage de valeurs :	`"L"`	Valeur limite inférieure
		`"U"`	Valeur limite supérieure
`<Résultat>` :	Valeur limite retournée		
	Type de données :	VAR REAL	

Exemple :

Code de programme	**Commentaire**
`DEF INT state=0`	
`DEF REAL result=0`	
`state=GETVARLIM("$MA_MAX_AX_VELO","L",result)`	`; Détermine la valeur limite inférieure du MD32000 $MA_MAX_AX_VELO.`
`IF (result < 0) GOTOF error`	

Lecture de la valeur de dimension

Syntaxe :
`<Résultat>=GETVARDIM(<Nom>,<Indice>)`

Signification :

`<Résultat>` :	Grandeur de la dimension spécifiée par le paramètre <Indice>		
	Type de données :	INT	
`GETVARDIM` :	Lecture de la grandeur de la 1ère, 2e ou 3e dimension du champ d'une variable système/utilisateur		
`<Nom>` :	Nom de la variable système ou utilisateur		
	Type de données :	STRING	
`<Indice>` :	Indice de tableau		
	Type de données :	INT	
	Plage de valeurs :	1 ... 3	
		1	Indice pour la 1ère dimension du champ
		2	Indice pour la 2e dimension du champ
		3	Indice pour la 3e dimension du champ

Exemple :

Code de programme	**Commentaire**
`N5 DEF REAL myReal[5,4]`	
`N10 R1=GETVARDIM("myReal",1)`	`; Détermine la grandeur de la 1ère dimension du champ.`
	`; Résultat : R1 = 5`
`N15 R2=GETVARDIM("myReal",2)`	`; Détermine la grandeur de la 2e dimension du champ.`
	`; Résultat : R2 = 4`

Lecture de la valeur par défaut

Syntaxe :
`<état>=GETVARDFT(<nom>,<résultat>[,<indice_1>,<indice_2>,<indice_3>])`

Signification :

<État> :	État de la fonction		
	Type de données :	INT	
	Plage de valeurs :	1	OK
		-1	Aucune valeur par défaut disponible (par ex. parce que la définition du type de données des variables de résultat ne correspond pas au type de données des variables système/utilisateur)
		-2	Le nom de variable spécifié n'est pas associé à un paramètre système ni à une variable utilisateur.
		-3	valeur incorrecte pour le paramètre <indice_1>, dimension inférieure à un (⇒ pas de champ, mais une variable scalaire)
		-4	valeur incorrecte pour le paramètre <indice_2>
		-5	valeur incorrecte pour le paramètre <indice_3>
GETVARDFT :	Lecture de la valeur par défaut d'une variable système ou utilisateur		
<Nom> :	Nom de la variable système ou utilisateur		
	Type de données :	STRING	
<Résultat> :	Valeur par défaut retournée		
	Type de données :	VAR REAL (pour la lecture de la valeur par défaut de variables de type INT, REAL, BOOL, AXIS)	
		VAR STRING (pour la lecture de la valeur par défaut de variables de type STRING et CHAR)	
		VAR FRAME (pour la lecture de la valeur par défaut de variables de type FRAME)	
<Indice_1> :	Indice pour la 1ère dimension du champ (en option)		
	Type de données :	INT	
	Non programmé = 0		
<Indice_2> :	Indice pour la 2e dimension du champ (en option)		
	Type de données :	INT	
	Non programmé = 0		
<Indice_3> :	Indice pour la 3e dimension du champ (en option)		
	Type de données :	INT	
	Non programmé = 0		

Exemple :

Code de programme	**Commentaire**
DEF INT state=0	
DEF REAL resultR=0	; Variable pour l'acquisition des valeurs par défaut de type INT, REAL, BOOL, AXIS.

```
Code de programme                                                    Commentaire
DEF FRAME resultF=0                                                  ; Variable pour l'acquisition des valeurs
                                                                     par défaut de type FRAME

IF (GETVARTYP("$MA_MAX_AX_VELO") <> 4) GOTOF error

state=GETVARDFT("$MA_MAX_AX_VELO",resultR, AXTOINT(X))               ; Détermine la valeur par défaut de l'axe
                                                                     "X".
IF (resultR < 0) GOTOF error

IF (GETVARTYP("$TC_TP8") <> 3) GOTOF error
state=GETVARDFT("$TC_TP8", resultR)

IF (GETVARTYP("$P_UBFR")  <> 7) GOTOF error
state=GETVARDFT("$P_UBFR", resultF )
```

Lecture du type de données

Syntaxe :
<Résultat>=GETVARTYP(<Nom>)

Signification :

<Résultat> :	Type de données des variables système ou utilisateur spécifiées		
	Type de données :	INT	
	Plage de valeurs :	1	= BOOL
		2	= CHAR
		3	= INT
		4	= REAL
		5	= STRING
		6	= AXIS
		7	= FRAME
		En cas d'erreur :	
		< 0	Le nom de variable spécifié n'est pas associé à un paramètre système ni à une variable utilisateur.
GETVARTYP :	Lecture du type de données d'une variable système ou utilisateur		
<Nom> :	Nom de la variable système ou utilisateur		
	Type de données :	STRING	

Exemple :

```
Code de programme                    Commentaire
DEF INT result=0
DEF STRING name="R"
result=GETVARTYP(name)               ; Détermine le type du paramètre R.
```

Code de programme	**Commentaire**
`IF (result < 0) GOTOF error`	

Le résultat signalé en retour est la valeur 4 qui correspond au type de données REAL.

3.1.1.17 Conversions de types possibles

La valeur numérique constante, la variable ou l'expression assignée à une variable doit être compatible avec le type de cette variable. Si cette condition n'est pas remplie, le type est automatiquement converti au moment de l'affectation.

Conversions de type possibles

en	REAL	INT	BOOL	CHAR	STRING	AXIS	FRAME
de							
REAL	oui	oui*	oui[1])	oui*	–	–	–
INT	oui	oui	oui[1])	oui[2])	–	–	–
BOOL	oui	oui	oui	oui	oui	–	–
CHAR	oui	oui	oui[1])	oui	oui	–	–
STRING	–	–	oui[4])	oui[3])	oui	–	–
AXIS	–	–	–	–	–	oui	–
FRAME	–	–	–	–	–	–	oui

Observations

* En cas de conversion du type REAL en type INT, une partie décimale >= 0,5 est arrondie par excès ; dans le cas contraire, elle est arrondie par défaut (voir fonction ROUND)

1) La valeur <> 0 équivaut à TRUE, la valeur == 0 équivaut à FALSE

2) Si la valeur se situe dans la plage numérique admise

3) Dans le cas d'un seul caractère

4) Longueur de chaîne de caractères 0 = >FALSE, sinon TRUE

Remarque

Un message d'erreur est émis lorsqu'une valeur de conversion est supérieure à la plage cible.

Lorsque des types différents figurent dans une même expression, leur adaptation s'effectue automatiquement. Des conversions de type sont aussi possibles dans des actions synchrones, voir le chapitre Actions synchrones au déplacement, conversion de type implicite".

3.1.2 Programmation indirecte

3.1.2.1 Programmation indirecte d'adresses

Dans le cas de la programmation indirecte d'adresses (Page 1323), l'adresse étendue (<Indice>) est remplacée par une variable du type approprié.

Remarque

La programmation indirecte d'adresses est impossible pour :

- N (numéro de bloc)
- L (sous-programme)
- Adresses réglables
  (par exemple, X[1] au lieu de X1 n'est pas autorisé)

Syntaxe

```
<ADRESSE>[<indice>]
```

Signification

`<ADRESSE>[...]` :	Adresse fixe avec extension (indice)
`<Indice>` :	Variable telle que n° de broche, axe, etc.

Exemples

Exemple 1 : Programmation indirecte d'un numéro de broche

Programmation directe :

Code de programme	Commentaire
`S1=300`	`; Vitesse de rotation de 300 tr/min pour la broche avec le numéro 1.`

Programmation indirecte :

Code de programme	Commentaire
`DEF INT NUMBRO=1`	`; Définition des variables de type INT et affectation de valeur.`
`S[SPINU]=300`	`; Vitesse de rotation de 300 tr/min pour la broche dont le numéro est consigné dans la variable SPINU (dans cet exemple, la broche avec le numéro 1).`

Exemple 2 : Programmation indirecte d'un axe

Programmation directe :

Code de programme	Commentaire
`FA[U]=300`	`; Avance de 300 pour l'axe "U".`

Programmation indirecte :

Code de programme	Commentaire
DEF AXIS AXVAR2=U	; Définition d'une variable de type AXIS et affectation de valeur.
FA[AXVAR2]=300	; Avance de 300 pour l'axe dont le nom d'adresse est consigné dans la variable AXVAR2.

Exemple 3 : Programmation indirecte d'un axe

Programmation directe :

Code de programme	Commentaire
$AA_MM[X]	; Lecture de la valeur de mesure du palpeur (SCM) de l'axe "X".

Programmation indirecte :

Code de programme	Commentaire
DEF AXIS AXVAR3=X	; Définition d'une variable de type AXIS et affectation de valeur.
$AA_MM[AXVAR3]	; Lecture de la valeur de mesure du palpeur (SCM) pour l'axe dont le nom est consigné dans la variable AXVAR3.

Exemple 4 : Programmation indirecte d'un axe

Programmation directe :

Code de programme
X1=100 X2=200

Programmation indirecte :

Code de programme	Commentaire
DEF AXIS AXVAR1 AXVAR2	; Définition de deux variables de type AXIS.
AXVAR1=(X1) AXVAR2=(X2)	; Affectation des noms d'axes.
AX[AXVAR1]=100 AX[AXVAR2]=200	; Déplacement des axes dont les noms d'adresse sont consignés dans les variables AXVAR1 et AXVAR2.

Exemple 5 : Programmation indirecte d'un axe

Programmation directe :

Code de programme
G2 X100 I20

Programmation indirecte :

Code de programme	Commentaire
DEF AXIS AXVAR1=X	; Définition d'une variable de type AXIS et affectation de valeur.
G2 X100 IP[AXVAR1]=20	; Programmation indirecte du centre de l'axe dont le nom d'adresse est consigné dans la variable AXVAR1.

Exemple 6 : Programmation indirecte d'éléments de tableau

Programmation directe :

Code de programme	Commentaire
`DEF INT TABLEAU1[4,5]`	`; Définition du tableau 1.`

Programmation indirecte :

Code de programme	Commentaire

`DEFINE DIM1 AS 4`
`DEFINE DIM2 AS 5`
`DEF INT TA-`
`BLEAU[DIM1,DIM2]`
`TABLEAU[DIM1-1,DIM2-1]=5`
	`; Les dimensions de tableau sont à indiquer comme valeurs fixes.`

Exemple 7 : Appel indirect de sous-programme

Code de programme	Commentaire
`CALL "L" << R10`	`; Appel du programme dont le numéro figure dans R10 (concaténation de chaînes de caractères).`

3.1.2.2 Programmation indirecte d'instructions G

La programmation indirecte des instructions G (Page 1337) permet une programmation effective des cycles.

Syntaxe

`G[<groupe>]=<numéro>`

Signification

`G[...]` :	Instruction G avec extension (indice)	
`<groupe>` :	Paramètre d'indice : Groupe G	
	Type :	INT
`<numéro>` :	Variable du numéro d'instruction G	
	Type :	INT ou REAL

Remarque

En règle générale, seules les instructions G non déterminantes pour la syntaxe peuvent être programmées indirectement.

Parmi les instructions G déterminantes pour la syntaxe, seuls celles du groupe de fonctions G 1 sont possibles.
Les instructions G déterminantes pour la syntaxe des groupes de fonction 2, 3 et 4 ne sont pas possibles.

Remarque

La programmation indirecte des instructions G n'admet pas les fonctions arithmétiques. Le calcul requis du numéro d'instruction G doit s'effectuer dans une ligne distincte du programme pièce, avant la programmation indirecte de l'instruction G.

Exemples

Exemple 1 : Décalage d'origine réglable (groupe G 8)

Code de programme	Commentaire
N1010 DEF INT INT_VAR	
N1020 INT_VAR=2	
...	
N1090 G[8]=INT_VAR G1 X0 Y0	;G54
N1100 INT_VAR=INT_VAR+1	; Calcul d'instruction G
N1110 G[8]=INT_VAR G1 X0 Y0	;G55

Exemple 2 : Sélection de niveau (groupe G 6)

Code de programme	Commentaire
N2010 R10=$P_GG[6]	; Lire l'instruction G active du Groupe G 6
...	
N2090 G[6]=R10	

3.1.2.3 Programmation indirecte d'attributs de position (GP)

Les attributs de position tels que la programmation relative ou absolue de la position d'axe peuvent être programmés indirectement en tant que variables, en liaison avec le mot clé GP.

Application

La programmation indirecte d'attributs de position s'utilise dans les **cycles de substitution**, car elle présente alors l'avantage suivant par rapport à la programmation des attributs de position comme mot clé (IC, AC, etc.)

Avec la programmation indirecte en tant que variables, **aucune** instruction CASE n'est nécessaire pour accéder à tous les attributs de position possibles.

Syntaxe

```
<INSTRUCTION DE POSITIONNEMENT>[<axe/broche>]=
GP(<position>,<attribut de position>)
<Axe/broche>=GP(<position>,<attribut de position>)
```

Signification

<INSTRUCTION DE POSITIONNEMENT>[] :	Les instructions de positionnement suivantes peuvent être programmées avec le mot clé GP : POS, POSA, SPOS, SPOSA Autres possibilités : • Tous les descripteurs d'axe/broche présents dans le canal : <Axe/broche> • Descripteur d'axe/broche variable AX
<axe/broche> :	Axe/broche à positionner
GP() :	Mot clé pour le positionnement
<position> :	Paramètre 1 Position d'axe/broche comme constante ou variable
<attribut de position> :	Paramètre 2 Attribut de position (par exemple le mode d'accostage de la position) comme variable (par exemple $P_SUB_SPOSMODE) ou comme mot clé (IC, AC, etc.)

Les valeurs fournies par les variables ont la signification suivante :

Valeur	Signification	Admis pour :
0	Aucune modification de l'attribut de position	
1	AC	POS, POSA,SPOS, SPOSA,AX, adresse d'axe
2	IC	POS, POSA,SPOS, SPOSA,AX, adresse d'axe
3	DC	POS, POSA,SPOS, SPOSA,AX, adresse d'axe
4	ACP	POS, POSA,SPOS, SPOSA,AX, adresse d'axe
5	ACN	POS, POSA,SPOS, SPOSA,AX, adresse d'axe
6	OC	-
7	PC	-
8	DAC	POS, POSA,AX, adresse d'axe
9	DIC	POS, POSA,AX, adresse d'axe
10	RAC	POS, POSA,AX, adresse d'axe
11	RIC	POS, POSA,AX, adresse d'axe
12	CAC	POS, POSA
13	CIC	POS, POSA
14	CDC	POS, POSA
15	CACP	POS, POSA
16	CACN	POS, POSA

Exemple

En cas de couplage actif de broches synchrones entre la broche pilote S1 et la broche asservie S2, l'instruction SPOS appelle le cycle de substitution suivant pour le positionnement des broches dans le programme principal.

Le positionnement est effectué par l'instruction de N2230 :

```
SPOS[1]=GP($P_SUB_SPOSIT,$P_SUB_SPOSMODE)
SPOS[2]=GP($P_SUB_SPOSIT,$P_SUB_SPOSMODE)
```

La position à accoster est lue dans la variable système $P_SUB_SPOSIT et le mode d'accostage de la position dans la variable système $P_SUB_SPOSMODE.

Code de programme	**Commentaire**
N1000 PROC LANG_SUB DISPLOF SBLOF	
...	
N2100 IF($P_SUB_AXFCT==2)	
N2110	; Substitution de l'instruction SPOS / SPOSA / M19 en cas de couplage actif de broches synchrones
N2185 DELAYFSTON	; Début de la plage d'arrêt temporisé
N2190 COUPOF(S2,S1)	; Désactivation du couplage de broches synchrones
N2200	; Positionnement de la broche pilote et de la broche asservie
N2210 IF($P_SUB_SPOS==TRUE) OR ($P_SUB_SPOSA==TRUE)	
N2220	; Positionnement de broche avec SPOS :
N2230 SPOS[1]=GP($P_SUB_SPOSIT, $P_SUB_SPOSMODE)	
SPOS[2]=GP($P_SUB_SPOSIT, $P_SUB_SPOSMODE)	
N2250 ELSE	
N2260	; Positionnement de broche avec M19 :
N2270 M1=19 M2=19	; Positionnement de la broche pilote et de la broche asservie
N2280 ENDIF	
N2285 DELAYFSTOF	; Fin de la plage d'arrêt temporisé
N2290 COUPON(S2,S1)	; Activation du couplage de broches synchrones
N2410 ELSE	
N2420	; Interrogation de l'existence d'autres substitutions
...	
N3300 ENDIF	
...	
N9999 RET	

Autres conditions à prendre en compte

Dans les actions synchrones, la programmation indirecte des attributs de position est impossible.

3.1.2.4 Programmation indirecte de lignes de programme pièce (EXECSTRING)

Une variable chaîne de caractères créée au préalable peut être exécutée en tant que ligne de programme pièce au moyen de l'instruction `EXECSTRING`.

Syntaxe

EXECSTRING est programmée dans une ligne spécifique du programme pièce :
EXECSTRING (<variable de type STRING>)

Signification

EXECSTRING :	Instruction pour l'exécution d'une variable chaîne de caractères en tant que ligne de programme pièce
<variable de type STRING> :	Variable du type STRING dans laquelle figure la ligne de programme pièce à exécuter proprement dite

Remarque

Tous les éléments susceptibles d'être programmés **dans la partie exécutable d'un programme pièce** peuvent être transmis avec EXECSTRING, à l'exception des structures de contrôle (Page 484). Les instructions PROC et DEF et, d'une manière générale, l'utilisation dans des fichiers INI et DEF en sont exclues.

Exemple

```
Code de programme                    Commentaire
N100 DEF STRING[100] MY_BLOCK        ; Définition des variables chaînes de caractères
                                     pour la prise en compte des lignes de programme piè-
                                     ce à exécuter.
N110 DEF STRING[10] MFCT1="M7"
...
N200 EXECSTRING(MFCT1 <<             ; Exécution de la ligne "M7 M4711" du programme piè-
"M4711")                             ce.
...
N300 R10=1
N310 MY_BLOCK="M3"
N320 IF(R10)
N330 MY_BLOCK = MY_BLOCK <<
MFCT1
N340 ENDIF
N350 EXECSTRING(MY_BLOCK)            ; Exécution de la ligne "M3 M7" du programme pièce.
```

3.1.3 Opérations

3.1.3.1 Fonctions de calcul

Opérateur arithmétique / fonction de calcul	Signification
+	Addition
–	Soustraction
*	Multiplication
/ [1)]	Division[1)]
DIV[1)]	Division entière[1)]
MOD[1)]	Division modulo (fournit le reste de la division entière)[1)]
:	Opérateur de concaténation pour variables de type FRAME
SIN()	Sinus
COS()	Cosinus
TAN()	Tangente
ASIN()	Arc sinus
ACOS()	Arc cosinus
ATAN2(,) [1)]	Arc tangente 2[1)]
SQRT( )	Racine carrée
ABS( )	Valeur absolue
POT( )	Puissance 2 (carré)
TRUNC( )	Partie entière Précision réglable avec TRUNC dans les instructions relationnelles (voir "Correction de la précision pour les erreurs relationnelles (TRUNC) (Page 454)")
ROUND()	arrondir à un nombre entier
LN( )	Logarithme naturel
EXP( )	Fonction exponentielle
MINVAL()	valeur la plus petite de deux variables (voir "Minimum, maximum et plage de variables (MINVAL, MAXVAL, BOUND) (Page 433)")
MAXVAL()	valeur la plus grande de deux variables (voir "Minimum, maximum et plage de variables (MINVAL, MAXVAL, BOUND) (Page 433)")
BOUND()	valeur de variable se trouvant dans la plage de valeurs définie (voir "Minimum, maximum et plage de variables (MINVAL, MAXVAL, BOUND) (Page 433)")
CTRANS()	Décalage
CROT ()	Rotation
CSCALE ()	Changement d'échelle
CMIRROR ()	Fonction miroir
1) Voir le paragraphe "Exemples"	

Programmation

La notation mathématique usuelle est valable pour les fonctions de calcul. Les priorités de traitement sont fixées par des parenthèses. Les fonctions trigonométriques et leurs inverses emploient la notation en degrés (angle droit = 90°).

Exemples

Division : /

(Type REAL) = (Type INT ou type REAL) / (Type INT ou type REAL) ;

Exemple : 3 / 4 = 0.75

Division entière : DIV

(Type INT) = (Type INT ou REAL) / (Type INT ou REAL) ;

Exemple : 7 DIV 4,1 = 1

Division modulo (fournit le reste de la division entière) : MOD

(Type REAL) = (Type INT ou REAL) MOD (Type INT ou REAL) ;

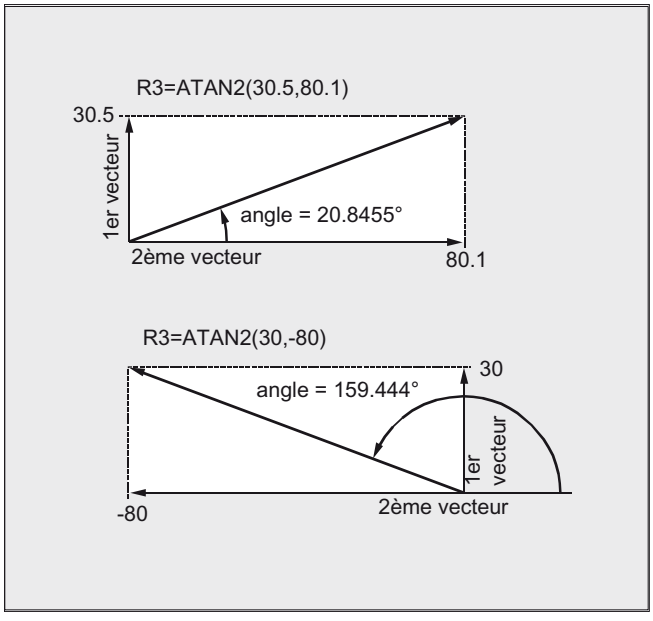
Exemple : 7 MOD 4,1 = 2,9

Arc tangente 2 : ATAN2

A partir de deux vecteurs orthogonaux, la fonction de calcul ATAN2 détermine l'angle du vecteur somme.

Le résultat s'inscrit dans la plage des quatre quadrants (-180° < 0 < +180°).

La base de la référence angulaire est toujours représentée par la 2e valeur dans le sens positif.



Exemples de programmation

Code de programme	Commentaire
R1=R1+1	; Nouveau R1 = ancien R1 + 1
R1=R2+R3 R4=R5-R6 R7=R8*R9	
R10=R11/R12 R13=SIN(25.3)	
R14=R1*R2+R3	; La multiplication est prioritaire.
R14=(R1+R2)*R3	; Les expressions entre parenthèses sont prioritaires.
R15=SQRT(POT(R1)+POT(R2))	; Les parenthèses internes sont levées en premier : R15 = racine carrée ((R1^2 + R2^2))
RESFRAME=FRAME1:FRAME2 FRAME3=CTRANS(…):CROT(…)	; Combinaison FRAME avec opérateur de concaténation Affectation de valeurs à une composante FRAME

3.1.3.2 Opérations relationnelles et opérations logiques

Les opérations relationnelles peuvent servir par exemple à formuler une condition de saut dans le programme. Elles s'appliquent également aux expressions complexes.

Les opérations relationnelles sont applicables aux variables de type CHAR, INT, REAL et BOOL. Avec le type CHAR, la valeur de code est comparée.
Pour les types STRING, AXIS et FRAME, sont possibles : == et <> qui peuvent aussi être employés pour les opérations de type STRING, y compris dans des actions synchrones.

Le résultat d'opérations relationnelles est toujours du type BOOL.

Les opérateurs logiques servent à la combinaison de valeurs booléennes.

Les opérations logiques s'appliquent exclusivement aux variables du type BOOL. Avec la conversion de type interne, ils sont également applicables aux types de données CHAR, INT et REAL.

Dans le cas des opérations logiques (booléennes), la convention suivante s'applique aux données du type BOOL, CHAR, INT et REAL :

- 0 correspond à : FALSE
- valeur différente de 0 correspond à : TRUE

Opérateurs logiques sur bits

Les variables des types CHAR et INT permettent également d'effectuer des opérations logiques sur bits. Le cas échéant, il est procédé à une conversion automatique de type.

Programmation

Opérateur relationnel	Signification
==	égal à
<>	différent de
>	supérieur à
<	inférieur à

>=	supérieur ou égal à
<=	inférieur ou égal à

Opérateur logique	**Signification**
AND	ET
OR	OU
NOT	Négation
XOR	OU exclusif

Opérateur logique sur bits	**Signification**
B_AND	ET sur bits
B_OR	OU sur bits
B_NOT	Négation sur bits
B_XOR	OU exclusif sur bits

Remarque

Des parenthèses placées dans les expressions arithmétiques permettent de fixer un ordre des opérations différent des règles normales de priorité.

Remarque

Entre les opérandes booléens et les opérateurs doivent figurer des espacements.

Remarque

L'opérateur B_NOT ne se rapporte qu'à un seul opérande qui suit l'opérateur.

Exemples

Exemple 1 : opérateurs relationnels

```
IF R10>=100 GOTOF DESTINATION
```

ou

```
R11=R10>=100
IF R11 GOTOF DESTINATION
```

Le résultat de la comparaison R10>=100 est d'abord mémorisé dans R11.

Exemple 2 : opérateurs logiques

```
IF (R10<50) AND ($AA_IM[X]>=17.5) GOTOF DESTINATION
```

ou

```
IF NOT R10 GOTOB DEPART
```

L'opérateur NOT ne se rapporte qu'à un seul opérande.

Exemple 3 : opérateurs logiques sur bits

```
IF $MC_RESET_MODE_MASK B_AND 'B10000' GOTOF ACT_PLANE
```

3.1.3.3 Priorité des opérations

A chaque opérateur est affectée une priorité. Lors du traitement d'une expression, les opérateurs au degré de priorité le plus élevé sont appliqués en premier lieu. En cas d'opérateurs d'égale priorité, le traitement s'opère de gauche à droite.

Des parenthèses placées dans les expressions arithmétiques permettent de fixer un ordre des opérations différent des règles normales de priorité.

Ordre des opérateurs

De la priorité la plus élevée à la moins élevée

1.	NOT, B_NOT	Négation, négation sur bits
2.	*, /, DIV, MOD	Multiplication, division
3.	+, –	Addition, soustraction
4.	B_AND	ET sur bits
5.	B_XOR	OU exclusif sur bits
6.	B_OR	OU sur bits
7.	AND	ET
8.	XOR	OU exclusif
9.	OR	OU
10.	<<	Concaténation de chaînes de caractères, type de résultat STRING
11.	==, <>, >, <, >=, <=	Opérateurs relationnels

Remarque

L'opérateur de concaténation ":" pour frames ne doit pas figurer avec d'autres opérateurs dans une même expression. Cet opérateur ne nécessite donc pas d'ordre de priorité.

Exemple d'instruction If

```
If (otto==10) and (anna==20) gotof end
```

3.1.3.4 Correction de la précision pour les erreurs relationnelles (TRUNC)

L'instruction TRUNC ampute l'opérande de la multiplication par un facteur de précision.

Précision réglable dans les instructions relationnelles

En interne, les données de type REAL du programme pièce sont représentées par 64 bits en format IEEE. De ce fait, les nombres décimaux ne peuvent pas être représentés de manière précise et leur comparaison avec des valeurs calculées de façon idéale peut conduire à des résultats inattendus.

Egalité relative

Afin que les imprécisions causées par le format de représentation ne faussent pas le flux du programme, les instructions relationnelles ne vérifient pas l'égalité absolue, mais l'égalité relative.

Syntaxe

Correction de la précision pour les erreurs relationnelles

```
TRUNC (R1*1000)
```

Signification

TRUNC :	troncature des décimales

Egalité relative de 10^{-12} pour

- Egalité : (==)
- Différence : (<>)
- Supérieur ou égal : (>=)
- Inférieur ou égal : (<=)
- Supérieur/inférieur : (><) avec égalité absolue
- Supérieur : (>)
- Inférieur : (<)

Compatibilité

Pour des raisons de compatibilité, le contrôle d'égalité relative de (>) et (<) peut être désactivé par définition du paramètre machine MD10280 $MN_ PROG_FUNCTION_MASK Bit0 = 1.

Remarque

Pour les raisons qui ont été citées, les comparaisons avec des données du type REAL connaissent d'une manière générale une certaine imprécision. Lorsque les écarts ne sont pas acceptables, vous devez effectuer des calculs de type INTEGER en multipliant les opérandes par un facteur de précision, puis en les amputant avec TRUNC.

Actions synchrones

Ce qui a été décrit pour les instructions relationnelles est aussi valable pour les actions synchrones.

Exemples

Exemple 1 : considérations relatives à la précision

Code de programme	Commentaire
`N40 R1=61.01 R2=61.02 R3=0.01`	`; affectation des valeurs initiales`

Code de programme	Commentaire
N41 IF ABS(R2-R1) > R3 GOTOF ERREUR	; Le saut serait exécuté jusqu'ici.
N42 M30	; Fin du programme
N43 ERREUR : SETAL(66000)	
R1=61.01 R2=61.02 R3=0.01	; affectation des valeurs initiales
R11=TRUNC(R1*1000) R12=TRUNC(R2*1000) R13=TRUNC(R3*1000)	; Correction de la précision
IF ABS(R12-R11) > R13 GOTOF ERREUR	; Le saut n'est plus exécuté.
M30	; Fin du programme
ERREUR : SETAL(66000)	

Exemple 2 : formation et analyse du quotient des deux opérandes

Code de programme	Commentaire
R1=61.01 R2=61.02 R3=0.01	; affectation des valeurs initiales
IF ABS((R2-R1)/R3)-1) > 10EX-5 GOTOF ERREUR	; Le saut n'est pas exécuté.
M30	; Fin du programme
ERREUR : SETAL(66000)	

3.1.3.5 Arrondissement (ROUNDUP)

La fonction "ROUNDUP" arrondit les valeurs d'entrée de type REAL (nombres rationnels avec point décimal) à l'entier supérieur le plus proche.

Syntaxe

```
ROUNDUP(<valeur>)
```

Signification

ROUNDUP :	Instruction d'arrondissement d'une valeur d'entrée
<valeur> :	Valeur d'entrée du type REAL

Remarque

Les valeurs d'entrée de type INT (nombres entiers) restent inchangées.

Exemples

Exemple 1 : arrondis de différentes valeurs d'entrée

Exemple	Arrondi
ROUNDUP(3.1)	4.0
ROUNDUP(3.6)	4.0
ROUNDUP(-3.1)	-3.0
ROUNDUP(-3.6)	-3.0

`ROUNDUP(3.0)`	3.0
`ROUNDUP(3)`	3.0

Exemple 2 : ROUNDUP dans le programme CN

Code de programme
`N10 X=ROUNDUP(3.5) Y=ROUNDUP(R2+2)`
`N15 R2=ROUNDUP($AA_IM[Y])`
`N20 WHEN X=100 DO Y=ROUNDUP($AA_IM[X])`
`...`

3.1.4 Opérations sur les chaînes de caractères

Opérations sur les chaînes de caractères

Outre les opérations classiques "Affectation" et "Comparaison", les opérations suivantes sont possibles sur les chaînes de caractères :

- Conversion de type en type STRING (AXSTRING) (Page 458)
- Conversion de type à partir de STRING (NUMBER, ISNUMBER, AXNAME) (Page 458)
- Concaténation de chaînes de caractères (<<) (Page 459)
- Conversion en minuscules/majuscules (TOLOWER, TOUPPER) (Page 460)
- Déterminer la longueur d'une chaîne de caractères (STRLEN) (Page 461)
- Recherche de caractères/chaînes de caractères dans une chaîne (INDEX, RINDEX, MINDEX, MATCH) (Page 462)
- Sélection d'une chaîne de caractères partielle (SUBSTR) (Page 463)
- Lecture et écriture de caractères individuels (Page 464)
- Formatage d'une chaîne (SPRINT) (Page 465)

Signification particulière du caractère 0

En interne, le caractère 0 est interprété comme identification de fin d'une chaîne de caractères. Le remplacement d'un caractère par 0 raccourcit donc la chaîne de caractères.

Exemple :

Code de programme	**Commentaire**
`DEF STRING[20] STRG="axe . à l'arrêt"`	
`STRG[6]="X"`	
`MSG(STRG)`	`; Fournit le message "Axe X à l'arrêt".`
`STRG[6]=0`	

Code de programme	Commentaire
`MSG(STRG)`	`; Fournit le message "Axe".`

3.1.4.1 Conversion de type en type STRING (AXSTRING)

La fonction "Conversion de type en type STRING" permet d'utiliser des variables de types différents comme éléments d'un message (MSG).

Lors de l'utilisation de l'opérateur <<, la fonction est exécutée implicitement pour les données de type INT, REAL, CHAR et BOOL (voir " Concaténation de chaînes de caractères (<<) (Page 459) ").

Une valeur INT est convertie sous la forme normale lisible. Les valeurs REAL sont données avec un nombre de décimales pouvant aller jusqu'à 10.

L'instruction `AXSTRING` convertit des variables de type AXIS en type STRING.

Syntaxe

```
<STRING_ERG> = << <type_quelconque>
<STRING_ERG> = AXSTRING(<descripteur d'axe>)
```

Signification

`<STRING_ERG>` :	Variable du résultat de la conversion de type	
	Type :	STRING
`<Type_quelconque>` :	Types de variables INT, REAL, CHAR, STRING et BOOL	
`AXSTRING` :	L'instruction `AXSTRING` fournit le descripteur d'axe indiqué comme chaîne de caractères.	
`<descripteur d'axe>` :	Variable du descripteur d'axe	
	Type :	AXIS

Remarque

Les variables FRAME ne peuvent pas être converties.

3.1.4.2 Conversion de type à partir de STRING (NUMBER, ISNUMBER, AXNAME)

Avec l'instruction `NUMBER`, la conversion se fait du type STRING en type REAL.
L'instruction `ISNUMBER` s'utilise pour vérifier si la conversion est possible.

L'instruction `AXNAME` convertit une chaîne de caractères de type STRING en type AXIS.

Syntaxe

```
<RES_REAL>=NUMBER("<chaîne de caractères>")
<RES_BOOL>=ISNUMBER("<chaîne de caractères>")
<RES_AXIS>=AXNAME("<chaîne de caractères>")
```

Signification

NUMBER :	L'instruction NUMBER donne en retour le nombre représenté par la <chaîne de caractères> comme valeur REAL.		
<chaîne> :	Variable de type STRING à convertir		
<REAL_ERG> :	Variable du résultat de la conversion de type avec NUMBER		
	Type :	REAL	
ISNUMBER :	L'instruction ISNUMBER permet de contrôler si la <chaîne de caractères> est convertible en un nombre valable.		
<BOOL_ERG> :	Variable du résultat de l'interrogation avec ISNUMBER		
	Type :	BOOL	
	Valeur :	TRUE	ISNUMBER fournit la valeur TRUE quand la <chaîne de caractères> représente un nombre REAL valide selon les règles du langage.
		FALSE	Si ISNUMBER indique la valeur FALSE, une alarme est émise lors de l'appel de NUMBER avec la même <chaîne de caractères>.
AXNAME :	L'instruction AXNAME transforme la <chaîne de caractères> indiquée en un descripteur d'axe. **Remarque :** Une alarme est émise si la <chaîne de caractères> ne peut pas être affectée à un descripteur d'axe existant.		
<AXIS_ERG> :	Variable du résultat de la conversion de type avec AXNAME		
	Type :	AXIS	

Exemple

```
Code de programme                                Commentaire
DEF BOOL BOOL_ERG
DEF REAL REAL_ERG
DEF AXIS AXIS_ERG
RES_BOOL=ISNUMBER("1234.9876Ex-7")               ; BOOL_ERG == TRUE
RES_BOOL=ISNUMBER("1234XYZ")                     ; BOOL_ERG == FALSE
RES_REAL=NUMBER("1234.9876Ex-7")                 ; REAL_ERG == 1234.9876Ex-7
RES_AXIS=AXNAME("X")                             ; AXIS_ERG == X
```

3.1.4.3 Concaténation de chaînes de caractères (<<)

La fonction "Concaténation de chaînes de caractères" permet de composer une chaîne de caractères à partir d'éléments individuels.

La concaténation s'effectue avec l'opérateur "<<". Cet opérateur possède, pour toutes les combinaisons des types de base, CHAR, BOOL, INT, REAL et STRING, comme type de destination STRING. Une conversion éventuellement indispensable est entreprise selon les règles établies.

Syntaxe

```
<Type_quelconque> << <type_quelconque>
```

Signification

<Type_quelconque> :	Variable de type CHAR, BOOL, INT, REAL ou STRING
<< :	Opérateur pour la concaténation de variables (<type_quelconque>) en chaîne de caractères composée (type STRING). Cet opérateur est également disponible seul en variante dite "unaire". permettant d'effectuer une conversion de type explicite en type STRING (à l'exception des types FRAME et AXIS) : << <type_quelconque>

Il est ainsi possible, par exemple, de composer un message ou un ordre à partir de listes de textes et d'insérer des paramètres (nom de bloc, etc.) :
`MSG(STRG_TAB[LOAD_IDX]<<NOM_BLOC)`

Remarque

Les résultats intermédiaires de la concaténation de la chaîne de caractères ne doivent pas dépasser la longueur de chaîne maximale.

Remarque

L'utilisation des types FRAME et AXIS est impossible avec l'opérateur "<<".

Exemples

Exemple 1 : Chaînage de caractères

```
DEF INT IDX=2
DEF REAL VALUE=9.654
DEF STRING[20] STRG="INDEX:2"
IF STRG=="Index:"<<IDX GOTOF NO_MSG
MSG("Index:"<<IDX<<"/valeur:"<<VALUE)          ; Affichage :
                                               "Indice:2/valeur:9,654"
NO_MSG:
```

Exemple 2 : conversion explicite de type avec <<

```
DEF REAL VALUE=3.5
<<VALUE                  ; La variable spécifiée de type REAL est convertie en type STRING.
```

3.1.4.4 Conversion en minuscules/majuscules (TOLOWER, TOUPPER)

La fonction "Conversion en minuscules/majuscules" permet d'harmoniser l'écriture au sein d'une chaîne de caractères.

Syntaxe

```
<RES_STRING>=TOUPPER("<chaîne de caractères>")
<RES_STRING>=TOLOWER("<chaîne de caractères>")
```

Signification

TOUPPER :	L'instruction TOUPPER convertit toutes les lettres d'une chaîne de caractères en **majuscules**.	
TOLOWER :	L'instruction TOLOWER convertit toutes les lettres d'une chaîne de caractères en **minuscules**.	
<chaîne> :	Chaîne de caractères à convertir	
	Type :	STRING
<STRING_ERG> :	Variable du résultat de la conversion	
	Type :	STRING

Exemple

Etant donné qu'il est possible de rattacher à l'interface utilisateur des indications introduites par l'utilisateur, cette fonctionnalité permet d'obtenir une présentation uniformisée en lettres minuscules ou majuscules:

Code de programme

```
DEF STRING [29] STRG
...
IF "LEARN.CNC"==TOUPPER(STRG) GOTOF LOAD_LEARN
```

3.1.4.5 Déterminer la longueur d'une chaîne de caractères (STRLEN)

L'instruction STRLEN permet de déterminer la longueur d'une chaîne de caractères.

Syntaxe

```
<RES_INT>=STRLEN("<chaîne de caractères>")
```

Signification

STRLEN :	L'instruction STRLEN détermine la longueur de la chaîne de caractères indiquée. Le résultat est le nombre de caractères comptés à partir du début de la chaîne et qui ne sont pas des caractères 0.	
<chaîne> :	Chaîne de caractères dont la longueur doit être déterminée	
	Type :	STRING
<INT_ERG> :	Variable du résultat de la détermination	
	Type :	INT

Exemple

En liaison avec l'accès à des caractères individuels, la fonction permet de déterminer la fin d'une chaîne de caractères :

Code de programme

```
IF (STRLEN(NOM_BLOC)>10) GOTOF ERREUR
```

3.1.4.6 Recherche de caractères/chaînes de caractères dans une chaîne (INDEX, RINDEX, MINDEX, MATCH)

Cette fonctionnalité permet de rechercher des caractères isolés ou une suite de caractères dans une chaîne de caractères. Les résultats signalent à quel endroit de la chaîne de caractères se trouve le caractère / la suite de caractères qui fait l'objet de la recherche.

Syntaxe

`INT_ERG=INDEX(STRING,CHAR)` ; **Type de résultat : INT**

`INT_ERG=RINDEX(STRING,CHAR)` ; **Type de résultat : INT**

`INT_ERG=MINDEX(STRING,STRING)` ; **Type de résultat : INT**

`INT_ERG=MATCH(STRING,STRING)` ; **Type de résultat : INT**

Sémantique

Fonctions de recherche : elles fournissent la position trouvée dans la chaîne de caractères (premier paramètre). Si le caractère ou la chaîne de caractères est introuvable, le résultat est -1, la position du premier caractère étant 0.

Signification

`INDEX` :	Recherche du caractère (depuis le début) indiqué comme deuxième paramètre dans le premier paramètre.
`RINDEX` :	Recherche du caractère (depuis la fin) indiqué comme deuxième paramètre dans le premier paramètre.
`MINDEX` :	Correspond à la fonction INDEX, sauf qu'elle fournit une liste de caractères (sous forme de chaîne) avec l'indice du premier caractère trouvé.
`MATCH` :	Recherche d'une chaîne de caractères dans une chaîne de caractères.

Il est ainsi possible de décomposer les chaînes de caractères suivant des critères définis tels que les espaces et les caractères de séparation des chemins ("/").

Exemple

Décomposition d'une entrée en noms de chemin et de bloc

Code de programme	Commentaire
`DEF INT CHEMIDX, PROGIDX`	
`DEF STRING[26] ENTREE`	

Code de programme	Commentaire
DEF INT LISTIDX	
ENTREE = "/_N_MPF_DIR/_N_EXECU- TE_MPF"	
LISTIDX = MINDEX (ENTREE, "M,N,O,P") + 1	; Le résultat de LISTIDX est la valeur 3, "N" étant le premier caractère du paramètre ENTREE dans la liste, depuis le début.
CHEMIDX = INDEX (ENTREE, "/") +1	; On a donc : CHEMIDX = 1
PROGIDX = RINDEX (ENTREE, "/") +1	; On a donc : PROGIDX = 12
	; Avec la fonction SUBSTR présentée au chapitre suivant, il est possible de décomposer la variable ENTREE en "chemin" et en "bloc" :
VARIABLE = SUBSTR (ENTREE, CHEMIDX, PROGIDX-CHEMIDX-1)	; Fournit alors "_N_MPF_DIR".
VARIABLE = SUBSTR (ENTREE, PROGIDX)	; Fournit alors "_N_EXECUTE_MPF".

3.1.4.7 Sélection d'une chaîne de caractères partielle (SUBSTR)

La fonction SUBSTRING permet de lire des sous-parties quelconques d'une chaîne de caractères.

Syntaxe

```
<STRING_ERG>=SUBSTR(<chaîne>,<indice>,<longueur>)
<STRING_ERG>=SUBSTR(<chaîne>,<indice>)
```

Signification

SUBSTR :	La fonction fournit une chaîne partielle d'une <chaîne> de caractères de la <longueur> spécifiée, à partir de <indice>. Si le paramètre <longueur> n'est pas spécifié, la fonction fournit une chaîne partielle qui va de l'<indice> à la fin de la chaîne de caractères.
<indice> :	Position de début de la chaîne partielle dans la chaîne de caractères. Si la position de début se situe après la fin de la chaîne de caractères, le résultat est une chaîne vide (""). Premier caractère de la chaîne : Indice = 0 Plage de valeurs : 0 ... (longueur de chaîne de caractères - 1)
<longueur> :	Longueur de la chaîne partielle. Si la longueur spécifiée est trop grande, la chaîne partielle retournée s'arrête à la fin de la chaîne de caractères. Plage de valeurs : 1 ... (longueur de chaîne de caractères - 1)

Exemple

Code de programme	Commentaire
DEF STRING[29] RES	

Code de programme	Commentaire
; 1	
; 0123456789012345678	
ERG = SUBSTR("ACQUITTEMENT : 10 à 99", 10, 2)	; ERG == "10"
ERG = SUBSTR("ACQUITTEMENT : 10 à 99", 10)	; ERG == "10 à 99"

3.1.4.8 Lecture et écriture de caractères individuels

Il est possible de lire ou d'écrire des caractères individuellement dans une chaîne.

Les conditions marginales suivantes sont à prendre en considération :

- Cette fonction ne concerne que les variables utilisateur ; elle est inapplicable aux variables système
- Les caractères individuels d'une chaîne ne sont transmis que par paramètres "call by value" lors d'appels de sous-programmes

Syntaxe

```
<caractère>=<chaîne>[<indice>]
<caractère>=<tableau chaîne>[<indice de tableau>,<indice>]
<chaîne>[<indice>]=<caractère>
<tableau chaîne>[<indice de tableau>,<indice>]=<caractère>
```

Signification

<chaîne> :	Chaîne de caractères quelconque
<caractère> :	Variable de type CHAR
<indice> :	Position du caractère dans la chaîne Premier caractère de la chaîne : Indice = 0 Plage de valeurs : 0 ... (longueur de chaîne de caractères - 1)

Exemples

Exemple 1 : variable Message

Code de programme	Commentaire
; 0123456789	
DEF STRING [50] MESSAGE = "Axe n a atteint sa position"	
MESSAGE[6] = "X"	
MSG(MESSAGE)	; "Axe X a atteint sa position"

Exemple 2 : évaluation d'une variable système

Code de programme	Commentaire
DEF STRING[50] STRG	; Mémoire tampon pour variable système
...	
STRG = $P_MMCA	; Chargement de la variable système

Code de programme	Commentaire
IF STRG[0] == "E" GOTO ...	; Evaluation de la variable système

Exemple 3 : transfert des paramètres "call by value" et "call by reference"

Code de programme	Commentaire
; 0123456	
DEF STRING[50] STRG = "axe X"	
DEF CHAR CHR	
...	
EXTERN UP_VAL(AXE)	; Définition du sous-programme avec le paramètre "call by value"
EXTERN UP_REF(VAR AXE)	; Définition du sous-programme avec le paramètre "call by ref."
...	
UP_VAL(STRG[6])	; Transfert du paramètre "by value"
...	
CHR = STRG[6]	; Mise en mémoire tampon
UP_REF(CHR)	; Transfert du paramètre "by reference"

3.1.4.9 Formatage d'une chaîne (SPRINT)

Avec la fonction prédéfinie SPRINT, des chaînes de caractères peuvent être formatées et préparées, par exemple, pour la sortie vers des périphériques externes (voir aussi "Process DataShare - Sortie sur un appareil/fichier externe (EXTOPEN, WRITE, EXTCLOSE) (Page 1033)").

Syntaxe

```
"<Chaîne_résultat>"=SPRINT("<Chaîne_format>",<valeur_1>,<valeur_2>,..., <valeur_n>)
```

Signification

SPRINT :	Descripteur d'une fonction prédéfinie qui fournit une valeur de type STRING.
"<Chaîne_format>" :	Chaîne de caractères contenant une partie fixe et une partie variable. Les parties variables sont définies à l'aide du caractère de commande de format % et d'une description de format qui suit.
< valeur_1>,< valeur_2>,…,< valeur_n> :	Valeur se présentant sous la forme d'une constante ou d'une variable CN qui est insérée à l'emplacement du n-ième caractère de commande %, en fonction de la description du format, dans la <Chaîne_format>.
"<Chaîne_résultat>" :	Chaîne de caractères formatée (maximum 400 octets)

Descriptions de format disponibles

%B :	Conversion en chaîne de caractères "TRUE" lorsque la valeur à convertir :
	• est différente de 0,
	• n'est pas une chaîne de caractères vide (dans le cas de valeurs de type String).
	Conversion en chaîne de caractères "FALSE" lorsque la valeur à convertir :
	• est égale à 0,
	• est une chaîne de caractères vide.
	Exemple :
	`N10 DEF BOOL BOOL_VAR=1`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF BOOL_VAR:%B", BOOL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF BOOL_VAR:TRUE".

%C :	Conversion en caractères ASCII.
	Exemple :
	`N10 DEF CHAR CHAR_VAR="X"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF CHAR_VAR:%C",CHAR_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF CHAR_VAR:X".

%D :	Conversion en une chaîne de caractères comportant un nombre entier (INTEGER).
	Exemple :
	`N10 DEF INT INT_VAR=123`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF INT_VAR:%D",INT_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF INT_VAR:123".

%<m>D :	Conversion en une chaîne de caractères comportant un nombre entier (INTEGER). La longueur minimale de la chaîne est de <m> caractères. Les chiffres manquants sont remplis avec des caractères non significatifs alignés à gauche.
	Exemple :
	`N10 DEF INT INT_VAR=-123`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF INT_VAR:%6D",INT_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF INT_VAR:xx-123" (dans l'exemple, "x" remplace un caractère non significatif).

%F :	Conversion en une chaîne de caractères comportant un nombre à 6 décimales. Les décimales peuvent être arrondies ou complétées par des 0 non significatifs.
	Exemple :
	`N10 DEF REAL REAL_VAR=-1.2341234EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%F",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR: -1234.123400".

%<m>F :	Conversion en une chaîne de caractères comportant un nombre à 6 décimales et ayant une longueur totale d'au moins <m> caractères. Les décimales peuvent être arrondies ou complétées par des 0 non significatifs. Les caractères manquants pour atteindre la longueur totale <m> sont remplis avec des caractères non significatifs alignés à gauche.
	Exemple :
	`N10 DEF REAL REAL_VAR=-1.23412345678EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15F",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR: xxx-1234.123457" (dans l'exemple, "x" remplace un caractère non significatif).

%.<n>F :	Conversion en une chaîne de caractères comportant un nombre à <n> décimales. Les décimales peuvent être arrondies ou complétées par des 0 non significatifs.
	Exemple :
	`N10 DEF REAL REAL_VAR=-1.2345678EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3F",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:-1234.568".

%<m>.<n>F :	Conversion en une chaîne de caractères comportant un nombre à <n> décimales et ayant une longueur totale d'au moins <m> caractères. Les décimales peuvent être arrondies ou complétées par des 0 non significatifs. Les caractères manquants pour atteindre la longueur totale <m> sont remplis avec des caractères non significatifs alignés à gauche.
	Exemple :
	`N10 DEF REAL REAL_VAR=-1.2341234567890EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%10.2F",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:xx-1234.12" (dans l'exemple, "x" remplace un caractère non significatif).

%E :	Conversion en une chaîne de caractères comportant un nombre au format exponentiel. La mantisse est enregistrée de façon normalisée avec un chiffre avant le séparateur de décimales et 6 chiffres après. Les décimales peuvent être arrondies ou complétées par des 0 non significatifs. L'exposant commence par le mot clé "EX". Il est suivi du signe ("+" ou "-") et d'un nombre à deux ou trois chiffres.
	Exemple :
	`N10 DEF REAL REAL_VAR=-1234.567890`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%E",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:-1.234568EX+03".

%<m>E :	Conversion en une chaîne de caractères comportant un nombre décimal au format exponentiel avec une longueur totale d'au moins <m> caractères. Les caractères manquants sont remplis par des caractères non significatifs alignés à gauche. La mantisse est enregistrée de façon normalisée avec un chiffre avant le séparateur de décimales et 6 chiffres après. Les décimales peuvent être arrondies ou complétées par des 0 non significatifs. L'exposant commence par le mot clé "EX". Il est suivi du signe ("+" ou "-") et d'un nombre à deux ou trois chiffres.
	Exemple :
	`N10 DEF REAL REAL_VAR=-1234.5`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%20E",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:xxxxxx-1.234500EX+03" (dans l'exemple, "x" remplace un caractère non significatif).

%.<n>E :	Conversion en une chaîne de caractères comportant un nombre au format exponentiel. La mantisse est enregistrée de façon normalisée avec un chiffre avant le séparateur de décimales et <n> chiffres après. Les décimales peuvent être arrondies ou complétées par des 0 non significatifs. L'exposant commence par le mot clé "EX". Il est suivi du signe ("+" ou "-") et d'un nombre à deux ou trois chiffres.
	Exemple :
	`N10 DEF REAL REAL_VAR=-1234.5678`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.2E",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:-1.23EX+03".

%<m>.<n>E :	Conversion en une chaîne de caractères comportant un nombre décimal au format exponentiel avec une longueur totale d'au moins <m> caractères. Les caractères manquants sont remplis par des caractères non significatifs alignés à gauche. La mantisse est enregistrée de façon normalisée avec un chiffre avant le séparateur de décimales et <n> chiffres après. Les décimales peuvent être arrondies ou complétées par des 0 non significatifs. L'exposant commence par le mot clé "EX". Il est suivi du signe ("+" ou "-") et d'un nombre à deux ou trois chiffres.
	Exemple :
	`N10 DEF REAL REAL_VAR=-1234.5678`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.2E",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:xx-1.23EX+03" (dans l'exemple, "x" remplace un caractère non significatif).

%G :	Conversion en une chaîne de caractères comportant un nombre décimal en fonction de la plage de valeurs, au format décimal ou exponentiel. Si la valeur absolue à représenter est inférieure à 1.0EX-04 ou supérieure ou égale à 1.0EX+06, le format exponentiel est choisi, sinon c'est le format décimal. Etant donné qu'il n'est pas possible d'afficher plus de 6 caractères significatifs, le nombre est arrondi si nécessaire.
	Exemple au format décimal :
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:0.000123457".
	Exemple au format exponentiel :
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+06`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:1.23457EX+06".

%<m>G :	Conversion en une chaîne de caractères comportant un nombre décimal au format décimal ou exponentiel en fonction de la plage de valeurs (tel que %G). La longueur totale minimale de la chaîne de caractères est d'au moins <m> caractères. Les caractères manquants sont remplis par des caractères non significatifs alignés à gauche.
	Exemple au format décimal :
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:xxxx0.000123457" (dans l'exemple, "x" remplace un caractère non significatif).
	Exemple au format exponentiel :
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+06`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:xxx1.23457EX+06" (dans l'exemple, "x" remplace un caractère non significatif).

%.<n>G :	Conversion en une chaîne de caractères comportant un nombre décimal au format décimal ou exponentiel en fonction de la plage de valeurs. Etant donné qu'il n'est pas possible d'afficher plus de <n> caractères significatifs, le nombre est arrondi si nécessaire. Si la valeur absolue à représenter est inférieure à 1.0EX-04 ou supérieure ou égale à 1.0EX(+<n>), le format exponentiel est choisi, sinon c'est le format décimal.
	Exemple au format décimal :
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:0.000123".
	Exemple au format exponentiel :
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT = SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:1.23EX+03".

%<m>.<n>G :	Conversion en une chaîne de caractères comportant un nombre décimal au format décimal ou exponentiel en fonction de la plage de valeurs (tel que %<n>G). La longueur totale minimale de la chaîne de caractères est d'au moins <m> caractères. Les caractères manquants sont remplis par des caractères non significatifs alignés à gauche.
	Exemple au format décimal :
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:xxx0.0001235" (dans l'exemple, "x" remplace un caractère non significatif).
	Exemple au format exponentiel :
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF REAL_VAR:xx1.235EX+06" (dans l'exemple, "x" remplace un caractère non significatif).

%.<n>P :	Conversion d'une valeur REAL en valeur INTEGER en tenant compte du nombre de décimales <n>. La valeur entière INTEGER est transmise en sortie sous forme de nombre binaire sur 32 bits. Si la valeur à convertir ne peut pas être représentée sur 32 bits, le traitement est interrompu et une alarme est générée.
	Etant donné qu'une série d'octets produite avec l'instruction de formatage %.<n>P peut également contenir des zéros binaires, la chaîne de caractères complète obtenue ne correspond plus aux conventions du type de données CN STRING. Il ne peut donc ni être stocké dans une variable de type STRING ni être traité à l'aide des instructions de type String du langage CN. La seule utilisation possible est le transfert de paramètres à l'instruction WRITE avec sortie sur un périphérique externe correspondant (voir exemple ci-après).
	Dès que la <chaîne_format> contient une description de format de type %P, la chaîne de caractères complète est transmise en sortie, à l'exception du nombre binaire généré avec %.<n>P conformément au PM10750 $MN_SPRINT_FORMAT_P_CODE dans le code de caractères ASCII, ISO (DIN6024) ou EIA (RS244). Si un caractère non convertible est programmé, le traitement est interrompu et une alarme est générée.
	Exemple :
	`N10 DEF REAL REAL_VAR=123.45`
	`N20 DEF INT ERROR`
	`N30 DEF STRING[20] EXT_DEVICE="/ext/dev/1"`
	`...`
	`N100 EXTOPEN(ERROR,EXT_DEVICE)`
	`N110 IF ERROR <> 0`
	`...  ; Traitement des erreurs`
	`N200 WRITE(ERROR,EXT_DEVICE,SPRINT("INTEGER BINARY CODED:%.3P",REAL_VAR)`
	`N210 IF ERROR <> 0`
	`… ; Traitement des erreurs`
	Résultat : La chaîne de caractères "INTEGER BINARY CODED: 'H0001E23A'" est transmise au périphérique de sortie /ext/dev/1. La valeur hexadécimale 0x0001E23A correspond à la valeur décimale 123450.

%<m>.<n>P :	Conversion d'une valeur REAL en fonction du réglage contenu dans le paramètre machine PM10751 $MN_SPRINT_FORMAT_P_DECIMA, dans une chaîne de caractères comportant :
	• un nombre entier de <m> + <n> chiffres **ou**
	• un nombre décimal comportant au maximum <m> chiffres avant le séparateur de décimales et exactement <n> après.
	Comme pour la description de format %.<n>P la chaîne de caractères entière est stockée dans le code de caractères défini par PM10750 $MN_SPRINT_FORMAT_P_CODE.
	Conversion pour PM10751 = 0 :
	La valeur réelle est convertie en une chaîne de caractères comportant un nombre entier de <m> + <n> chiffres. Le cas échéant, les chiffres après le séparateur de décimales sont arrondis à <n> ou complétés avec des 0. Les chiffres manquants avant la virgule sont remplis avec des caractères non significatifs. Le signe moins est ajouté à gauche et un espace est inséré à la place du signe plus.
	Exemple :
	`N10 DEF REAL REAL_VAR=-123.45`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("PUNCHED TAPE FORMAT:%5.3P",REAL_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "PUNCHED TAPE FORMAT:-xx123450" (dans l'exemple, "x" remplace un caractère non significatif).
	Conversion pour PM10751 = 1 :
	La valeur REAL est convertie en une chaîne de caractères comportant un nombre décimal avec au maximum <m> chiffres avant le séparateur de décimales et exactement <n> chiffres après . Le cas échéant, les chiffres avant le séparateur de décimales sont tronqués et les chiffres après le séparateur de décimales sont arrondis ou complétés avec des 0. Si <n> est égal à 0, le signe décimal est omis.
	Exemple :
	`N10 DEF REAL REAL_VAR1=-123.45`
	`N20 DEF REAL REAL_VAR2=123.45`
	`N30 DEF STRING[80] RESULT`
	`N40 RESULT=SPRINT("PUNCHED TAPE FORMAT:%5.3P VAR2:%2.0P", REAL_VAR1,REAL_VAR2)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "PUNCHED TAPE FORMAT:-123.450 VAR2:23".

%S :	Insertion d'une chaîne de caractères.
	Exemple :
	`N10 DEF STRING[16] STRING_VAR="ABCDEFG"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%S",STRING_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF STRING_VAR:ABCDEFG".

%<m>S :	Insertion d'une chaîne de caractères comportant au moins <m> caractères. Les caractères manquants sont remplis avec des caractères non significatifs.
	Exemple :
	`N10 DEF STRING[16] STRING_VAR="ABCDEFG"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10S",STRING_VAR)`
	Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF STRING_VAR:xxxABCDEFG" (dans l'exemple, "x" remplace un caractère non significatif).

%.<n>S :	Insertion de <n> caractères d'une chaîne (à partir du premier caractère). **Exemple :** N10 DEF STRING[16] STRING_VAR="ABCDEFG" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%.3S",STRING_VAR) Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF STRING_VAR:ABC".
%<m>.<n>S :	Insertion de <n> caractères d'une chaîne (à partir du premier caractère). La longueur totale minimale de la chaîne obtenue créée est d'au moins <m> caractères. Les caractères manquants sont remplis avec des caractères non significatifs. **Exemple :** N10 DEF STRING[16] STRING_VAR="ABCDEFG" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10.5S", STRING_VAR) Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "CONTENT OF STRING_VAR:xxxxxABCDE" (dans l'exemple, "x" remplace un caractère non significatif).
%X :	Conversion d'une valeur INTEGER en un chaîne de caractères au format hexadécimal. **Exemple :** N10 DEF INT INT_VAR='HA5B8' N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("INTEGER HEXADECIMAL:%X",INT_VAR) Résultat : La variable de type String RESULT est décrite par la chaîne de caractères "INTEGER HEXADECIMAL:A5B8".

Remarque

La caractéristique du langage CN, en ce qui concerne les descripteurs et les mots-clés, qui consiste à ne pas différencier les minuscules des majuscules, s'applique également aux descriptions de format. Lors de la programmation, il n'existe pas de différence fonctionnelle entre les minuscules et les majuscules.

Possibilités de combinaison

Le tableau ci-après fournit des indications sur les types de données CN pouvant être combinés aux différentes descriptions de format. Les règles relatives à la conversion implicite de types de données s'appliquent ici (voir "Types de données (Page 432)").

	Types de données CN						
	BOOL	**CHAR**	**INT**	**REAL**	**STRING**	**AXIS**	**FRAME**
%B	+	+	+	+	+	-	-
%C	-	+	-	-	+	-	-
%D	+	+	+	+	-	-	-
%F	-	-	+	+	-	-	-
%E	-	-	+	+	-	-	-
%G	-	-	+	+	-	-	-
%S	-	+	-	-	+	-	-

	Types de données CN						
	BOOL	**CHAR**	**INT**	**REAL**	**STRING**	**AXIS**	**FRAME**
%X	+	+	+	-	-	-	-
%P	-	-	+	+	-	-	-

Remarque

Le tableau montre que les types de données CN AXIS et FRAME ne peuvent pas être utilisés directement dans la fonction SPRINT. Il est toutefois possible de :

- convertir le type de données AXIS avec la fonction AXSTRING en une chaîne de caractères, qui peut ensuite être traitée avec SPRINT ;
- lire les différentes valeurs du type de données FRAME via un accès aux composants Frame. On obtient ainsi un paramètre de type REAL qui peut être traité avec SPRINT.

3.1.5 Sauts de programme

3.1.5.1 Retour au début du programme (GOTOS)

Pour une répétition de programme, l'instruction `GOTOS` permet de retourner au début d'un programme principal ou d'un sous-programme.

Pour le retour au début du programme, les paramètres machine permettent de régler :

- la mise à "0" du temps d'exécution du programme,
- l'incrémentation du comptage de pièces de la valeur "1".

Syntaxe

```
GOTOS
```

Signification

`GOTOS` :	Instruction de saut avec destination au début du programme.	
	L'exécution est commandée par le signal d'interface CN/AP : DB21, ... DBX384.0 (commander un saut de programme)	
	Valeur :	Signification :
	0	Sans retour au début du programme. Après `GOTOS`, l'exécution du programme reprend au bloc suivant du programme pièce.
	1	Retour au début du programme et répétition du programme pièce.

Autres conditions à prendre en compte

- GOTOS déclenche un STOPRE interne (arrêt du prétraitement).
- Dans le cas d'un programme pièce avec des définitions de données (variables LUD), GOTOS exécute un retour au premier bloc de programme qui suit la section de définition, autrement dit l'exécution des définitions de données n'est pas répétée. Les variables définies gardent donc la valeur atteinte dans le bloc GOTOS et ne sont pas remises aux valeurs par défaut programmées dans la section de définition.
- L'instruction GOTOS n'est pas disponible dans les actions synchrones et les cycles technologiques.

Exemple

Code de programme	Commentaire
N10 ...	; Début du programme.
...	
N90 GOTOS	; Saut au début du programme.
...	

3.1.5.2 Sauts de programme sur repères de saut (IF, GOTOB, GOTOF, GOTO, GOTOC)

Des repères de saut (étiquettes) à rejoindre au sein d'un programme avec les instructions GOTOF, GOTOB, GOTO et GOTOC peuvent être définis dans ce programme. L'exécution du programme reprend alors à l'instruction qui suit immédiatement le repère de saut. Il est donc possible de réaliser des sauts à l'intérieur du programme.

Outre les repères de saut, les destinations de saut peuvent être également des numéros de bloc principal ou auxiliaire.

Si une condition de saut (IF ...) est formulée avant l'instruction de saut, le saut de programme n'est exécuté que si cette condition est remplie.

Syntaxe

```
GOTOB <destination de saut>
IF <condition de saut> == TRUE GOTOB <destination de saut>

GOTOF <destination de saut>
IF <condition de saut> == TRUE GOTOF <destination de saut>

GOTO <destination de saut>
IF <condition de saut> == TRUE GOTO <destination de saut>

GOTOC <destination de saut>
IF <condition de saut> == TRUE GOTOC <destination de saut>
```

Signification

`GOTOB` :	Instruction de saut avec destination de saut en direction du début du programme.	
`GOTOF` :	Instruction de saut avec destination de saut en direction de la fin du programme.	
`GOTO` :	Instruction de saut avec recherche de destination de saut. La destination est d'abord recherchée en direction de la fin du programme, puis en direction du début du programme.	
`GOTOC` :	Même effet que GOTO, sauf que l'alarme 14080 "Destination de saut pas trouvée" est inhibée. Cela signifie que l'exécution du programme n'est pas annulée en cas de recherche de destination de saut sans résultat, mais qu'elle reprend à la ligne de programme qui suit l'instruction GOTOC.	
`<destination de saut>` :	Paramètre de la destination de saut Les indications possibles sont :	
	<repère de saut> :	La destination de saut correspond au repère de saut défini dans le programme avec un nom défini par l'utilisateur : `<repère de saut>` :
	<numéro de bloc> :	La destination de saut est un numéro de bloc principal ou auxiliaire (par exemple : `200`, `N300`)
	Variable de type STRING :	Destination de saut variable. La variable correspond à un repère de saut ou à un numéro de bloc.
`IF` :	Mot clé pour formuler la condition de saut. Pour la condition de saut, vous pouvez utiliser toutes les opérations relationnelles et logiques (résultat : TRUE ou FALSE). Le saut est exécuté si le résultat de l'opération est TRUE.	

Remarque

Repères de saut (étiquettes)

Les repères de saut figurent toujours au début d'un bloc. S'il existe un numéro de programme, le repère de saut figure immédiatement après le numéro de bloc.

Les règles suivantes s'appliquent à la définition des repères de saut :

- Nombre de caractères :
  - Minimum 2
  - Maximum 32
- Caractères autorisés :
  - Caractères alphabétiques
  - Chiffres
  - Traits de soulignement
- Les deux premiers caractères doivent être obligatoirement des lettres ou des traits de soulignement.
- Le nom d'un repère de saut est suivi de deux points (":").

Autres conditions à prendre en compte

- La destination du saut doit être un bloc avec un repère de saut ou un numéro de bloc figurant **dans** le programme.
- Une instruction de saut sans condition de saut doit être programmée dans un bloc spécifique. Cette restriction ne s'applique pas aux instructions de saut avec des conditions de saut. Dans ce cas, plusieurs instructions de saut peuvent être formulées dans un bloc.
- Dans le cas de programmes comportant des instructions de saut sans condition de saut, l'instruction de fin de programme M2/M30 peut figurer avant la fin du programme.

Exemples

Exemple 1 : sauts sur repères de saut

```
Code de programme                Commentaire
N10 …
N20 GOTOF Label_1                ; Saut en direction de la fin du programme sur le
                                 ; repère de saut "Label_1".
N30 …
N40 Label_0: R1=R2+R3            ; Définition du repère de saut "Label_0".
N50 …
N60 Label_1:                     ; Définition du repère de saut "Label_1".
N70 …
N80 GOTOB Label_0                ; Saut en direction du début du programme sur le
                                 ; repère de saut "Label_0".
N90 …
```

Exemple 2 : saut indirect sur numéro de bloc

```
Code de programme                Commentaire
IF <condition> == TRUE
   R10=100                       ; Attribuer la destination de saut
ELSE
   R10=110                       ; Attribuer la destination de saut
ENDIF
; Saut en direction de la fin du programme sur le bloc dont le numéro est indiqué dans
R10
N10 GOTOF "N"<<R10
...
N90 ...
N100 ...                         ; Destination de saut
N110 ...
...
```

Exemple 3 : saut sur destination de saut variable

```
Code de programme                Commentaire
DEF STRING[20] DESTINATION
IF <condition> == TRUE
   DESTINATION = "repère1"       ; Attribuer la destination de saut
ELSE
   DESTINATION = "repère2"       ; Attribuer la destination de saut
ENDIF
; Saut en direction de la fin du programme sur la destination de saut variable "Con-
tenu de DESTINATION".
GOTOF DESTINATION
Repère1: T="foret1"              ; Destination de saut 1
...
Repère2: T="foret2"              ; Destination de saut 2
...
```

Exemple 4 : saut avec condition de saut

```
Code de programme                                        Commentaire
N40 R1=30 R2=60 R3=10 R4=11 R5=50 R6=20                  ; Affectation des valeurs initiales
N41 LA1: G0 X=R2*COS(R1)+R5 Y=R2*SIN(R1)+R6              ; Repère de saut LA1
N42 R1=R1+R3 R4=R4-1
; IF condition de saut == TRUE
; THEN saut en direction du début du programme sur le repère de saut LA1
N43 IF R4>0 GOTOB LA1
N44 M30                                                  ; Fin du programme
```

3.1.5.3 Saut de programme (CASE ... OF ... DEFAULT ...)

La fonction CASE permet de vérifier la valeur actuelle (type : INT) d'une variable ou d'une fonction de calcul et de sauter à différents endroits dans le programme en fonction du résultat obtenu.

Syntaxe

```
CASE(<expression>) OF <constante_1> GOTOF <destination de saut_1>
<constante_2> GOTOF <destination de saut_2> ... DEFAULT GOTOF
<destination de saut_n>
```

Signification

CASE :	Instruction de saut	
<expression> :	Variable ou fonction de calcul	
OF :	Mot clé pour formuler les sauts de programme conditionnels.	
<constante_1> :	Première valeur constante indiquée pour la variable ou la fonction de calcul	
	Type :	INT

<constante_2> :	Deuxième valeur constante indiquée pour la variable ou la fonction de calcul	
	Type :	INT
DEFAULT :	Au cas où la variable ou la fonction de calcul ne prend aucune des valeurs constantes indiquées, la destination du saut peut être définie par l'instruction DEFAULT. **Remarque :** Au cas où l'instruction DEFAULT n'est pas programmée, le bloc qui suit l'instruction CASE devient destination du saut.	
GOTOF :	Instruction de saut avec destination de saut en direction de la fin du programme. Au lieu de GOTOF, il est également possible de programmer toutes les autres instructions GOTO (voir chapitre "Sauts de programme sur repères de saut").	
<destination de saut_1> :	Destination de saut si la valeur de la variable ou de la fonction de calcul correspond à la première constante indiquée. La destination de saut peut être indiquée de la manière suivante :	
	<repère de saut> :	La destination de saut correspond au repère de saut défini dans le programme avec un nom défini par l'utilisateur : <repère de saut> :
	<numéro de bloc> :	La destination de saut est un numéro de bloc principal ou auxiliaire (par exemple : 200, N300)
	Variable de type STRING :	Destination de saut variable. La variable correspond à un repère de saut ou à un numéro de bloc.
<destination de saut_2> :	Destination de saut si la valeur de la variable ou de la fonction de calcul correspond à la deuxième constante indiquée.	
<destination de saut_n> :	Destination de saut si la valeur de la variable ou de la fonction de calcul ne correspond à aucune des valeurs constantes indiquées.	

Exemple

Code de programme

```
...
N20 DEF INT VAR1 VAR2 VAR3
N30 CASE(VAR1+VAR2-VAR3) OF 7 GOTOF Label_1 9 GOTOF La-
bel_2 DEFAULT GOTOF Label_3
N40 Label_1: G0 X1 Y1
N50 Label_2: G0 X2 Y2
N60 Label_3: G0 X3 Y3
...
```

L'instruction CASE de N30 définit les possibilités de saut de programme suivantes :

1. Si la valeur de la fonction de calcul VAR1+VAR2-VAR3 = 7, la destination de saut correspond au bloc avec la définition du repère de saut "Label_1" (→ N40).
2. Si la valeur de la fonction de calcul VAR1+VAR2-VAR3 = 9, la destination de saut correspond au bloc avec la définition du repère de saut "Label_2" (→ N50).
3. Si la valeur de la fonction de calcul VAR1+VAR2-VAR3 n'est ni égale à 7 ni à 9, la destination de saut correspond au bloc avec la définition du repère de saut "Label_3" (→ N60).

3.1.6 Répétition de sections de programme (REPEAT, REPEATB, ENDLABEL, P)

Cette fonction permet la répétition de sections de programme dans un ordre quelconque.

Les lignes ou sections de programme à répéter sont repérées par des repères de saut (étiquettes).

Remarque

Repères de saut (étiquettes)

Les repères de saut figurent toujours au début d'un bloc. S'il existe un numéro de programme, le repère de saut figure immédiatement après le numéro de bloc.

Les règles suivantes s'appliquent à la définition des repères de saut :

- Nombre de caractères :
  - Minimum 2
  - Maximum 32
- Caractères autorisés :
  - Caractères alphabétiques
  - Chiffres
  - Traits de soulignement
- Les deux premiers caractères doivent être obligatoirement des lettres ou des traits de soulignement.
- Le nom d'un repère de saut est suivi de deux points (":").

Syntaxe

1. répéter des lignes de programme individuelles :

```
<repère de saut> : ...
...
REPEATB <repère de saut> P=<n>
...
```

2. répéter la section de programme entre le repère de saut et l'instruction REPEAT :

```
<repère de saut> : ...
...
REPEAT <repère de saut> P=<n>
...
```

3. répéter la section entre deux repères de saut :

```
<repère de saut de début> : ...
...
<repère de saut de fin> : ...
...
```

```
REPEAT <repère de saut de début> <repère de saut de fin>
P=<n>
...
```

Remarque

Il n'est pas possible de placer l'instruction REPEAT entre les deux repères de saut. Si le <repère de saut de début> est placé devant l'instruction REPEAT et si le <repère de saut de fin> n'est pas atteint avant l'instruction REPEAT, la répétition a lieu entre le <repère de saut de début> et l'instruction REPEAT.

4. répéter la section entre le repère de saut et ENDLABEL :

```
<repère de saut> : ...
...
ENDLABEL : ...
...
REPEAT <repère de saut> P=<n>
...
```

Remarque

Il n'est pas possible de placer l'instruction REPEAT entre le <repère de saut> et ENDLABEL. Si le <repère de saut> est placé devant l'instruction REPEAT et si le ENDLABEL n'est pas atteint avant l'instruction REPEAT, la répétition a lieu entre le <repère de saut> et l'instruction REPEAT.

Signification

REPEATB :	Instruction de répétition d'une ligne de programme
REPEAT :	Instruction de répétition d'une section de programme
<repère de saut> :	Le <repère de saut> repère : • la ligne de programme à répéter (avec REPEATB) ou • le début de la section de programme à répéter (avec REPEAT)
	La ligne de programme repérée par le <repère de saut> peut se trouver avant ou après l'instruction REPEAT/REPEATB. La recherche commence en direction du début du programme. Si le repère de saut n'est pas trouvée dans cette direction, la recherche est s'effectue en direction de la fin du programme. **Exception :** si la section de programme doit être répétée entre le repère de saut et l'instruction REPEAT (voir le point 2 sous Syntaxe), la ligne de programme repérée par le <repère de saut> doit se trouver **devant** l'instruction REPEAT, car dans ce cas, la rechercher s'effectue **uniquement** en direction du début du programme. Si la ligne contenant le <repère de saut> contient d'autres instructions, celles-ci sont exécutées une nouvelle fois à chaque répétition.

ENDLABEL :	Mot-clé qui repère la fin d'une section de programme à répéter Si la ligne contenant ENDLABEL contient d'autres instructions, celles-ci sont exécutées une nouvelle fois à chaque répétition. ENDLABEL peut être utilisée plusieurs fois dans le programme.
P :	Adresse pour l'indication du nombre de répétitions
<n> :	Nombre de répétitions de la section de programme
	Type : INT
	La section de programme à répéter est répétée <n> fois. Après la dernière répétition, l'exécution du programme reprend à la ligne suivant celle qui contient REPEAT/REPEATB. **Remarque :** si P=<n> n'est pas indiqué, la section de programme a répéter est répétée exactement une fois.

Exemples

Exemple 1 : répéter des lignes de programme individuelles

```
Code de programme                    Commentaire
N10 POSITION1: X10 Y20
N20 POSITION2: CYCLE(0,,9,8)         ; Cycle de positions
N30 ...
N40 REPEATB POSITION1 P=5            ; Exécuter le bloc N10 cinq fois.
N50 REPEATB POSITION2                ; Exécuter le bloc N20 une fois.
N60 ...
N70 M30
```

Exemple 2 : répéter la section de programme entre le repère de saut et l'instruction REPEAT

```
Code de programme                Commentaire
N5 R10=15
N10 Begin: R10=R10+1             ; Côté
N20 Z=10-R10
N30 G1 X=R10 F200
N40 Y=R10
N50 X=-R10
N60 Y=-R10
N70 Z=10+R10
N80 REPEAT BEGIN P=4             ; Exécuter la section N10 à N70 quatre fois.
N90 Z10
N100 M30
```

Exemple 3 : répéter la section entre deux repères de saut

```
Code de programme                    Commentaire
N5 R10=15
N10 Begin: R10=R10+1                 ; Côté
```

```
N20 Z=10-R10
N30 G1 X=R10 F200
N40 Y=R10
N50 X=-R10
N60 Y=-R10
N70 END: Z=10
N80 Z10
N90 CYCLE(10,20,30)
N100 REPEAT BEGIN END P=3        ; Exécuter la section N10 à N70 trois fois.
N110 Z10
N120 M30
```

Exemple 4 : répéter la section entre le repère de saut et ENDLABEL

```
N10 G1 F300 Z-10
N20 BEGIN1:
N30 X10
N40 Y10
N50 BEGIN2:
N60 X20
N70 Y30
N80 ENDLABEL: Z10
N90 X0 Y0 Z0
N100 Z-10
N110 BEGIN3: X20
N120 Y30
N130 REPEAT BEGIN3 P=3       ; Exécuter la section N110 à N120 trois fois.
N140 REPEAT BEGIN2 P=2       ; Exécuter la section N50 à N80 deux fois.
N150 M100
N160 REPEAT BEGIN1 P=2       ; Exécuter la section N20 à N80 deux fois.
N170 Z10
N180 X0 Y0
N190 M30
```

Exemple 5 : réaliser un fraisage, des trous taraudés avec différentes technologies

```
N10 FORET_CENTRER()          ; Mettre en place le foret à centrer.
N20 POS_1:                   ; Réseau de positions 1
N30 X1 Y1
N40 X2
N50 Y2
N60 X3 Y3
N70 ENDLABEL :
```

Code de programme	Commentaire
N80 POS_2 :	; Réseau de positions 2
N90 X10 Y5	
N100 X9 Y-5	
N110 X3 Y3	
N120 ENDLABEL :	
N130 FORET()	; Mettre en place le foret et le cycle de perçage.
N140 TARAUD(6)	; Mettre en place le taraud M6 et le cycle de taraudage.
N150 REPEAT POS_1	; Répéter la section de programme une fois à partir de POS_1 jusqu'à ENDLABEL.
N160 FORET()	; Mettre en place le foret et le cycle de perçage.
N170 TARAUD(8)	; Mettre en place le taraud M8 et le cycle de taraudage.
N180 REPEAT POS_2	; Répéter la section de programme une fois à partir de POS_2 jusqu'à ENDLABEL.
N190 M30	

Informations complémentaires

- Des imbrications sont autorisées dans une section de programme à répéter. Chaque appel correspond à un niveau de sous-programme.
- Si M17 ou RET sont programmés pendant l'exécution de la répétition d'une section de programme, celle-ci est annulée. L'exécution du programme reprend au bloc suivant la ligne REPEAT.
- Dans l'affichage du programme en cours, la répétition de la section de programme est visualisée en tant que niveau spécifique de sous-programme.
- Si un abandon est déclenché pendant l'exécution de la section de programme, l'exécution du programme reprend après le bloc comportant l'instruction REPEAT.
  Exemple :

Code de programme	Commentaire
N5 R10=15	
N10 BEGIN: R10=R10+1	; Côté
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	; Abandon
N50 X=-R10	
N60 Y=-R10	
N70 END: Z10	
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	
N120 Z10	; Poursuivre l'exécution du programme.
N130 M30	

- Structures de contrôle et répétition de sections de programme peuvent être combinées. Il y a cependant lieu d'éviter les chevauchements. Il est recommandé que la répétition d'une section de programme se trouve à l'intérieur d'une branche de structure de contrôle ou d'une structure de contrôle se trouvant à l'intérieur d'une répétition de section de programme.
- En cas de programmation mixte de sauts et d'une répétition de section de programme, les blocs sont exécutés de façon purement séquentielle. En cas de saut à partir d'une boucle de répétition de section de programme par exemple, l'exécution se poursuit jusqu'à ce que la fin de la section de programme à répéter soit trouvée.
  Exemple :

Code de programme

```
N10 G1 F300 Z-10
N20 BEGIN1:
N30 X=10
N40 Y=10
N50 GOTOF BEGIN2
N60 ENDLABEL:
N70 BEGIN2:
N80 X20
N90 Y30
N100 ENDLABEL: Z10
N110 X0 Y0 Z0
N120 Z-10
N130 REPEAT BEGIN1 P=2
N140 Z10
N150 X0 Y0
N160 M30
```

Remarque

Il est recommandé que l'instruction `REPEAT` se trouve après les blocs de déplacement.

3.1.7 Structures de contrôle

En version standard, la commande traite les blocs CN dans l'ordre de leur programmation.

Il est possible de varier cet ordre en programmant des blocs de programme et des boucles de programme alternatifs. La programmation de ces structures de contrôle est effectuée avec les mots-clés IF, ELSE, ENDIF, LOOP, FOR, WHILE et REPEAT.

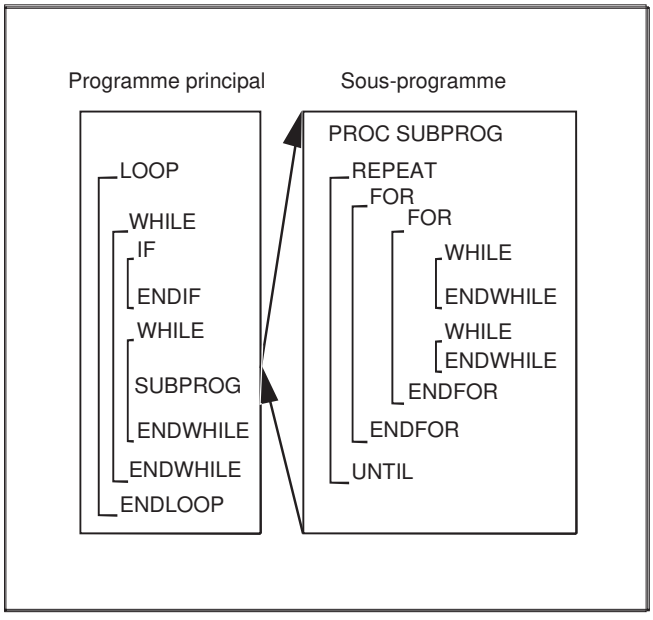
IMPORTANT
Erreur de programmation Les structures de contrôle ne sont possibles qu'à l'intérieur de la partie instructions d'un programme. Les définitions dans l'en-tête de programme ne peuvent pas être exécutées de façon conditionnelle ou répétitive. Les mots-clés pour les structures de contrôle, tout comme les destinations de sauts, ne doivent pas être utilisés dans des macros. Ceci ne fait pas l'objet d'une vérification lors de la définition des macros.

Prise d'effet

Une structure de contrôle ne peut pas être utilisée de façon globale par tous les programmes.

Niveaux d'imbrication

Une imbrication de 16 structures de contrôle au maximum est possible à l'intérieur de chaque niveau de sous-programme.



Durée d'exécution des programmes

En mode interpréteur standard, l'utilisation de sauts permet d'obtenir une exécution du programme plus rapide qu'avec les structures de contrôle.

Dans des cycles précompilés, il n'y a pas de différence entre les sauts dans le programme et les structures de contrôle.

Affichage de blocs actuel pour les boucles de programme

Si dans une boucle de programme seuls des blocs de prétraitement sont exécutés, l'affichage de blocs actuel représente le dernier bloc principal **avant** la boucle de programme.

Afin que, par exemple à des fins de diagnostic, les blocs de prétraitement exécutés deviennent également visibles dans l'affichage de blocs actuel, le décodage bloc par bloc SBL2 doit être activé.

Rectification sans bloc principal

Si dans une boucle de programme aucun bloc principal n'est programmé, la boucle est exécutée en prétraitement tant que la condition de la boucle n'est pas satisfaite.

Ceci engendre une charge de traitement élevée pouvant ralentir l'affichage.

Un remède possible est l'insertion de l'instruction STOPRE ou d'un arrêt temporisé G04 de 0 seconde **dans** la boucle.

Autres conditions à prendre en compte

- Les blocs comportant des éléments de structures de contrôle ne peuvent pas être déclarés optionnels.
- Les repères de saut (étiquettes) ne sont pas autorisés dans les blocs contenant des éléments de structures de contrôle.
- Les structures de contrôle sont exécutées de façon interprétative. En cas d'identification d'une fin de boucle, le début de la boucle est recherché compte tenu des structures de contrôle ainsi trouvées. Par conséquent, en mode interpréteur, la structure d'un programme n'est pas complètement testée.
- En principe, il est recommandé de ne pas utiliser en commun des structures de contrôle et des sauts dans le programme.
- Si des cycles subissent un prétraitement, l'imbrication correcte des structures de contrôle est vérifiée.

3.1.7.1 Instruction conditionnelle et branchement (IF, ELSE, ENDIF)

Instruction conditionnelle : IF - Bloc de programme - ENDIF

En présence d'une instruction conditionnelle, le bloc de programme entre `IF` et `ENDIF` n'est exécuté que lorsque la condition est remplie.

Branchement : IF - Bloc de programme_1 - ELSE - Bloc de programme_2 - ENDIF

En présence d'un branchement, l'un de deux blocs de programme est toujours exécuté.

Si la condition est remplie, le bloc de programme_1 entre `IF` et `ELSE` est exécuté.

Si la condition n'est **pas** remplie, le bloc de programme_2 entre ELSE et ENDIF est exécuté.

Remarque

ELSE dans les actions synchrones

Le mot-clé ELSE est également programmable dans les actions synchrones. Ainsi, une action synchrone peut être complétée par des actions devant être exécutées lorsque la condition n'est pas remplie.

Syntaxe

Instruction conditionnelle

```
IF <condition>
   Bloc de programme                    ; Exécution lorsque : <condition> == TRUE
ENDIF
```

Branchement

```
IF <condition>
   Bloc de programme_1                  ; Exécution lorsque : <condition> == TRUE
ELSE
   Bloc de programme_2                  ; Exécution lorsque : <condition> == FALSE
ENDIF
```

Signification

IF :	Début de l'instruction conditionnelle ou du branchement.
ELSE :	Début du bloc de programme alternatif.
ENDIF :	Fin de l'instruction conditionnelle ou du branchement.
<condition > :	Expression logique, dont l'évaluation a pour résultat TRUE ou FALSE.

Exemple : sous-programme de changement d'outil

Code de programme	**Commentaire**
PROC L6	Routine de changement d'outil
N500 DEF INT TNR_COURANT	Variable du numéro T actif
N510 DEF INT TNR_PRESELECTION	Variable du numéro T présélectionné
	Détermination de l'outil courant
N520 STOPRE	
N530 IF $P_ISTEST	En mode de test de programme ...
N540 TNR_COURANT = $P_TOOLNO	... lecture de l'outil "courant" du contexte du programme.
N550 ELSE	Sinon ...

Code de programme	Commentaire
N560 TNR_COURANT = $TC_MPP6[9998,1]	... lecture de l'outil de la broche.
N570 ENDIF	
N580 GETSELT(TNR_PRESELECTION)	Lecture du numéro T de l'outil présélectionné sur la broche.
N590 IF TNR_COURANT <> TNR_PRESELECTION	Si l'outil présélectionné n'est pas encore l'outil courant, ...
N600 G0 G40 G60 G90 SUPA X450 Y300 Z300 D0	... accostage de la position de changement d'outil ...
N610 M206	... et exécution du changement d'outil.
N620 ENDIF	
N630 M17	

3.1.7.2 Boucle de programme sans fin (LOOP, ENDLOOP)

La boucle infinie trouve son application dans des programmes infinis. Un retour au début de boucle a toujours lieu en fin de boucle.

Syntaxe

```
LOOP
...
ENDLOOP
```

Signification

LOOP :	Début de la boucle sans fin.
ENDLOOP :	Fin de la boucle provoquant un retour au début de la boucle.

Exemple

Code de programme
...
LOOP
MSG ("Pas de changement d'outil actif")
M0
STOPRE
ENDLOOP
...

3.1.7.3 Boucle de comptage (FOR ... TO ..., ENDFOR)

La boucle de comptage est utilisée lorsqu'une section de programme doit être répétée un certain nombre de fois.

Syntaxe

```
FOR <variable> = <valeur initiale> TO <valeur finale>
...
ENDFOR
```

Signification

FOR :	Début de la boucle de comptage.	
ENDFOR :	Fin de la boucle provoquant un retour au début de la boucle tant que la valeur finale du comptage n'est pas atteinte.	
<variable> :	Variable de comptage incrémentée de la valeur "1" à chaque exécution, de la valeur initiale à la valeur finale.	
	type	INT ou REAL **Remarque :** Le type REAL est pris en compte par exemple lorsque le paramètre R est programmé pour une boucle de comptage. Si la variable de comptage est de type REAL, sa valeur est arrondie à un nombre entier.
<valeur initiale> :	Valeur initiale du comptage Condition : La valeur initiale doit être inférieure à la valeur finale.	
<valeur finale> :	Valeur finale du comptage	

Exemples

Exemple 1 : variable INT ou paramètre R en tant que variable de comptage

Variable INT en tant que variable de comptage :

Code de programme	Commentaire
DEF INT iVARIABLE1	
R10=R12-R20*R1 R11=6	
FOR iVARIABLE1 = R10 TO R11	; Variable de comptage = variable INT
R20=R21*R22+R33	
ENDFOR	
M30	

Paramètre R en tant que variable de comptage :

Code de programme	Commentaire
R11=6	
FOR R10=R12-R20*R1 TO R11	; Variable comptage = paramètre R (variable REAL)

Code de programme	Commentaire
R20=R21*R22+R33	
ENDFOR	
M30	

Exemple 2 : fabrication d'un nombre de pièces défini

Code de programme	Commentaire
DEF INT SERIE	; Définition de la variable INT avec le nom "SERIE".
FOR SERIE = 0 TO 100	; Début de la boucle de comptage. La variable "SERIE" est incrémentée de la valeur initiale "0" à la valeur finale "100".
G01 …	
ENDFOR	; Fin de la boucle de comptage.
M30	

3.1.7.4 Boucle de programme avec condition en début de boucle (WHILE, ENDWHILE)

Dans une boucle WHILE, la condition figure au début de la boucle. La boucle WHILE est exécutée tant que la condition est remplie.

Syntaxe

```
WHILE <condition>
...
ENDWHILE
```

Signification

WHILE :	Début de la boucle de programme.
ENDWHILE :	Fin de la boucle provoquant un retour au début de la boucle.
<condition > :	Condition à remplir pour l'exécution de la boucle WHILE.

Exemple

Code de programme	Commentaire
...	
WHILE $AA_IW[AXE DE PERÇAGE] > -10	; Appel de la boucle WHILE sous la condition suivante : la consigne SCP courante de l'axe de perçage doit être supérieure à -10.
G1 G91 F250 AX[AXE DE PERÇAGE] = -1	
ENDWHILE	
...	

3.1.7.5 Boucle de programme avec condition en fin de boucle (REPEAT, UNTIL)

Dans une boucle REPEAT, la condition figure à la fin de la boucle. La boucle REPEAT est exécutée une fois, puis répétée jusqu'à ce que la condition soit remplie.

Syntaxe

```
REPEAT
...
UNTIL <condition>
```

Signification

`REPEAT :`	Début de la boucle de programme.
`UNTIL :`	Fin de la boucle provoquant un retour au début de la boucle.
`<condition >:`	Condition à remplir pour que la boucle REPEAT ne soit plus exécutée.

Exemple

Code de programme	**Commentaire**
`...`	
`REPEAT`	`; Appel de la boucle REPEAT.`
`...`	
`UNTIL ...`	`; Vérifie si la condition est remplie.`
`...`	

3.1.7.6 Exemple de programme avec des structures de contrôle imbriquées

Code de programme	**Commentaire**
`LOOP`	
`  IF NOT $P_SEARCH`	`; IF pas de recherche de bloc`
`    G1 G90 X0 Z10 F1000`	
`    WHILE $AA_IM[X] <= 100`	`; WHILE (Consigne axe X <= 100)`
`      G1 G91 X10 F500`	`; Réseau de trous`
`      Z-5 F100`	
`      Z5`	
`    ENDWHILE`	
`  ELSE`	`; ELSE recherche de bloc`
`    MSG("Pas de perçage pendant la recherche de bloc")`	
`  ENDIF`	`; ENDIF`
`  $A_OUT[1] = 1`	`; plaque à percer suivante`
`  G4 F2`	
`ENDLOOP`	

Code de programme	Commentaire
M30	

3.1.8 Coordination de programmes dans tous les canaux (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)

Un canal de la CN peut en principe exécuter son propre programme, indépendamment des autres canaux de son groupe de mode de fonctionnement (GMF). Si plusieurs programmes dans plusieurs canaux du GMF sont au même moment partie prenante dans la réalisation d'une pièce, alors le déroulement des programmes dans les différents canaux doit être coordonné à l'aide des ordres de coordination suivants.

Condition

Tous les canaux concernés par la coordination de programmes doivent appartenir au **même** groupe de mode de fonctionnement (GMF) :

PM10010 $MC_ASSIGN_CHAN_TO_MODE_GROUP[<canal>] = <numéro GMF>

Nom de canal au lieu de numéro de canal

Il est également possible d'utiliser, au lieu des numéros de canal, les noms de paramètre renseignés dans MD20000 $MC_CHAN_NAME[<indice de canal>] en tant que paramètres des procédures prédéfinies de coordination de programmes. À cet effet, l'utilisation des noms de canal doit d'abord être débloquée dans les programmes CN.

MD10280 $MN_PROG_FUNCTION_MASK, bit 1 = TRUE

Remarque

Distance minimale entre les instructions

Entre les instructions `INIT`, `START`, `WAITE`, `WAITM`, `SETM`, `CLEARM` et l'instruction `WAITMC`, un écart d'au moins deux blocs de déplacement doit être respecté. WAITMC est un bloc exécutable, mais il est déplacé pour l'optimisation dans le bloc précédent puis effacé en tant que bloc. `SETM`, à titre d'exemple, n'est pas un bloc exécutable et est déplacé dans le bloc suivant, de telle sorte qu'avec un bloc d'écart, les deux instructions se trouveraient dans le bloc central. Étant donné que seul un bloc est possible, l'optimisation n'est pas exécutée avec un bloc d'écart pour `WAITMC`.

Le programme ralentit alors et l'exécution s'arrête brièvement.

Syntaxe

```
INIT(<ChanNr>, <Prog>, <AckMode>)
START(<ChanNr>, <ChanNr>, ...)
WAITM(<MarkNr>, <ChanNr>, <ChanNr>, ...)
WAITE(<ChanNr>, <ChanNr>, ...)
WAITMC(<MarkNr>, <ChanNr>, <ChanNr>, ...)
SETM(<MarkNr>, <MarkNr>, ...)
CLEARM(<MarkNr>, <MarkNr>, ...)
```

Signification

`INIT()` :	Procédure prédéfinie de sélection du programme CN à exécuter dans le canal spécifié		
`START()` :	Procédure prédéfinie pour le démarrage du programme sélectionné dans chaque canal		
`WAITM()` :	Procédure prédéfinie pour attendre qu'un repère d'attente dans les canaux spécifiés soit atteint Dans le canal propre à l'instruction, le repère d'attente spécifié est défini par `WAITM`. Le bloc précédent est terminé par un arrêt précis. Le repère d'attente est effacé après la synchronisation. Jusqu'à 10 repères peuvent être définis dans un même canal.		
`WAITE()` :	Procédure prédéfinie pour attendre la fin du programme dans un ou plusieurs autres canaux		
`WAITMC()` :[1]	Procédure prédéfinie pour attendre qu'un repère d'attente dans les canaux spécifiés soit atteint À la différence de `WAITM`, le freinage des axes pour l'arrêt précis n'est déclenché que si les autres canaux n'ont pas encore atteint le repère d'attente.		
`SETM()` :[1]	Procédure prédéfinie pour la définition d'un ou de plusieurs repères d'attente pour la coordination de canaux L'exécution dans le canal propre n'en est pas affectée. `SETM` conserve sa validité au-delà de la remise à zéro du canal et du démarrage programme.		
`CLEARM()` :[1]	Procédure prédéfinie pour la suppression d'un ou de plusieurs repères d'attente de la coordination de canaux L'exécution dans le canal propre n'en est pas affectée. `CLEARM()` supprime tous les repères d'attente du canal. `CLEARM(0)` ne supprime que le repère d'attente "0". `CLEARM` conserve sa validité au-delà de la remise à zéro du canal et du démarrage programme.		
`<ChanNr>` :	Numéro de canal Il n'est pas nécessaire de spécifier le numéro du canal propre.		
	Type :	INT	
`<Prog>` :	Chemin d'accès absolu ou relatif (**facultatif**) + nom de programme		
	Type :	STRING	
	Pour plus d'informations sur l'indication du chemin d'accès, voir Manuel de programmation CN, chapitre "Adressage des fichiers de la mémoire de programmes" (Page 567).		
`<AckMode>` :	Mode d'acquittement (**facultatif**)		
	Type :	CHAR	
	Valeurs :	"N"	Sans acquittement L'exécution du programme se poursuit après l'envoi de l'ordre. Aucune notification n'est adressée à l'émetteur si l'ordre n'a pas pu être exécuté.
		"S"	Acquittement synchrone L'exécution du programme est suspendue tant que l'élément récepteur n'a pas acquitté l'ordre. En cas d'acquittement positif, l'instruction suivante est exécutée. En cas d'acquittement négatif, un message de défaut s'affiche.

<MarkNr> :	Numéro de la marque d'attente **Remarque** Dans un système multicanal, 100 repères d'attente sont disponibles au maximum (repère d'attente 0 ... 99). Dans un système monocanal, seul le repère d'attente 0 est disponible.
1) Pour la communication personnalisée et/ou la coordination de canaux, les repères d'attente peuvent également être utilisés via SETM / CLEARM sans l'instruction d'attente WAITMC. Dans ce cas, les repères d'attente conservent leurs valeurs au-delà de la remise à zéro du canal et du démarrage programme.	

Exemples

START à l'aide du nom de canal dans MD20000

- Paramétrage
```
MD10280 $MN_PROG_FUNCTION_MASK, bit 1 = TRUE
$MC_CHAN_NAME[ 0 ] = "USINAGE" ; nom du canal 1
$MC_CHAN_NAME[ 1 ] = "ALIMENTATION" ; nom du canal 2
```
- Programmation

Code de programme	Commentaire
START(USINAGE)	; Départ de canal 1
START(ALIMENTATION)	; Départ de canal 2

START par "noms de canaux" locaux et variable utilisateur

Code de programme	Commentaire
DEF INT MACHINE = 1	; Définition variable utilisateur pour canal 1
DEF INT CHARGEUR = 2	; Définition variable utilisateur pour canal 2
...	
START(MACHINE)	; Départ de canal 1
START(CHARGEUR)	; Départ de canal 2

START par "noms de canaux" locaux, variable utilisateur et noms de canaux paramétrés

Code de programme	Commentaire
DEF INT chanNo1	; Définition variable utilisateur pour canal 1
DEF INT chanNo2	; Définition variable utilisateur pour canal 2
chanNo1 = CHAN_1	; Affectation de noms de canaux paramétrés canal 1
chanNo2 = CHAN_2	; Affectation de noms de canaux paramétrés canal 2
...	
START(chanNo1)	; Départ de canal 1
START(chanNo2)	; Départ de canal 2

instruction INIT avec chemin d'accès absolu

Sélection de programme /_N_MPF_DIR/_N_CHARIOTAGE1_MPF dans canal 2.

Code de programme

```
INIT(2,"/_N_WKS_DIR/_N_ARBRE1_WPD/_N_CHARIOTAGE1_MPF")
```

Instruction INIT avec noms de programmes

Sélection du programme avec le nom "MYPROG". La commande recherche le programme à partir du chemin de recherche.

Code de programme

```
INIT(2,"MYPROG")
```

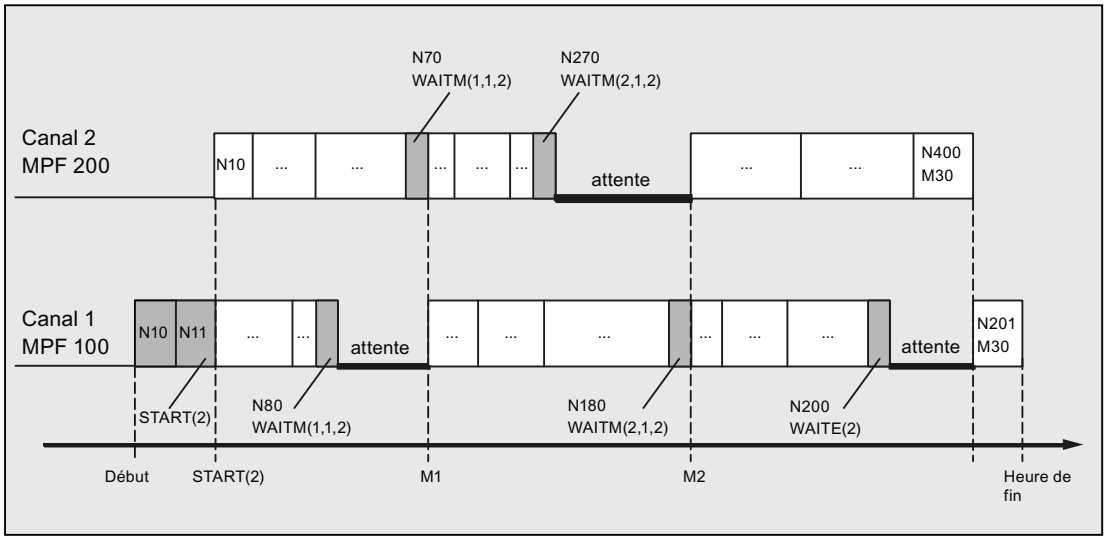
Coordination de programmes avec WAITM

- Canal 1 : Le programme /_N_MPF_DIR/_N_MPF100_MPF est déjà sélectionné et démarré.

Code de programme	**Commentaire**
	; Programme MPF100
N10 INIT(2,"MPF200","N")	; Sélection programme MPF200, canal 2
N11 START (2)	; Départ de canal 2
...	
N80 WAITM(1,1,2)	; Attente du repère WAIT 1 dans le canal 1 et 2
N81 ...	; Canal 1, N81 et canal 2, N71 sont ; démarrés de manière synchrone
...	
N180 WAITM(2,1,2)	; Attente du repère WAIT 2 dans le canal 1 et 2
N181 ...	; Canal 1, N181 et canal 2, N271 sont ; démarrés de manière synchrone
...	
N200 WAITE(2)	; Attente de la fin de programme dans le canal 2
N201 ...	; N201 n'est démarré qu'après la fin de programme de ; MPF200 dans le canal 2
N201 M30	; Fin de programme dans le canal 1

- Canal 2 : Dans le canal 1, le programme MPF200_MPF est sélectionné et démarré au moyen des blocs N10 et N20 pour le canal 2.

Code de programme	**Commentaire**
;$PATH=/_N_MPF_DIR	; Programme MPF200
...	
N70 WAITM(1,1,2)	Attente du repère WAIT 1 dans le canal 1 et 2
N71 ...	; Canal 1, N81 et canal 2, N71 sont ; démarrés de manière synchrone
...	
N270 WAITM(2,1,2)	Attente du repère WAIT 2 dans le canal 1 et 2
N271 ...	; Canal 1, N181 et canal 2, N271 sont ; démarrés de manière synchrone
...	
N400 M30	Fin de programme dans le canal 2



Autres conditions à prendre en compte

Démarrage non synchrone de l'exécution de blocs asservis selon des repères WAIT

Lors d'une coordination de canaux au moyen de repères WAIT, un démarrage non synchrone de l'exécution des blocs asservis peut se produire. Ce comportement survient lorsqu'une action est déclenchée immédiatement avant que le repère d'attente commun dans un des canaux en synchronisation ne soit atteint, ce qui a pour conséquence l'effacement de la distance restant à parcourir avec repositionnement implicite (REPOSA).

Hypothèse : Affectation d'axe actuelle dans les canaux 1 et 2

- Canal 1 : Axes X1 et U
- Canal 2 : Axe X2

Tableau 3-2 Déroulement chronologique dans les canaux 1 et 2

Canal 1	Canal 2	Description
. . .	. . .	Exécution quelconque dans le canal 1 et 2
`N100 WAITM(20,1,2)`		Canal 1 : atteint le repère `WAIT` et attend la synchronisation avec le canal 2
Début de l'exécution GETD(U) : • *Permutation d'axe* • *Effacement de la distance restant à parcourir* • *REPOSA* *Fin*	`N200 GETD(U)`	Canal 2 : exige l'axe U à partir du canal 1 Canal 1 : Exécution de `GET(U)` à l'arrière plan
	`N210 WAITM(20,1,2)`	Canal 2 : Atteint le repère `WAIT`. ⇒ La synchronisation des canaux 1 et 2 est terminée
	`N220 G0 X2=100`	Canal 2 : Début de l'exécution de N220
`N110 G0 X1=100`		Canal 1 : Début différé de l'exécution de N110

3.1.9 Macroprogrammation (DEFINE ... AS)

IMPORTANT
Les macro-instructions accroissent la complexité de la programmation. Les macro-instructions peuvent entraîner une modification importante du langage de programmation de la CN. L'emploi des macro-instructions exige une grande précaution.

Une macro-instruction réunit différentes instructions en une nouvelle instruction globale qui a son propre nom. En appelant la macro-instruction pendant l'exécution du programme, les instructions programmées dans la macro-instruction sont exécutées l'une après l'autre.

Les catégories suivantes de macros sont disponibles en fonction du domaine de validité, autrement dit du domaine dans lequel la définition d'une macro-instruction est active :

- Macros locales
  Les macros locales sont des macros définies au début d'un programme CN qui n'est pas le programme principal au moment de l'exécution. Elles sont déclarées à l'appel du programme CN et supprimées au reset de fin de programme ou au démarrage suivant de la commande. L'accès aux macros locales peut uniquement se faire à l'intérieur du programme CN dans lequel elles sont définies.
- Macros globales de programme
  Les macros globales de programme sont des macros définies au début d'un programme CN utilisé comme programme principal. Elles sont déclarées à l'appel du programme CN et supprimées au reset de fin de programme ou au démarrage suivant de la commande. L'accès aux macros globales de programme peut se faire dans le programme principal et dans tous les sous-programmes.

  Remarque

  Disponibilité des macros globales de programme

  Les macros globales de programme définies dans le programme principal ne sont également disponibles dans les sous-programmes que si le paramètre machine suivant est mis à 1 :

  MD11120 $MN_LUD_EXTENDED_SCOPE = 1

  Lorsque PM11120 = 0, les macros globales de programme définies dans le programme principal sont uniquement disponibles dans le programme principal.

- Macros globales
  Les macros globales sont des macros globales d'une CN ou d'un canal, définies dans un fichier de définition (fichier macro) et conservées même après un reset de fin de programme ou au démarrage suivant de la commande. Les macros globales peuvent être appelées et exécutées dans tout programme principal ou sous-programme.

  Remarque

  Pour pouvoir utiliser les macros d'un fichier de macros **externe** dans le programme CN, il faut charger ce fichier dans la CN.

Les macros doivent avoir été définies avant leur utilisation. Il convient de respecter les règles suivantes :

- Dans une macro, vous pouvez définir des descripteurs, des fonctions G, M, H et des noms de sous-programme L quelconques.
- La définition d'une macro peut se faire en début de programme ou dans un fichier de définition spécifique (fichier macro).
- Les macros locales et globales de programme sont définies en début de programme.
- Les macros globales doivent être définies dans un fichier macro, par ex. _N_DEF_DIR/ _N_UMAC_DEF.
- Les macros d'instruction G ne peuvent être définies qu'en tant que macros globales.
- Les fonctions H et L sont programmables à deux chiffres.
- Les instructions M et G sont programmables à trois chiffres.

Remarque

Les mots-clés et les noms réservés ne doivent pas être écrasés par des macros. Ceci est valable également pour toutes les destinations de sauts dans une instruction GOTO et pour les mots-clés dans les boucles de programmes telles que FOR, WHILE, LOOP, REPEAT.

Syntaxe

Définition d'une macro-instruction :
```
DEFINE <Macro_Name> AS <Operation_1> <Operation_2> ...
```

Appel dans le programme CN :
```
<Macro_Name>
```

Signification

`DEFINE ... AS` :	Combinaison de mots clés pour la définition d'une macro-instruction
`<Macro_Name>` :	Nom de la macro-instruction Seuls des descripteurs sont autorisés comme noms de macro-instructions. Le nom de la macro-instruction permet de l'appeler depuis le programme CN.
`<Operation_1>` :	Première instruction de programmation dans la macro-instruction
`<Operation_2>` :	Deuxième instruction de programmation dans la macro-instruction

Exemples

Exemple 1 : macro-instruction définie en début de programme

Code de programme	Commentaire
`DEFINE LIGNE AS G1 G94 F300`	`; Définition d'une macro-instruction`
`...`	
`N70 LINIE X10 Y20`	`; Appel de la macro-instruction`

Code de programme	Commentaire
...	

Exemple 2 : définitions de macro-instruction dans un fichier de macro-instructions

Code de programme	Commentaire
DEFINE M6 AS L6	; Lors du changement d'outil, un sous-programme qui se charge du transfert de données nécessaire est appelé. La fonction M de changement d'outil (par exemple M106) proprement dite est sortie dans le sous-programme.
DEFINE G81 AS DRILL(81)	; Simulation de l'instruction DIN G.
DEFINE G33 AS M333 G333	; La synchronisation avec l'AP est requise pour le filetage. L'instruction G33 initiale a été renommée en G333 par un PM, si bien que la programmation ne change pas pour l'utilisateur.

Exemple 3 : fichier externe de macro-instructions

Après le chargement du fichier externe de macro-instructions dans la commande, il est nécessaire de charger ce fichier dans la CN. Les macro-instructions ne peuvent être utilisées dans le programme CN qu'ensuite.

Code de programme	Commentaire
%_N_UMAC_DEF	
;$PATH=/_N_DEF_DIR	; Macro-instructions personnalisées
DEFINE PI AS 3.14	
DEFINE TC1 AS M3 S1000	
DEFINE M13 AS M3 M7	; Broche sens horaire, arrosage en marche
DEFINE M14 AS M4 M7	; Broche sens antihoraire, arrosage en marche
DEFINE M15 AS M5 M9	; Arrêt broche, arrêt arrosage
DEFINE M6 AS L6	; Appel du programme de changement d'outil
DEFINE G80 AS MCALL	; Désactivation du cycle de perçage
M30	

3.2 Technique des sous-programmes

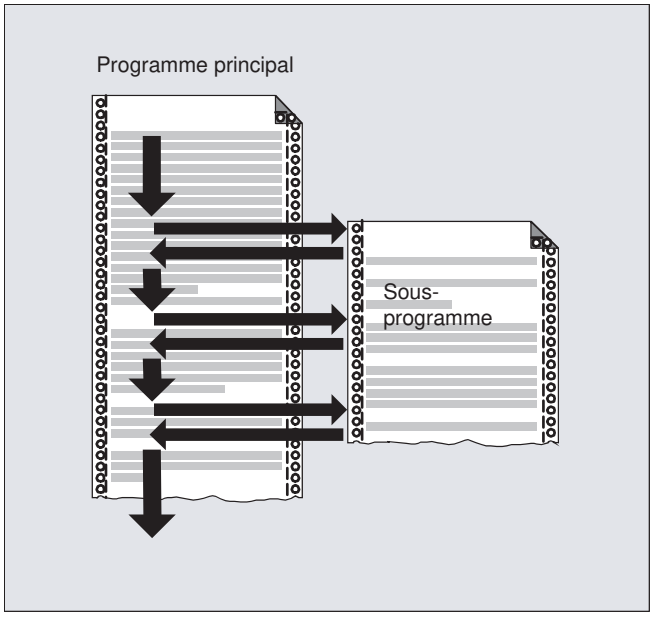
3.2.1 Généralités

3.2.1.1 Sous-programme

La désignation "Sous-programme" date encore de l'époque où les programmes pièce étaient répartis de façon déterminée en programmes principaux et sous-programmes. Les programmes principaux correspondaient aux programmes pièce qui, en vue de leur exécution, étaient sélectionnés, puis démarrés sur la commande. Les sous-programmes correspondaient aux programmes pièce qui étaient appelés par le programme principal.

Cette répartition déterminée n'a plus lieu d'être avec la langage CN SINUMERIK actuel. Chaque sous-programme peut par principe être sélectionné et démarré comme programme principal ou être appelé comme sous-programme par un autre programme pièce.

Ainsi, nous désignerons par la suite par sous-programme, un programme pièce appelé par un autre programme pièce.



Application

Comme dans tous les langages de programmation évolués, les sous-programmes dans le langage CN servent également à transférer des sections de programme, qui sont exécutées plusieurs fois, en tant que programmes complets, autonomes.

Les sous-programmes offrent les avantages suivants :

- Augmentation de la clarté et de la lisibilité des programmes
- Augmentation de la qualité grâce à la réutilisation de sections de programme testées

- Possibilité de créer des bibliothèques d'usinage spécifiques
- Economie de capacité mémoire

3.2.1.2 Nom de sous-programme

Règles de dénomination

Le nom du sous-programme peut être choisi librement dans la mesure où les règles suivantes sont respectées :

- Caractères autorisés :
  - Caractères alphabétiques : A ... Z, a ... z
  - Chiffres : 0 ... 9
  - Tiret bas : **(_)**
- Les deux premiers caractères doivent être deux lettres, ou bien un tiret bas suivi d'une lettre.

  Remarque

  Si cette condition est remplie, un programme CN peut être appelé, par simple indication du nom du programme, comme sous-programme depuis un autre programme. En revanche, si le nom du programme commence par des chiffres, l'appel en tant que sous-programme est possible uniquement avec l'instruction CALL.

- Longueur maximale : 24 caractères

Remarque

Majuscules/minuscules

Dans le langage CN SINUMERIK, **aucune** distinction n'est faite entre majuscules et minuscules.

Remarque

Noms de sous-programme interdits

Pour éviter tout problème avec les applications Windows, les noms de programme suivants ne doivent **pas** être utilisés :

- CON, PRN, AUX, NUL
- COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9
- LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8, LPT9

Extensions internes à la commande

Un préfixe et un postfixe sont ajoutés en interne à la commande comme extensions au nom de sous-programme, attribué lors la création de celui-ci :

- Préfixe : `_N_`
- Postfixe : `_SPF`

Utilisation du nom de programme

Pour l'utilisation du nom de programme, p. ex. lors d'un appel de sous-programme, toutes les combinaisons entre préfixe, nom de programme et postfixe sont possibles.

Exemple :

Le sous-programme avec le nom `SUB_PROG` peut être démarré par les descripteurs suivants :

1. `SUB_PROG`
2. `_N_SUB_PROG`
3. `SUB_PROG_SPF`
4. `_N_SUB_PROG_SPF`

Noms identiques pour programmes principaux et sous-programmes

S'il existe un programme principal (.MPF) et un sous-programme (.SPF) portant le même nom de programme, l'extension de fichier correspondante doit être indiquée à des fins d'identification univoque lors de l'utilisation du nom du programme dans le programme CN. Sinon, le programme utilisé sera le premier trouvé avec le nom indiqué dans le chemin de recherche.

Nom de fichier et nom du programme

Le nom de fichier et le nom du programme contenu dans ce fichier doivent toujours être identiques.

Un fichier contient donc toujours un programme.

3.2.1.3 Imbrication de sous-programmes

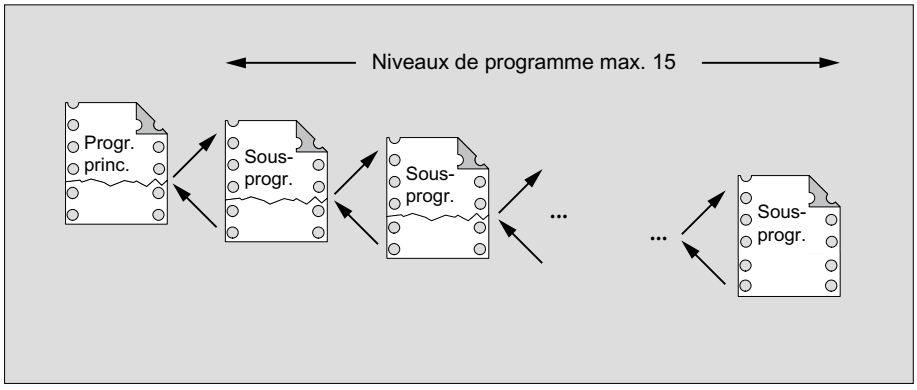
Un programme principal peut appeler des sous-programmes qui, à leur tour, appellent des sous-programmes. L'exécution des programmes est ainsi imbriquée. Chaque programme s'exécute dans un niveau de programme spécifique.

Niveaux d'imbrication

Le langage CN met actuellement à disposition 16 niveaux de programme. Le programme principal s'exécute toujours dans le niveau de programme supérieur, le niveau 0. Un sous-programme s'exécute toujours dans le niveau de programme directement inférieur consécutif à l'appel. Le niveau de programme 1 correspond ainsi au premier niveau de sous-programme.

Cloisonnement des niveaux de programme :

- Niveau de programme 0 : Niveau de programme principal
- Niveau de programme 1 - 15 : Niveau de sous-programme 1 - 15



Routines d'interruption (ASUP)

Si un sous-programme est appelé dans le cadre d'une routine d'interruption, il n'est pas traité dans le niveau de programme actuel (n) dans le canal actif, mais également dans le niveau de programme immédiatement consécutif (n+1). Afin que ceci soit également possible dans le niveau de programme le plus bas, 2 niveaux de programme supplémentaires (16 et 17) sont disponibles en ce qui concerne les routines d'interruption.

Si plus de 2 niveaux de programme sont requis, il faut en prendre compte de manière explicite dans la structuration du programme pièce exécuté dans le canal. Autrement dit, le nombre maximum de niveaux de programme utilisé doit ne doit pas faire en sorte qu'il n'en existe pas suffisamment pour l'exécution de l'interruption.

Si l'exécution de l'interruption requiert p. ex. 4 niveaux de programme, le programme pièce doit être structuré de sorte à ce qu'il occupe au maximum le niveau de programme 13. Si une interruption a ensuite lieu, les 4 niveaux de programme (14 à 17) qu'elle requiert sont disponibles.

Cycles Siemens

Les cycles Siemens requièrent 3 niveaux de programme. L'appel d'un cycle Siemens doit donc être réalisé au plus tard lors de :

- l'exécution du programme pièce : niveau de programme 12
- la routine d'interruption : niveau de programme 14

3.2.1.4 Chemin de recherche

Lors de l'appel d'un sous-programme sans indication de chemin, la commande effectue une recherche, selon un ordre prédéfini (voir "Chemin de recherche pour l'appel de sous-programmes (Page 571)"), dans la mémoire de programme existante.

3.2.1.5 Paramètres formels et paramètres effectifs

On parle de paramètres formels et de paramètres effectifs dans le contexte de la définition et de l'appel de sous-programmes avec transfert de paramètres.

Paramètres formels

Lors de la définition d'un sous-programme, les paramètres à transférer au sous programme, appelés paramètres formels, doivent être définis avec leur type et nom de paramètre.

Les paramètres formels définissent ainsi l'interface du sous-programme.

Exemple :

Code de programme	**Commentaire**
`PROC CONTOUR (REAL X, REAL Y)`	`; Paramètres formels : X et Y, tous deux du type REAL`
`N20 X1=X Y1=Y`	`; Déplacement de l'axe X1 à la position X et de l'axe Y1 à la position Y`
`...`	
`N100 RET`	

Paramètres effectifs

A l'appel d'un sous-programme, des valeurs absolues ou variables, appelés paramètres formels, doivent être transférées au sous-programme.

Les paramètres effectifs fournissent ainsi des valeurs effectives à l'interface du sous-programme.

Exemple :

Code de programme	**Commentaire**
`N10 DEF REAL LARGEUR`	`; Définition de variable`
`N20 LARGEUR=20.0`	`; Affectation de variable`
`N30 CONTOUR (5.5, LARGEUR)`	`; Appel de sous-programme avec paramètres effectifs : 5.5 et LARGEUR`
`...`	
`N100 M30`	

3.2.1.6 Transfert de paramètres

Définition d'un sous-programme avec transfert de paramètres

La définition d'un sous-programme avec transfert de paramètres s'effectue avec le mot-clé `PROC` et une liste complète de tous les paramètres attendus par le sous-programme.

Transfert incomplet de paramètres

A l'appel du sous-programme, il n'est pas toujours nécessaire de transférer de manière explicite l'ensemble des paramètres définis dans l'interface du sous-programme. Si un paramètre est omis, la valeur par défaut "0" est transférée pour ce paramètre.

Pour une identification univoque de l'ordre des paramètres, les virgules doivent cependant toujours être indiquées comme caractères de séparation des paramètres. Le dernier paramètre constitue une exception. Si celui-ci est omis à l'appel, la dernière virgule peut également l'être.

Exemple :

Sous-programme :

Code de programme	Commentaire
`PROC SUB_PROG (REAL X, REAL Y, REAL Z)`	`; Paramètres formels : X, Y et Z.`
`...`	
`N100 RET`	

Programme principal :

Code de programme	Commentaire
`PROC MAIN_PROG`	
`...`	
`N30 SUB_PROG(1.0,2.0,3.0)`	`; Appel de sous-programme avec transfert complet de paramètres :` `X=1.0, Y=2.0, Z=3.0`
…	
`N100 M30`	

Exemples d'appel de sous-programme dans `N30`, avec indication complète des paramètres :

```
N30 SUB_PROG( ,2.0,3.0)          ; X=0.0, Y=2.0, Z=3.0
N30 SUB_PROG(1.0, ,3.0)          ; X=1.0, Y=0.0, Z=3.0
N30 SUB_PROG(1.0,2.0)            ; X=1.0, Y=2.0, Z=0.0
N30 SUB_PROG( , ,3.0)            ; X=0.0, Y=0.0, Z=3.0
N30 SUB_PROG( , , )              ; X=0.0, Y=0.0, Z=0.0
```

IMPORTANT
Transfert de paramètres de type Call-by-Reference Les paramètres transférés via Call-by-Reference ne doivent pas être omis à l'appel du sous-programme.

IMPORTANT
Type de données AXIS Les paramètres du type de données AXIS ne doivent pas être omis à l'appel du sous-programme.

Vérification des paramètres de transfert :

La variable système $P_SUBPAR [n] avec n = 1, 2, ... permet de vérifier dans le sous-programme, si un paramètre a été transféré de manière explicite ou omis. L'indice n se rapporte à l'ordre des paramètres formels. L'indice n = 1 se rapporte au 1er paramètre formel, l'indice n = 2 au 2e paramètre formel, etc.

L'extrait de programme suivant illustre à titre d'exemple comment réaliser une vérification pour le 1er paramètre formel :

Programmation	**Commentaire**
PROC SUB_PROG (REAL X, REAL Y, REAL Z)	; Paramètres formels : X, Y et Z.
N20 IF $P_SUBPAR[1]==TRUE	; Vérification du 1er paramètre formel X.
...	; Ces actions sont réalisées si le paramètre formel X a été transféré de manière explicite.
N40 ELSE	
...	; Ces actions sont réalisées si le paramètre formel X n'a pas été transféré.
N60 ENDIF	
...	; Actions générales
N100 RET	

3.2.2 Définition d'un sous-programme

3.2.2.1 Sous-programme sans transfert de paramètres

Dans le cas de la définition de sous-programmes sans transfert de paramètres, la ligne de définition au début du programme n'est pas nécessaire.

Syntaxe

```
[PROC <nom de programme>]
...
```

Signification

PROC :	Instruction de définition au début d'un programme
<nom du programme> :	Nom du programme

Exemple

Exemple 1 : Sous-programme avec l'instruction PROC

Code de programme	Commentaire
PROC SUB_PROG	; Ligne de définition
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; Retour en sautant des sous-programmes

Exemple 2 : Sous-programme sans instruction PROC

Code de programme	Commentaire
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; Retour en sautant des sous-programmes

Voir aussi

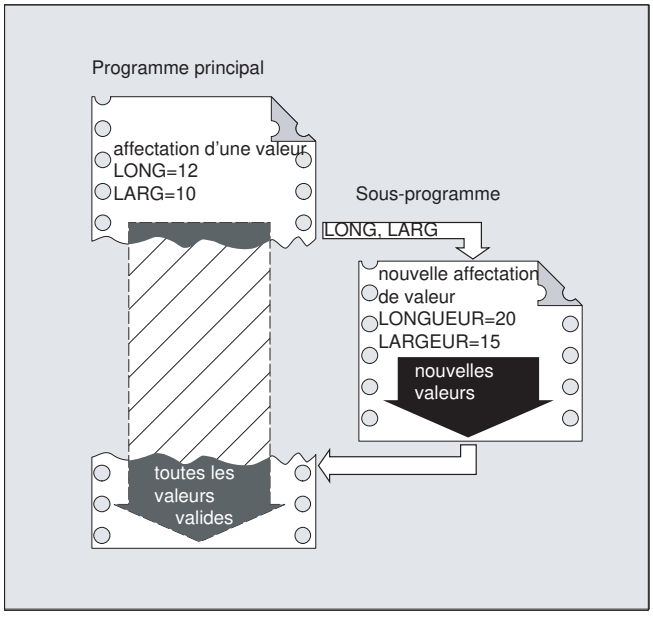
Appel de sous-programme sans transfert de paramètres (Page 534)

3.2.2.2 Sous-programme avec transfert de paramètres Call-by-Value (PROC)

La définition d'un sous-programme avec transfert de paramètres Call-by-Value s'effectue avec le mot-clé PROC, suivi du nom de programme et d'une liste complète de tous les paramètres, y compris le type et le nom. L'instruction de définition doit figurer dans la première ligne du programme.

Call-by-Value

Avec un transfert de paramètres Call-by-Value, le programme appelant transfère au sous-programme uniquement la valeur d'une variable. Ainsi, le sous-programme n'obtient pas d'accès direct à la variable. Ainsi, en cas de modification de la valeur de paramètre, seule la valeur visible dans le sous-programme est modifiée. La valeur des variables définies dans le programme appelant reste inchangée. Le transfert de paramètres Call-by-Value n'a donc aucun effet sur le programme appelant.



Syntaxe

```
PROC <nom de programme> (<type de paramètre> <nom de
paramètre>=<valeur d'initialisation>, ...)
```

Remarque

Il est possible de transférer jusqu'à 127 paramètres.

Signification

`PROC` :	Instruction de définition au début d'un programme
`<nom du programme>` :	Nom du programme
`<type de paramètre>` :	Type de données du paramètre (p. ex. REAL, INT, BOOL)
`<nom de paramètre>` :	Nom du paramètre
`<valeur d'initialisation>` :	Valeur facultative pour l'initialisation du paramètre (**facultatif**) Si aucun paramètre n'est spécifié lors de l'appel du sous-programme, la valeur d'initialisation est affectée au paramètre.

Exemples

Exemple 1

Définition d'un sous-programme SUB_PROG avec trois paramètres de type REAL et leurs valeurs par défaut :

Code de programme

```
PROC SUB_PROG(REAL LENGTH=10.0, REAL WIDTH=20.0, REAL HEIGHT=30.0)
```

Exemple 2

Différentes variantes d'appel

Code de programme

```
PROC MAIN_PROG
  REAL PAR_1 = 100
  REAL PAR_2 = 200
  REAL PAR_3 = 300
  ; variantes d'appel
  SUB_PROG
  SUB_PROG(PAR_1, PAR_2, PAR_3)
  SUB_PROG(PAR_1)
  SUB_PROG(PAR_1, , PAR_3)
  SUB_PROG( , , PAR_3)
N100 RET
```

Voir aussi

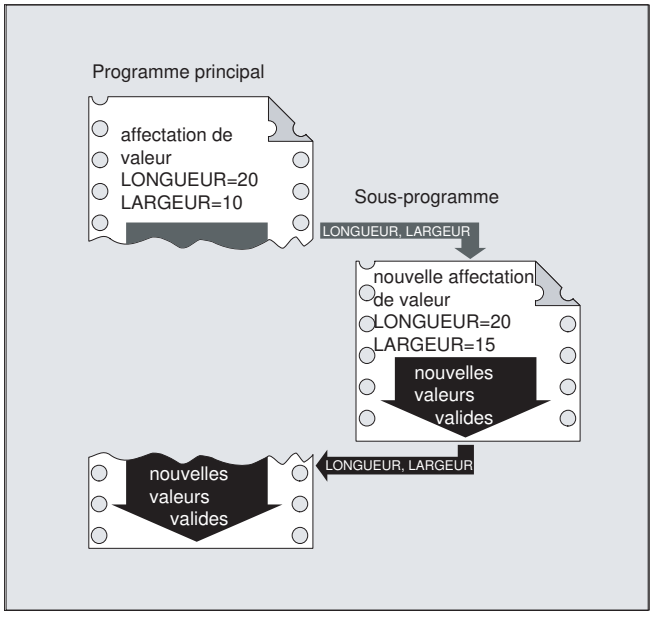
Appel d'un sous-programme avec transfert de paramètres (EXTERN) (Page 536)

3.2.2.3 Sous-programme avec transfert de paramètres Call-by-Reference (PROC, VAR)

La définition d'un sous-programme avec transfert de paramètres Call-by-Reference s'effectue avec le mot-clé `PROC`, suivi du nom de programme et d'une liste complète de tous les paramètres avec le mot-clé `VAR`, y compris le type et le nom. L'instruction de définition doit figurer dans la première ligne du programme. Des références à des tableaux peuvent également être transférées en tant que paramètres.

Call-by-Reference

Avec un transfert de paramètres Call-by-Reference, le programme appelant ne transfère pas au sous-programme la valeur d'une variable, mais une référence (pointeur) à la variable. Ainsi, le sous-programme obtient un accès direct à la variable. Ainsi, en cas de modification de la valeur de paramètre, la valeur visible dans le sous-programme est modifiée, mais aussi la valeur des variables définies dans le programme appelant. Le transfert de paramètres Call-by-Reference a donc un effet sur le programme appelant, même après la fin du sous-programme.



Remarque

Le transfert de paramètres Call-by-Reference est uniquement nécessaire si la variable transférée est définie localement dans un programme appelant (LUD). Il n'est pas nécessaire de transférer les variables globales du canal ou à la CN, car il est possible d'y accéder directement à partir du sous-programme.

Syntaxe

```
PROC <nom de programme> (VAR <type de paramètre> <nom de
paramètre>, ...)
PROC <nom de programme> (VAR <type de tableau> <nom de tableau>
[<m>,<n>,<o>], ...)
```

Remarque

Il est possible de transférer jusqu'à 127 paramètres.

Signification

PROC :	Instruction de définition au début d'un programme
VAR :	Mot-clé pour le transfert de paramètres par référence
<nom du programme> :	Nom du programme
<type de paramètre> :	Type de données du paramètre (p. ex. REAL, INT, BOOL)
<nom de paramètre> :	Nom du paramètre
<type de tableau> :	Type de données des éléments de tableau (p. ex. REAL, INT, BOOL)
<nom de tableau> :	Nom du tableau

[<m>,<n>,<o>] :	Dimension du tableau A l'heure actuelle, l'utilisation de tableaux à 3 dimensions est possible	
	<m> :	Taille du tableau dans la 1ère dimension
	<n> :	Taille du tableau dans la 2ème dimension
	<o> :	Taille du tableau dans la 3ème dimension

Remarque

- Le nom de programme indiqué après le mot-clé PROC doit correspondre avec le nom de programme attribué dans l'interface utilisateur.
- L'utilisation de tableaux de longueur indéterminée en tant que paramètres formels permet aux sous-programmes de traiter des tableaux de longueur variable. A cet effet, la longueur de la première dimension n'est p. ex. pas indiquée dans la définition d'un tableau bidimensionnel utilisé en tant que paramètre formel. Par contre, la virgule doit y figurer.
  Exemple : `PROC <nom de programme> (VAR REAL FELD[ ,5])`

Exemple

Définition d'un sous-programme avec deux paramètres comme référence au type REAL :

Code de programme

```
; Paramètre 1 : référence au type : REAL, nom : LONGUEUR
; Paramètre 2 : référence au type : REAL, nom : LARGEUR
PROC SUB_PROG(VAR REAL LONGUEUR, VAR REAL LARGEUR)
```

Voir aussi

Appel d'un sous-programme avec transfert de paramètres (EXTERN) (Page 536)

3.2.2.4 Sauvegarde des fonctions G modales (SAVE)

Avec l'attribut SAVE, les instructions G modales actives avant l'appel du sous-programme sont sauvegardées, puis réactivées à la fin de celui-ci.

IMPORTANT
Interruption du contournage Si un sous-programme avec l'attribut SAVE est appelé alors que le contournage est actif, le contournage est interrompu à la fin du sous-programme (saut de retour).

Syntaxe

```
PROC <non de sous-programme> SAVE
```

Signification

SAVE :	Sauvegarde des instructions G modales avant l'appel du sous-programme et restauration à la fin du sous-programme

Exemple

Dans le sous-programme CONTOUR, c'est l'instruction modale G91 (cote relative) qui s'applique. L'instruction G modale G90 (cote absolue) est active dans le programme principal. En raison de la définition de SAVE dans le sous-programme, G90 agit de nouveau dans le programme principal après la fin du sous-programme.

Définition de sous-programme :

Code de programme	Commentaire
PROC CONTOUR (VALEUR 1 REELLE) SAVE	; Définition de sous-programme avec paramètre SAVE
N10 G91 ...	; Instruction modale G G91 : Cote relative
N100 M17	; Fin de sous-programme

Programme principal :

Code de programme	Commentaire
N10 G0 X... Y... G90	; Instruction modale G G90 : Cote absolue
N20 ...	
...	
N50 CONTOUR (12.4)	; Appel de sous-programme
N60 X... Y...	; Instruction G modale G90 réactivée par SAVE

Autres conditions à prendre en compte

Frames

Dans le cas de sous-programmes avec l'attribut SAVE, le comportement des frames dépend du type de frame et se règle dans les paramètres machine.

3.2.2.5 Inhibition de l'exécution bloc par bloc (SBLOF, SBLON)

Même avec le traitement bloc par bloc actif, l'utilisateur peut exécuter complètement ou partiellement un programme CN sans interruption. L'inhibition du traitement bloc par bloc est activée via l'instruction SBLOF et désactivée via la commande SBLON.

Inhibition du traitement bloc par bloc pour le programme CN complet

Si la désactivation du traitement bloc par bloc (SBLOF) est programmée dans la première ligne (PROC ...) d'un **programme principal**, celle-ci est effective jusqu'à la fin ou l'interruption du programme CN. Le programme CN est alors exécuté sans interruption en mode bloc par bloc.

Si la désactivation du traitement bloc par bloc (SBLOF) est programmée dans la première ligne (PROC ...) d'un **sous-programme**, elle est effective jusqu'à la fin ou l'interruption du programme

CN. L'instruction de retour programmée permet de déterminer si l'exécution doit ou non être interrompue à la fin du sous-programme :

- Retour avec M17 : Arrêt en fin de sous-programme
- Retour avec RET : **Pas** d'arrêt en fin de sous-programme

Inhibition du traitement bloc par bloc au sein du programme CN

Si la désactivation du traitement bloc par bloc (SBLOF) est programmée dans un bloc du programme CN, le traitement bloc par bloc est désactivé à partir de ce bloc et jusqu'à l'activation programmée suivante du traitement bloc par bloc (SBLON) ou jusqu'à la fin du niveau de sous-programme actif.

Syntaxe

Inhibition du traitement bloc par bloc pour le programme CN complet :

```
PROC ... SBLOF
...
```

Inhibition du traitement bloc par bloc au sein du programme CN :

```
...
SBLOF
...
SBLON
...
```

Signification

PROC :	Première instruction d'un programme	
SBLOF :	Procédure prédéfinie de désactivation du traitement bloc par bloc	
	Uniquement dans le bloc :	oui, possible dans le bloc PROC
	Prise d'effet :	modale
SBLON :	Procédure prédéfinie pour l'activation du traitement bloc par bloc	
	Uniquement dans le bloc :	oui
	Prise d'effet :	modale

Particularités

- **Affichage de bloc en cas d'inhibition de traitement bloc par bloc**
  L'affichage du bloc en cours peut être inhibé dans les sous-programmes avec DISPLOF. Si DISPLOF est programmé avec SBLOF, l'appel du sous-programme est affiché en cas d'arrêt bloc par bloc au sein du sous-programme.
- **Inhibition du traitement bloc par bloc pour les sous-programmes asynchrones (ASUP)**
  Pour exécuter un ASUP bloc par bloc en une seule étape, il est nécessaire de programmer une instruction PROC avec SBLOF dans cet ASUP. Ceci est également valable pour la fonction "ASUP système modifiable" (MD11610 $MN_ASUP_EDITABLE).
  Si l'arrêt bloc par bloc est inhibé dans l'ASUP système ou utilisateur par la programmation de SBLOF dans la ligne PROC ou par les réglages du paramètre machine MD10702 $MN_IGNORE_SINGLEBLOCK_MASK (bit0 = 1 et bit1 = 1), la programmation de SBLON permet de réactiver l'arrêt bloc par bloc dans l'ASUP.
  Si l'arrêt bloc par bloc est inhibé dans l'ASUP utilisateur par le réglage du paramètre machine MD20117 $MC_IGNORE_SINGLEBLOCK_ASUP, il n'est **pas** possible de réactiver l'arrêt bloc par bloc dans l'ASUP en programmant SBLON.
- **Particularités des différents types de traitement bloc par bloc**
  - "SB2: bloc de calcul" ET MD10702 $MN_IGNORE_SINGLEBLOCK_MASK, bit 12 = 1:
    → Pas d'interruption dans le bloc SBLON.
  - "SB3: bloc par bloc fin" :
    → L'instruction SBLOF est inhibée**.**
- **Inhibition du traitement bloc par bloc dans les programmes imbriqués**
  Si, dans un sous-programme, SBLOF a été programmé à l'intérieur de l'instruction PROC, l'arrêt intervient lors du retour au sous-programme avec M17. Le traitement du bloc suivant dans le programme appelant est ainsi inhibé. Si une inhibition du traitement bloc par bloc a été activée dans un sous-programme avec SBLOF, sans SBLOF dans l'instruction PROC, le programme s'arrête après l'exécution du bloc machine suivant dans le programme appelant. Si ce n'est pas le résultat souhaité, il convient de reprogrammer SBLON dans le sous-programme juste avant le retour (M17). Dans le cas d'un retour dans un programme de niveau supérieur avec RET, l'exécution du programme n'est pas arrêtée.

Exemples

Exemple 1 : Inhibition du traitement bloc par bloc au sein du programme CN

Situation de départ : Le traitement bloc par bloc est activé.

Code de programme	Commentaire
N10 G1 X100 F1000	
N20 **SBLOF**	; Désactivation du traitement bloc par bloc
N30 Y20	
N40 M100	
N50 R10=90	
N60 **SBLON**	; Réactivation du traitement bloc par bloc
N70 M110	
N80 ...	

En mode bloc par bloc, la section de programme entre N20 et N60 est exécutée comme un pas.

Exemple 2 : Cycle devant apparaître comme une instruction pour l'utilisateur

Situation de départ : Le traitement bloc par bloc est activé.

Programme principal :

```
Code de programme
...
N100 G1 X10 G90 F200
N120 X-4 Y6
N130 CYCLE1
N140 G1 X0
N150 M30
```

CYCLE1 :

```
Code de programme                          Commentaire
N100 PROC CYCLE1 DISPLOF SBLOF             ; Inhibition du traitement bloc par bloc pour
                                           le programme complet
N110 R10=3*SIN(R20)+5
N120 IF (R11 <= 0)
N130  SETAL(61000)
N140 ENDIF
N150 G1 G91 Z=R10 F=R11
N160 M17
```

Le cycle CYCLE1 est exécuté lorsque le traitement bloc par bloc est actif. Cela signifie que la touche Start doit être actionnée une fois pour le traitement du CYCLE1.

Exemple 3 : Un ASUP lancé par l'AP pour l'activation de décalage d'origine et de corrections d'outil modifiés ne doit pas être visible.

```
Code de programme                          Commentaire
N100 PROC NV SBLOF DISPLOF                 ; Inhibition du traitement bloc par bloc et de
                                           l'affichage de bloc.
N110 CASE $P_UIFRNUM OF
       0 GOTOF _G500
       1 GOTOF _G54
       2 GOTOF _G55
       3 GOTOF _G56
       4 GOTOF _G57
     DEFAULT GOTOF END
N120 _G54: G54 D=$P_TOOL T=$P_TOOLNO
N130 RET
N140 _G54: G55 D=$P_TOOL T=$P_TOOLNO
N150 RET
N160 _G56: G56 D=$P_TOOL T=$P_TOOLNO
N170 RET
N180 _G57: G57 D=$P_TOOL T=$P_TOOLNO
N190 RET
```

Code de programme	Commentaire
N200 END: D=$P_TOOL T=$P_TOOLNO	
N210 RET	

Exemple 4 : Interruption ciblée dans le sous-programme

Situation de départ :

- Le traitement bloc par bloc est activé.
- MD10702 $MN_IGNORE_SINGLEBLOCK_MASK, bit12 = 1

Programme principal :

Code de programme	Commentaire
N10 G0 X0	; Arrêt bloc par bloc
N20 X10	; Arrêt bloc par bloc
N30 **CYCLE**	; Bloc de déplacement généré par le cycle.
N50 G90 X20	; Arrêt bloc par bloc
M30	

CYCLE :

Code de programme	Commentaire
PROC CYCLE **SBLOF**	; Inhibition du traitement bloc par bloc
N100 R0=1	
N110 **SBLON**	; Pas d'arrêt bloc par bloc car MD10702 bit12=1
N120 X1	; Arrêt bloc par bloc
N140 **SBLOF**	
N150 R0=2	
RET	

Exemple 5 : Inhibition du traitement bloc par bloc en cas d'imbrication de programmes

Situation de départ :

- Le traitement bloc par bloc de type 2 est actif.
- Aucune interruption ne doit se produire dans le bloc SBLON (MD10702 $MN_IGNORE_SINGLEBLOCK_MASK, bit12 = 1)

Code de programme	Commentaire
N10 X0 F1000	; Arrêt bloc par bloc
N20 UP1(0)	
PROC UP1(INT _NR) **SBLOF**	; Désactivation du traitement bloc par bloc pour UP1
N100 X10	
N110 UP2(0)	
PROC UP2(INT _NR)	
N200 X20	
N210 **SBLON**	; Activation du traitement bloc par bloc
N220 X22	; Arrêt bloc par bloc

Code de programme	Commentaire
N230 UP3(0)	
PROC UP3(INT _NR)	
N300 **SBLOF**	; Désactivation du traitement bloc par bloc
N305 X30	
N310 **SBLON**	; Activation du traitement bloc par bloc
N320 X32	; Arrêt bloc par bloc
N330 **SBLOF**	; Désactivation du traitement bloc par bloc
N340 X34	
N350 M17	; Arrêt bloc par bloc (M17)
N240 X24	; Arrêt bloc par bloc (N210)
N250 M17	; Arrêt bloc par bloc (M17)
N120 X12	
N130 M17	; Arrêt bloc par bloc (M17)
N30 X0	; Arrêt bloc par bloc
N40 M30	; Arrêt bloc par bloc

3.2.2.6 Inhibition de l'affichage du bloc courant (DISPLOF, DISPLON, ACTBLOCNO)

Le bloc courant du programme s'affiche par défaut. Dans les cycles ou sous-programmes, l'affichage du bloc courant peut être inhibé par l'instruction DISPLOF. C'est l'appel du cycle ou du sous-programme qui s'affiche alors à la place du bloc courant. L'instruction DISPLON permet d'annuler l'inhibition de l'affichage du bloc courant.

DISPLOF ou DISPLON sont programmées dans la ligne de programme avec l'instruction PROC et s'appliquent à l'ensemble du sous-programme et, implicitement, à tous les sous-programmes appelés par celui-ci et ne contenant pas d'instruction DISPLON ou DISPLOF. Ce comportement s'applique également aux ASUP.

Syntaxe

```
PROC … DISPLOF
PROC … DISPLOF ACTBLOCNO
PROC … DISPLON
```

Signification

DISPLOF :	Instruction d'inhibition de l'affichage du bloc courant.	
	Positionnement :	A la fin de la ligne de programme avec l'instruction PROC
	Prise d'effet :	Jusqu'au retour depuis le sous-programme ou la fin du programme.
	Remarque : si, avec l'instruction DISPLOF, vous appelez d'autres sous-programmes depuis le sous-programme, l'affichage du bloc courant y sera également inhibé, à condition que DISPLON n'y soit pas programmée de manière explicite.	

DISPLON :	Instruction d'annulation de l'inhibition de l'affichage du bloc courant	
	Positionnement :	A la fin de la ligne de programme avec l'instruction PROC
	Prise d'effet :	Jusqu'au retour depuis le sous-programme ou la fin du programme.
	Remarque : si, avec l'instruction DISPLON, vous appelez d'autres sous-programmes depuis le sous-programme, le bloc de programme courant s'y affiche également, à condition que DISPLOF n'y soit pas programmée de manière explicite.	
ACTBLOCNO :	Utilisé avec l'attribut ACTBLOCNO, DISPLOF a pour effet d'afficher le numéro du bloc courant dans lequel une éventuelle alarme est survenue. Ceci est également valable lorsque DISPLOF est programmé seul dans un niveau de programme inférieur. Lorsque DISPLOF est programmé sans ACTBLOCNO, c'est par contre le numéro de bloc de l'appel de cycle ou de sous-programme du dernier niveau de programme sans DISPLOF qui s'affiche.	

Exemples

Exemple 1 : inhibition de l'affichage du bloc courant dans le cycle

```
Code de programme                          Commentaire
PROC CYCLE (AXIS TOMOV, REAL POSITION)     ; Inhibition de l'affichage du bloc courant.
SAVE DISPLOF                               L'appel de cycle doit s'afficher à sa place,
                                           p. ex. : CYCLE(X,100.0)
DEF REAL DIFF                              ; Contenu du cycle
G01 ...
...
RET                                        ; Retour de sous-programme. Le bloc qui suit
                                           l'appel du cycle s'affiche.
```

Exemple 2 : affichage de bloc en cas d'émission d'une alarme

Sous-programme SUBPROG1 (avec ACTBLOCNO) :

```
Code de programme                  Commentaire
PROC SUBPROG1 DISPLOF ACTBLOC-
NO
N8000 R10 = R33 + R44
...
N9040 R10 = 66 X100                ; Déclenchement de l'alarme 12080
...
N10000 M17
```

Sous-programme SUBPROG2 (sans ACTBLOCNO) :

```
Code de programme                  Commentaire
PROC SUBPROG2 DISPLOF
```

Code de programme	Commentaire
N5000 R10 = R33 + R44	
...	
N6040 R10 = 66 X100	; Déclenchement de l'alarme 12080
...	
N7000 M17	

Programme principal :

Code de programme	Commentaire
N1000 G0 X0 Y0 Z0	
N1010 ...	
...	
N2050 SUBPROG1	; Message d'alarme = "12080 canal K1 bloc N9040 erreur de syntaxe dans texte R10="
N2060 ...	
N2350 SUBPROG2	; Message d'alarme = "12080 canal K1 bloc N2350 erreur de syntaxe dans texte R10="
...	
N3000 M30	

Exemple 3 : Annulation de l'inhibition d'affichage du bloc courant

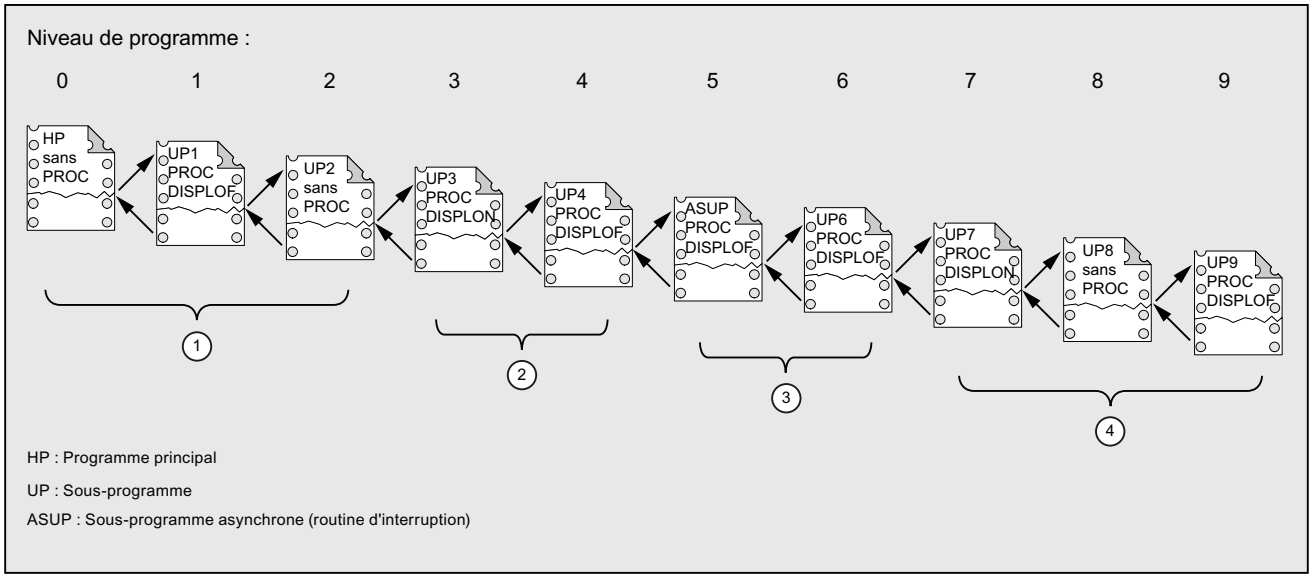
Sous-programme SUB1 avec inhibition :

Code de programme	Commentaire
PROC SUB1 DISPLOF	; Inhiber l'affichage du bloc courant dans le sous-programme SUB1. Afficher à la place le bloc avec l'appel SUB1.
...	
N300 SUB2	; Appeler le sous-programme SUB2.
...	
N500 M17	

Sous-programme SUB2 sans inhibition :

Code de programme	Commentaire
PROC SUB2 DISPLON	; Annuler l'inhibition d'affichage du bloc courant dans le sous-programme SUB2.
...	
N200 M17	; Retour au sous-programme SUB1. L'affichage du bloc courant est de nouveau inhibé dans SUB1.

Exemple 4 : comportement d'affichage pour diverses combinaisons DISPLON/DISPLOF



① L'affichage du bloc courant affiche les lignes du programme pièce du niveau 0.
② L'affichage du bloc courant affiche les lignes du programme pièce du niveau 3.
③ L'affichage du bloc courant affiche les lignes du programme pièce du niveau 3.
④ L'affichage du bloc courant affiche les lignes du programme pièce du niveau 7/8.

3.2.2.7 Identifier les sous-programmes avec prétraitement (PREPRO)

Le mot-clé `PREPRO` permet d'identifier tous les fichiers pendant le démarrage réalisé à la fin de la ligne d'instruction `PROC`.

Remarque

Ce type de préparation de la programmation dépend des réglages du paramètre machine MD10700 $MN_PREPROCESSING_LEVEL. Il convient de tenir compte des indications du constructeur de machines.

Syntaxe

```
PROC … PREPRO
```

Signification

`PREPRO` :	Mot clé pour l'identification de tous les fichiers prétraités pendant le démarrage relatifs aux programmes se trouvant dans les répertoires cycliques

Lecture de sous-programmes avec préparation et appel de sous-programme

Les répertoires de cycle sont traités dans le même ordre, aussi bien à la mise en route de sous-programmes préparés avec des paramètres, qu'à l'appel du sous-programme :

1. _N_CUS_DIR cycles utilisateurs
2. _N_CMA_DIR cycles constructeurs
3. _N_CST_DIR cycles standard

S'il existe des programmes CN de même nom avec un ciblage différent, l'instruction `PROC` trouvée en premier est activée et l'autre instruction `PROC` est ignorée sans qu'il y ait de message d'alarme.

3.2.2.8 Retour de sous-programme M17

A la fin d'un sous-programme, on trouve l'instruction de retour `M17` (ou l'instruction de fin du programme pièce `M30`). Elle entraîne le retour dans le programme appelant sur le bloc du programme pièce suivant l'appel du sous-programme.

Remarque

`M17` et `M30` sont traités de la même manière dans le langage CN.

Syntaxe

```
PROC <nom de programme>
...
M17/M30
```

Autres conditions à prendre en compte

effet du retour du sous-programme sur le mode contournage

Si `M17` (ou `M30`) figure seul dans le bloc du programme pièce, un contournage actif dans le canal est interrompu.

Pour éviter l'interruption du contournage, `M17` (ou `M30`) doit également être inscrit dans le dernier bloc de déplacement. De plus, le paramètre machine suivant doit être mis sur "0" :

PM20800 $MC_SPF_END_TO_VDI = 0 (pas de sortie M30/M17 dans l'interface CN/AP)

Exemple

1. Sous-programme avec `M17` dans un bloc spécifique

Code de programme	Commentaire
`N10 G64 F2000 G91 X10 Y10`	
`N20 X10 Z10`	
`N30 M17`	`; Retour avec interruption du contournage.`

2. Sous-programme avec M17 dans le dernier bloc

Code de programme	Commentaire
N10 G64 F2000 G91 X10 Y10	
N20 X10 Z10 M17	; Retour sans interruption du contournage.

3.2.2.9 Retour de sous-programme RET

A la place de l'instruction de M17, il est également possible d'utiliser l'instruction RET dans le sous-programme. RET doit être programmée dans un bloc spécifique du programme pièce. Comme M17, RET entraîne le retour dans le programme appelant sur le bloc du programme pièce suivant l'appel du sous-programme.

Remarque

La programmation de paramètres permet de modifier le comportement de retour de RET (voir "Retour paramétrable dans les sous-programmes (RET ...) (Page 524)").

Application

L'instruction RET doit être utilisée lorsqu'un contournage G64 (G641 ... G645) ne doit pas être interrompu par le saut de retour.

Condition

L'instruction RET peut uniquement être utilisée dans des sous-programmes n'ayant pas été définis avec l'attribut SAVE.

Syntaxe

```
PROC <nom de programme>
...
RET
```

Exemple

Programme principal :

Code de programme	Commentaire
PROC MAIN_PROGRAM	; Début du programme
...	
N50 SUB_PROG	; Appel de sous-programme : SUB_PROG
N60 ...	
...	
N100 M30	; Fin du programme

Sous-programme :

Code de programme	Commentaire
PROC SUB_PROG	
...	
N100 RET	; Le retour s'effectue sur le bloc N60 dans le programme principal.

3.2.2.10 Retour paramétrable dans les sous-programmes (RET ...)

En général, l'instruction `RET` permet de retourner dans le programme appelant depuis un sous-programme. L'exécution reprend alors à la ligne de programme qui suit l'appel du sous-programme. Si l'exécution du programme doit reprendre à un autre endroit, les possibilités suivantes sont disponibles :

- Poursuite de l'exécution du programme après l'appel des cycles de chariotage en mode dialecte ISO (après la description du contour).
- Retour dans le programme principal à partir d'un niveau de sous-programme quelconque (après un sous-programme asynchrone ASUP aussi) pour traiter une erreur.
- Retour sur plusieurs niveaux de programmes pour des applications spéciales dans des cycles de compilation et en mode dialecte ISO.

Pour cela, l'instruction `RET` doit être programmée avec d'autres paramètres.

Sens de recherche

En spécifiant le paramètre `<bloc de destination>`, le retour s'effectue d'abord au bloc suivant le bloc d'appel. Ensuite, la recherche de la destination s'effectue en direction de la **fin** du programme auquel on retourne. Si la recherche est infructueuse, elle s'effectue alors en direction du début du programme.

Syntaxe

```
RET("<bloc de destination>")
RET("<bloc de destination>",<bloc suivant le bloc de destination>)
RET("<bloc de destination>",<bloc suivant le bloc de destination>,<n
ombre de niveaux de retour>)
RET("<bloc de destination>", ,<bloc suivant le bloc de destination>)
RET("<bloc de destination>",<bloc suivant le bloc de destination>,<n
ombre de niveaux de retour>,
<retour au début du programme>)
RET(, ,<nombre de niveaux de retour>,<retour au début du programme>)
```

Signification

<table>
<tr><td>RET :</td><td colspan="3">Fin de sous-programme</td></tr>
<tr><td><Bloc de destination> :</td><td colspan="3">Le bloc où l'exécution du programme doit reprendre est indiqué dans le paramètre comme destination de saut.
Si le paramètre <nombre de niveaux de retour> n'est pas programmé, la destination de saut se trouve dans le programme depuis lequel le sous-programme courant a été appelé.
Les indications possibles sont :</td></tr>
<tr><td></td><td><Numéro de bloc></td><td colspan="2">Numéro du bloc de destination.
La recherche du numéro de bloc s'effectue d'abord en direction de la fin du programme dans le programme auquel on retourne.</td></tr>
<tr><td></td><td><Repère de saut></td><td colspan="2">Repère de saut qui doit être présent dans le programme auquel on retourne.
La recherche du repère de saut s'effectue d'abord en direction de la fin du programme dans le programme auquel on retourne.</td></tr>
<tr><td></td><td><Chaîne de caractères></td><td colspan="2">Chaîne de caractères qui doit être présente dans le programme auquel on retourne (par exemple, nom de programme ou de variable).
La recherche de la chaîne de caractères s'effectue d'abord en direction de la fin du programme dans le programme auquel on retourne.
Les règles suivantes sont valables pour la programmation de la chaîne de caractères dans le bloc de destination :
• Espace à la fin (contrairement au repère de saut identifié par un ":" à la fin).
• Seuls un numéro de bloc et/ou un repère de saut peuvent précéder la chaîne de caractères (pas d'instructions de programme).</td></tr>
<tr><td><Bloc suivant le bloc de destination> :</td><td colspan="3">Le paramètre permet d'indiquer si l'exécution du programme doit être poursuivie dans le bloc indiqué par le paramètre <bloc de destination> ou dans le bloc qui suit ce dernier.</td></tr>
<tr><td></td><td>Type :</td><td colspan="2">INT</td></tr>
<tr><td></td><td rowspan="2">Valeur :</td><td>0</td><td>Le retour s'effectue dans le bloc indiqué par le paramètre <bloc de destination>.</td></tr>
<tr><td></td><td>> 0</td><td>Le retour s'effectue dans le bloc qui suit celui indiqué par le paramètre <bloc de destination>.</td></tr>
</table>

<Nombre de niveaux de retour> :	Ce paramètre permet d'indiquer le nombre de niveaux de programme à remonter pour chercher le bloc de destination et reprendre l'exécution du programme.		
	Type :	INT	
	Valeur :	1	L'exécution du programme se poursuit au "niveau de programme courant -1" (comme RET sans paramètre).
		2	L'exécution du programme se poursuit au "niveau de programme courant -2", un niveau étant sauté.
		3	L'exécution du programme se poursuit au "niveau de programme courant -3", deux niveaux étant sautés.
		...	
	Plage de valeurs :	1 ... 15	
<Retour au début du programme> :	Ce paramètre permet d'indiquer si le programme doit reprendre au début lors d'un retour dans le programme principal lorsque le **mode de dialecte ISO** y est activé.		
	Type :	BOOL	
	Valeur :	1	Si le retour se fait dans le programme principal et si le **mode dialecte ISO** y est activé, le saut se fera au début du programme.

Remarque

Lorsqu'une chaîne de caractères est indiquée pour la recherche du bloc de destination d'un saut de retour de sous-programme, le système recherche toujours d'abord un repère de saut dans le programme appelant.

Pour qu'une chaîne de caractères définisse une destination de saut univoque, la chaîne de caractères ne doit pas être identique à un nom de repère de saut, le saut de retour de sous-programme étant sinon toujours effectué sur le repère de saut et non pas sur la chaîne de caractères (voir l'exemple 2).

Autres conditions à prendre en compte

Lors d'un retour sur plusieurs niveaux de programme, les instructions SAVE des différents niveaux sont évaluées.

Si un sous-programme modal est actif lors d'un retour sur plusieurs niveaux de programme et que l'instruction de désélection MCALL est programmée pour le sous-programme modal dans l'un des sous-programmes sautés, le sous-programme modal reste actif.

IMPORTANT

Erreur de programmation

Lors d'un retour sur plusieurs niveaux de programmes, il relève de la seule responsabilité de l'utilisateur de veiller à ce que l'exécution reprenne avec les réglages modaux requis. Ceci peut être obtenu, par exemple, par la programmation d'un bloc principal adéquat.

Exemples

Exemple 1 : retour dans le programme principal après l'exécution d'un sous-programme asynchrone ASUP

Programmation	Commentaire
N10010 CALL "UP1"	; Niveau de programme 0 (programme principal)
N11000 PROC UP1	; Niveau de programme 1
N11010 CALL "UP2"	
N12000 PROC UP2	; Niveau de programme 2
...	
N19000 PROC ASUP	; Niveau de programme 3 (exécution d'un sous-programme asynchrone ASUP)
...	
N19100 RET("N10900", ,$P_STACK)	; Retour du sous-programme au programme principal
	; $P_STACK : niveau de programme actuel
N10900	; Bloc de destination dans le programme principal
N10910 MCALL	; Désactivation de l'appel modal de sous-programme
N10920 G0 G60 G40 M5	; Initialisation des autres réglages modaux

Exemple 2 : chaîne de caractères (<chaîne de caractères>) en tant qu'indication pour la recherche du bloc de destination

Programme principal :

Code de programme	Commentaire
PROC MAIN_PROGRAM	
N1000 DEF INT iVar1=1, iVar2=4	
N1010 ...	
N1200 subProg1	; Appel du sous-programme "subProg1"
N1210 M2 S1000 X10 F1000	
N1220	
N1400 subProg2	; Appel du sous-programme "subProg2"
N1410 M3 S500 Y20	
N1420 ..	
N1500 lab1: iVar1=R10*44	
N1510 F500 X5	
N1520 ...	
N1550 subprog1: G1 X30	; "subProg1" est défini ici comme repère de saut.
N1560 ...	
N1600 subProg3	; Appel du sous-programme "subProg3"
N1610 ...	
N1900 M30	

Sous-programme subProg1 :

Code de programme	**Commentaire**
PROC subProg1	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg2")	; Retour au bloc N1400 du programme principal

Sous-programme subProg2 :

Code de programme	**Commentaire**
PROC subProg2	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("iVar1")	; Retour au bloc N1500 du programme principal

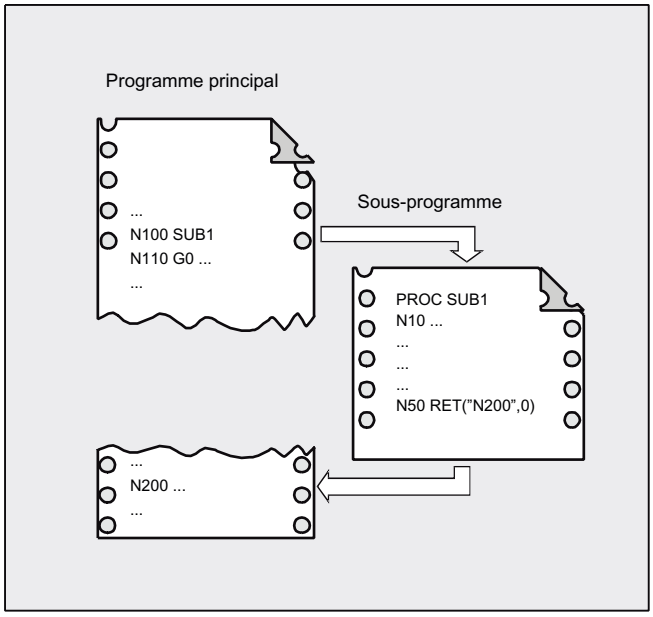
Sous-programme subProg3 :

Code de programme	**Commentaire**
PROC subProg3	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg1")	; Retour au bloc N1550 du programme principal

Informations complémentaires

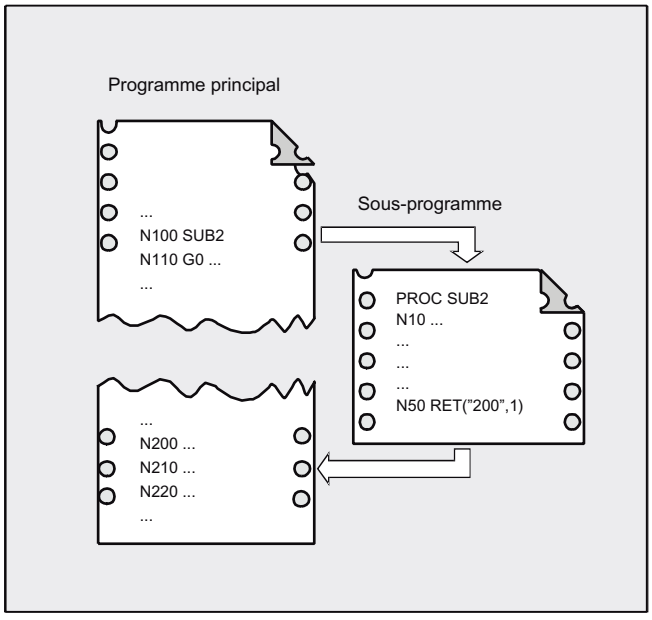
Les graphiques suivants montrent les différents effets des paramètres de renvoi.

1. <bloc de destination> = "N200", <bloc suivant le bloc de destination> = 0



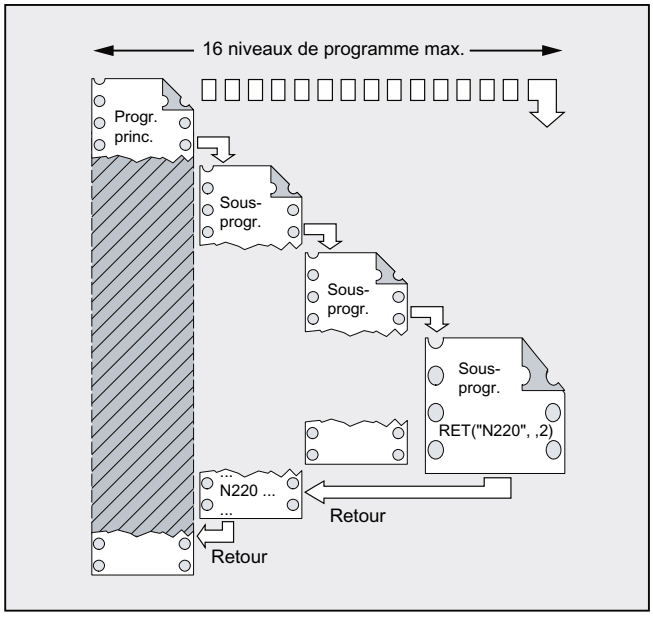
Après l'instruction `RET`, l'exécution du programme reprend au bloc `N200` du programme principal.

2. <bloc de destination> = "N200", <bloc suivant le bloc de destination> = 1



Après l'instruction `RET`, l'exécution du programme reprend au bloc (`N210`) qui suit le bloc `N200` dans programme principal.

3. <bloc de destination> = "N220", <nombre de niveaux de retour> = 2



Après l'instruction `RET`, l'exécution du programme retourne de deux niveaux de programme et reprend au bloc `N220`.

3.2.2.11 Retour paramétrable dans les sous-programmes (RETB...)

En général, l'instruction `RETB` permet de retourner dans le programme appelant depuis un sous-programme. L'exécution reprend alors à la ligne de programme qui suit l'appel du sous-programme. Si l'exécution du programme doit reprendre à un autre endroit, les possibilités suivantes sont disponibles :

- Poursuite de l'exécution du programme après l'appel des cycles de chariotage en mode dialecte ISO (après la description du contour).
- Retour dans le programme principal à partir d'un niveau de sous-programme quelconque (après un sous-programme asynchrone ASUP aussi) pour traiter une erreur.
- Retour sur plusieurs niveaux de programmes pour des applications spéciales dans des cycles de compilation et en mode dialecte ISO.

Pour cela, l'instruction `RETB` doit être programmée avec d'autres paramètres.

Sens de recherche

En spécifiant le paramètre `<bloc de destination>`, le retour s'effectue d'abord au bloc suivant le bloc d'appel. Ensuite, la recherche de la destination s'effectue en direction du **début** du programme auquel on retourne. Si la recherche est infructueuse, elle s'effectue alors en direction de la fin du programme.

Syntaxe

```
RETB("<bloc de destination>")
RETB("<bloc de destination>",<bloc suivant le bloc de destination>)
```

```
RETB("<bloc de destination>",<bloc suivant le bloc de destination>,<
nombre de niveaux de retour>)
RETB("<bloc de destination>", ,<nombre de niveaux de retour>)
RETB("<bloc de destination>",<bloc suivant le bloc de destination>,<
nombre de niveaux de retour>,
<retour au début du programme>)
RETB( , ,<nombre de niveaux de retour>,<retour au début du programme>)
```

Signification

RETB :	Fin de sous-programme		
<Bloc de destination> :	Le bloc où l'exécution du programme doit reprendre est indiqué dans le paramètre comme destination de saut. Si le paramètre <nombre de niveaux de retour> n'est pas programmé, la destination de saut se trouve dans le programme depuis lequel le sous-programme courant a été appelé. Les indications possibles sont :		
	<Numéro de bloc>	Numéro du bloc de destination. La recherche du numéro de bloc s'effectue d'abord en direction du début du programme dans le programme auquel on retourne.	
	<Repère de saut>	Repère de saut qui doit être présent dans le programme auquel on retourne. La recherche du repère de saut s'effectue d'abord en direction du début du programme dans le programme auquel on retourne.	
	<Chaîne de caractères>	Chaîne de caractères qui doit être présente dans le programme auquel on retourne (par exemple, nom de programme ou de variable). La recherche de la chaîne de caractères s'effectue d'abord en direction du début du programme dans le programme auquel on retourne. Les règles suivantes sont valables pour la programmation de la chaîne de caractères dans le bloc de destination : • **Espace à la fin** (contrairement au repère de saut identifié par un ":" à la fin). • Seuls un numéro de bloc et/ou un repère de saut peuvent **précéder la chaîne de caractères** (**pas d'instructions de programme**).	
<Bloc suivant le bloc de destination> :	Le paramètre permet d'indiquer si l'exécution du programme doit être poursuivie dans le bloc indiqué par le paramètre <bloc de destination> ou dans le bloc qui suit ce dernier.		
	Type :	INT	
	Valeur :	0	Le retour s'effectue dans le bloc indiqué par le paramètre <bloc de destination>.
		> 0	Le retour s'effectue dans le bloc qui suit celui indiqué par le paramètre <bloc de destination>.

<Nombre de niveaux de retour> :	Ce paramètre permet d'indiquer le nombre de niveaux de programme à remonter pour chercher le bloc de destination et reprendre l'exécution du programme.		
	Type :	INT	
	Valeur :	1	L'exécution du programme se poursuit au "niveau de programme courant -1" (comme RET sans paramètre).
		2	L'exécution du programme se poursuit au "niveau de programme courant -2", un niveau étant sauté.
		3	L'exécution du programme se poursuit au "niveau de programme courant -3", deux niveaux étant sautés.
		...	
	Plage de valeurs :	1 ... 15	
<Retour au début du programme> :	Ce paramètre permet d'indiquer si le programme doit reprendre au début lors d'un retour dans le programme principal lorsque le **mode de dialecte ISO** y est activé.		
	Type :	BOOL	
	Valeur :	1	Si le retour se fait dans le programme principal et si le **mode dialecte ISO** y est activé, le saut se fera au début du programme.

Remarque

Lorsqu'une chaîne de caractères est indiquée pour la recherche du bloc de destination d'un saut de retour de sous-programme, le système recherche toujours d'abord un repère de saut dans le programme appelant.

Pour qu'une chaîne de caractères définisse une destination de saut univoque, la chaîne de caractères ne doit pas être identique à un nom de repère de saut, le saut de retour de sous-programme étant sinon toujours effectué sur le repère de saut et non pas sur la chaîne de caractères (voir l'exemple 2).

Autres conditions à prendre en compte

Lors d'un retour sur plusieurs niveaux de programme, les instructions SAVE des différents niveaux sont évaluées.

Si un sous-programme modal est actif lors d'un retour sur plusieurs niveaux de programme et que l'instruction de désélection MCALL est programmée pour le sous-programme modal dans l'un des sous-programmes sautés, le sous-programme modal reste actif.

IMPORTANT
Erreur de programmation Lors d'un retour sur plusieurs niveaux de programmes, il relève de la seule responsabilité de l'utilisateur de veiller à ce que l'exécution reprenne avec les réglages modaux requis. Ceci peut être obtenu, par exemple, par la programmation d'un bloc principal adéquat.

Exemple

Code de programme	Commentaire
EXEMPLE.MPF	
…	
N3000 START_CYC(parm1, param2, …)	
N3010 TECH_CYC1(param1, param2, …)	
N3020 TECH_CYC2(param1, param2, …)	
N3030 TECH_CYC3(param1, param2, …)	
N3040 END_CYC(param1, param2, …)	
N3040 **END_CYC**(param1, param2, …)	
N3050 …	
N4500 START_CYC(param11, param12, …)	
N4510 …	
N4590 END_CYC(param11, param12, ..)	
N5000 …	
…	
N6000 M30	

Code de programme	Commentaire
PROC END_CYC(…)	; Appel dans la ligne N3040 du programme principal
N10000 …	
N15000 if status == 1	
N15010 RETB("START_CYC")	; Retour au programme appelant EXEMPLE.MPF
	; Recherche de la chaîne de caractères "START_CYC"
	; Sens de recherche : en arrière en direction du début du programme
	; Poursuite de l'exécution du programme à la ligne N3000
N15020 endif	
N15030 if status == 0	
N15040 RET	; Retour au programme appelant EXEMPLE.MPF
	; Poursuite de l'exécution du programme à la ligne N3050
N15050 endif	
N16000 RET("START_CYC")	; Retour au programme appelant EXEMPLE.MPF
	; Recherche de la chaîne de caractères "START_CYC"
	; Sens de recherche : en avant, en direction de la fin du programme
	; Poursuite de l'exécution du programme à la ligne N4500
N17060 RETB	; Retour au programme appelant EXEMPLE.MPF
	; Poursuite de l'exécution du programme à la ligne N3050
	; RETB sans paramètre est identique à RET

3.2.3 Appel d'un sous-programme

3.2.3.1 Appel de sous-programme sans transfert de paramètres

L'appel d'un sous-programme s'effectue soit avec l'adresse L et le numéro de sous-programme, soit par indication du nom du programme.

Un programme principal peut être appelé aussi en tant que sous-programme. Dans ce cas, la fin du programme `M2` ou `M30` spécifiée dans le programme principal est traitée comme `M17` (fin du programme avec retour au programme appelant).

Remarque

Inversement, un sous-programme peut également être lancé comme programme principal.

Stratégie de recherche de la commande :

*_MPF existe-t-il ?

*_SPF existe-t-il ?

Il en résulte : si le nom du sous-programme appelant est identique au nom du programme principal, c'est le programme principal qui sera rappelé. Pour éviter cet effet qu'on ne désire généralement pas, il convient de nommer sans aucune ambiguïté les programmes principaux et les sous-programmes.

Remarque

Des sous-programmes ne nécessitant pas de transfert de paramètres peuvent également être appelés à partir d'un fichier d'initialisation.

Syntaxe

```
L<numéro>/<nom de programme>
```

Remarque

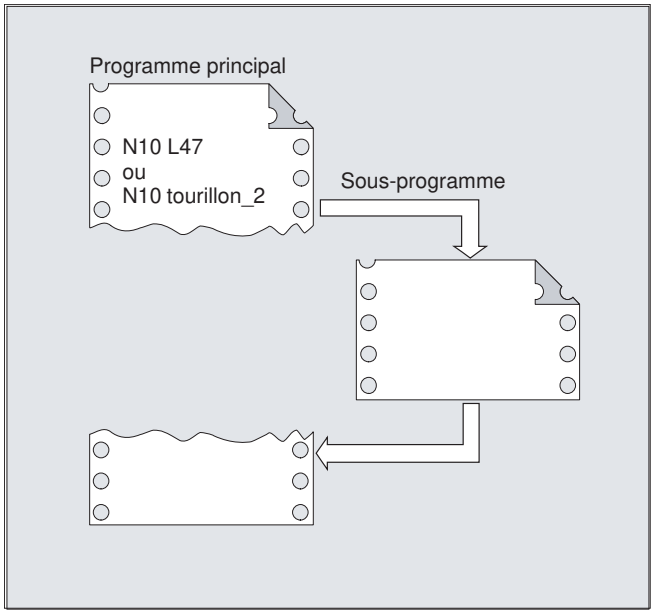
L'appel d'un sous-programme doit toujours être programmé dans le bloc CN spécifique.

Signification

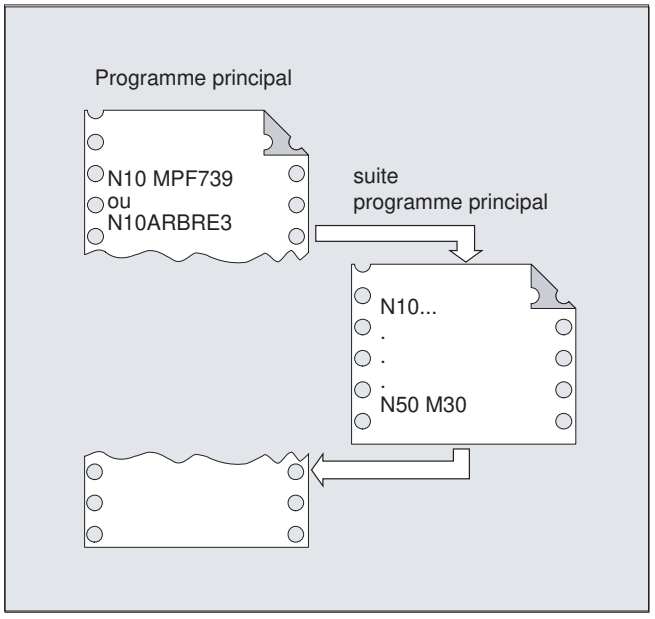
`L` :	Adresse pour l'appel du sous-programme	
`<numéro>` :	Numéro du sous-programme	
	Type :	INT
	Valeur :	Maximum 7 chiffres en notation décimale **Attention :** Les zéros en début de nom ont une signification (⇒ L123, L0123 et L00123 sont trois sous-programmes différents).
`<nom du programme>` :	Nom du sous-programme (ou programme principal)	

Exemples

Exemple 1 : appel d'un sous-programme sans transfert de paramètres



Exemple 2 : appel d'un programme principal comme sous-programme



Voir aussi

Sous-programme sans transfert de paramètres (Page 507)

3.2.3.2 Appel d'un sous-programme avec transfert de paramètres (EXTERN)

A l'appel d'un sous-programme avec transfert de paramètres, des variables ou valeurs peuvent être transférées directement (sauf pour les paramètres VAR).

Les sous-programmes avec transfert de paramètres doivent être déclarés dans le programme principal avec l'instruction EXTERN avant leur appel (par exemple en début de programme). Il faut alors indiquer le nom du sous-programme et les types de variables dans l'ordre du transfert.

IMPORTANT
Risque de confusion Les types de variables de même que l'ordre de transfert doivent correspondre aux définitions déclarées sous PROC dans le sous-programme. Les noms des paramètres peuvent être différents dans le programme principal et le sous-programme.

Syntaxe

```
EXTERN <nom de programme>(<type_Par1>,<type_Par2>,<type_Par3>)
...
<Nom de programme>(<valeur_Par1>,<valeur_Par2>,<valeur_Par3>)
```

Remarque

Un appel de sous-programme doit toujours être programmé dans un bloc CN spécifique.

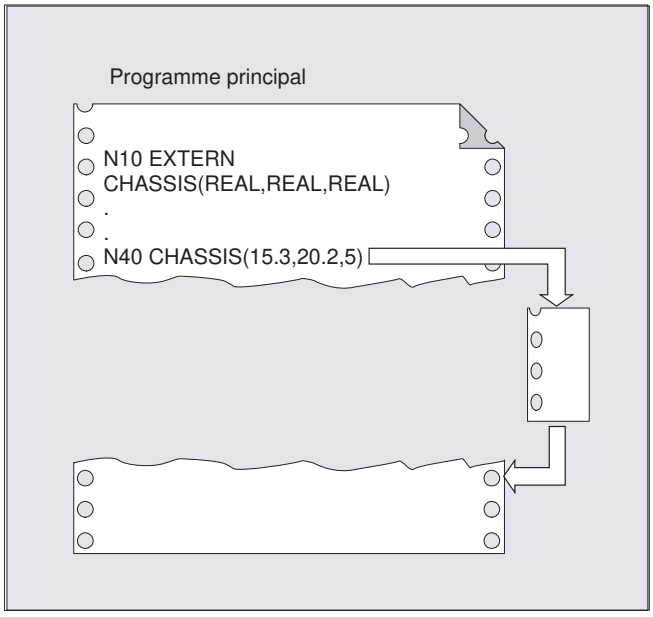
Signification

<nom du programme> :	Nom du sous-programme
EXTERN :	Mot clé pour la déclaration d'un sous-programme avec transfert de paramètres **Remarque :** L'instruction EXTERN ne doit être indiquée que si le sous-programme se trouve dans le répertoire pièce ou dans le répertoire des sous-programmes globaux. Il n'est pas nécessaire de déclarer les cycles avec l'instruction EXTERN .
<type_par1>,<type_par2>,<type_par3> :	Types de variable des paramètres à transférer, dans l'ordre du transfert
<valeur_par1>,<valeur_par2>,<valeur_par3> :	Valeurs de variable des paramètres à transférer

Exemples

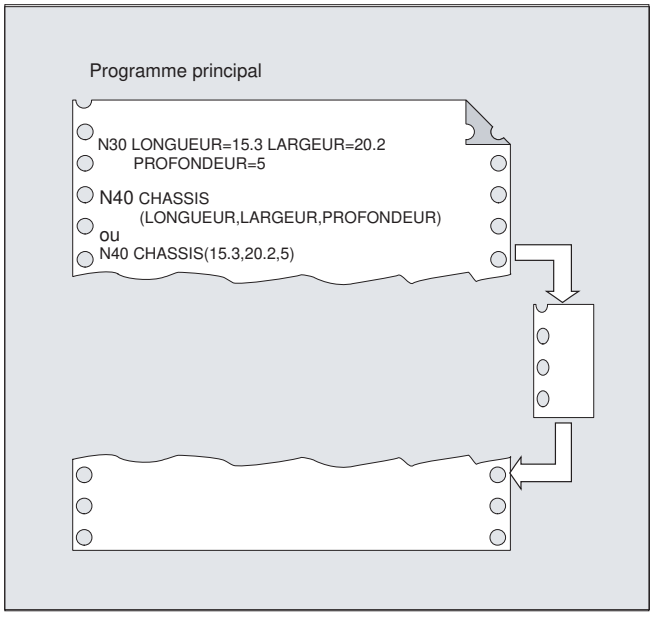
Exemple 1 : appel d'un sous-programme avec déclaration préalable

Code de programme	Commentaire
N10 EXTERN CADRE(REAL,REAL,REAL)	; Indication du sous-programme.
...	
N40 CADRE(15.3,20.2,5)	; Appel du sous-programme avec transfert de paramètres.



Exemple 2 : appel d'un sous-programme sans déclaration préalable

Code de programme	Commentaire
N10 DEF REAL LONGUEUR, LARGEUR, PROFONDEUR	
N20 …	
N30 LONGUEUR=15.3 LARGEUR=20.2 PROFONDEUR=5	
N40 CADRE(LONGUEUR,LARGEUR,PROFONDEUR)	; ou : N40 CADRE(15.3,20.2,5)



Voir aussi

Sous-programme avec transfert de paramètres Call-by-Value (PROC) (Page 508)

Sous-programme avec transfert de paramètres Call-by-Reference (PROC, VAR) (Page 510)

3.2.3.3 Nombre de répétitions de programme (P)

Si un sous-programme doit être exécuté plusieurs fois d'affilée, le nombre souhaité de répétitions peut être programmé sous l'adresse `P`, dans le bloc qui contient l'appel du sous-programme.

⚠ **PRUDENCE**
Appel de sous-programme avec répétition et transfert de paramètres Les paramètres sont transférés uniquement à l'appel du sous-programme, c'est-à-dire lors de la première exécution. Ils restent inchangés lors des répétitions. Si vous désirez que les paramètres soient modifiés lors de répétitions, vous devez définir des conventions correspondantes dans le sous-programme.

Syntaxe

```
<nom du programme> P<valeur>
```

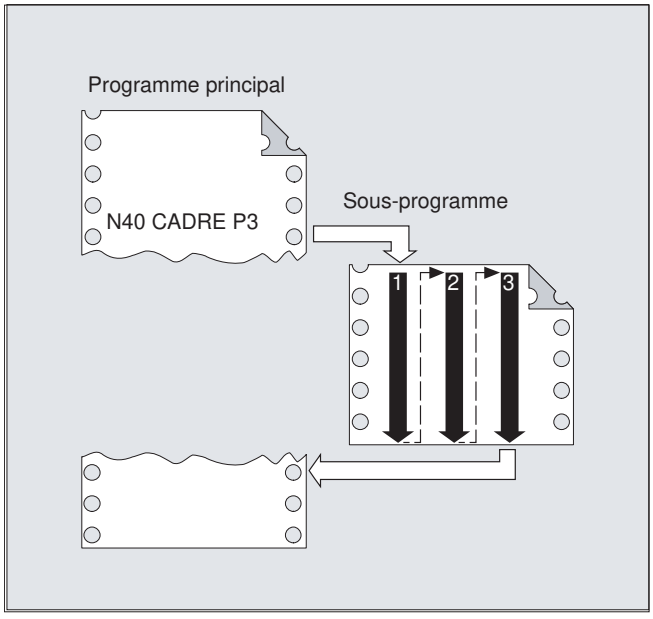
Signification

`<nom du programme>` :	appel de sous-programme
`P` :	Adresse pour la programmation des répétitions de programme

<valeur> :	Nombre de répétitions de programme	
	Type :	INT
	Plage de valeurs :	1 ... 9999 (sans signe)

Exemple

Code de programme	**Commentaire**
...	
N40 CADRE P3	; Le sous-programme CADRE doit être exécuté trois fois d'affilée.
...	



3.2.3.4 Appel modal d'un sous-programme (MCALL)

Avec l'appel modal de sous-programme `MCALL <ProgName>`, le sous-programme spécifié n'est pas immédiatement appelé. Au lieu de cela, à partir de cet instant, l'appel a lieu automatiquement dans le programme pièce après chaque bloc de déplacement comportant des déplacements avec interpolation, également sur plusieurs niveaux de programme.

Remarque

Dans une exécution de programme, c'est toujours le dernier appel modal de sous-programme `MCALL <ProgName>` qui est actif, et uniquement celui-là. L'appel modal de sous-programme actuel remplace l'appel actif jusqu'à présent.

Si les paramètres sont transmis au sous-programme, le transfert de paramètres s'effectue uniquement lors de l'appel :

```
MCALL(<ProgName>(Par1, Par2, ...))
```

Remarque

La plage de définition des données au début du sous-programme n'est parcourue qu'une fois lors de l'exécution du bloc avec l'appel MCALL programmé. Lors des appels de sous-programme suivants après les blocs de déplacement, la plage de définition des données n'est plus parcourue. Par conséquent, après le premier appel, ce n'est plus une valeur attribuée lors de la définition d'une variable utilisateur locale (LUD) qui est disponible, mais la dernière valeur écrite dans le cycle. La variable n'est pas créée à nouveau après le premier appel, mais conserve la dernière valeur de l'appel précédent. Pour contourner ce comportement, il est recommandé de séparer la définition de la LUD de son attribution de valeur, et de programmer l'attribution de valeur dans un bloc propre après la plage de définition des données.

IMPORTANT

Appels modaux de sous-programme sans déplacement avec interpolation

Dans les conditions suivantes, le sous-programme modal est appelé même sans programmation d'un déplacement avec interpolation :

- Programmation des adresses S ou F lorsque G0 ou G1 est actif.
- Lorsque G0 ou G1 est seul dans le bloc ou a été programmé en même temps que d'autres instructions G.

Syntaxe

```
MCALL <ProgName>
...
MCALL
```

Signification

MCALL <ProgName> :	Activation de la fonction "Appel modal de sous-programme"
<ProgName> :	Nom du sous-programme
MCALL :	Avec MCALL, si aucun nom de programme n'est spécifié, la fonction "Appel modal de sous-programme" est désactivée.

Autres conditions à prendre en compte

ASUP

Si l'exécution d'un programme pièce est interrompue par un ASUP (voir le chapitre "Routine d'interruption (ASUP) (Page 551)"), aucun appel modal de sous-programme n'est exécuté dans cet ASUP.

Si un ASUP est lancé à l'état de canal "Reset", il se comporte comme un programme pièce normal en ce qui concerne les appels modaux de sous-programme.

Cycle de changement d'outil

Si la fonction "Appel modal de sous-programme" est désélectionnée dans le cycle de changement d'outil, il convient de noter que le cycle de changement d'outil est appelé implicitement par l'ASUP de recherche, éventuellement aussi après une recherche de bloc, ou manuellement par écrasement en mémoire. Dans cette situation, la fonction "Appel modal de sous-programme" ne doit pas être désélectionnée, car cela fausserait le résultat de la recherche. Par conséquent, il est recommandé de programmer la désélection de la fonction "Appel modal de sous-programme" dans le cycle de changement d'outil comme suit :

```
Code de programme          Commentaire
...
  IF $AC_ASUP == 0         ; L'appel ne s'effectue pas par l'ASUP de recherche ou un
                             écrasement en mémoire.
    MCALL                  ; Désactivation de la fonction "Appel modal de sous-program-
                             me".
  ENDIF
...
```

Exemples

Exemple 1

```
Code de programme          Commentaire
N10 G0 X0 Y0
N20 MCALL L70              ; Activation de l'appel modal de sous-programme pour L70.
N30 X10 Y10                ; X10 Y10 est accosté, puis L70 est appelé.
N40 X20 Y20                ; X20 Y20 est accosté, puis L70 est appelé.
...
N100 MCALL                 ; Désactivation de la fonction "Appel modal de sous-program-
                             me".
```

Code de programme	Commentaire
N110 X0 Y0	; X0 Y0 est accosté, L70 n'est **pas** appelé.

Exemple 2

Code de programme
N10 G0 X0 Y0
N20 MCALL L70
N30 L80

Dans cet exemple, les blocs CN ci-après figurent avec des axes à interpolation programmés dans le sous-programme L80. L70 est appelé par L80.

3.2.3.5 Appel indirect de sous-programme (CALL)

Dans certaines conditions, il peut être nécessaire de faire appel à différents sous-programmes à un endroit donné du programme. Pour ce faire, indiquer le nom du sous-programme dans une variable du type STRING. L'appel du sous-programme s'effectue avec CALL et le nom de la variable.

Remarque

L'appel indirect de sous-programme ne s'adresse qu'à des sous-programmes sans transfert de paramètres. Pour l'appel direct d'un sous-programme, rangez le nom de ce dernier dans une constante de type STRING.

Syntaxe

```
CALL <nom_de_programme>
```

Signification

CALL :	Instruction pour l'appel indirect d'un sous-programme	
<nom du programme> :	Nom du sous-programme (variable ou constante)	
	Type :	STRING

Exemple

Appel direct avec constante de type STRING :

Code de programme	Commentaire
…	
CALL "/_N_WKS_DIR/_N_SUBPROG_WPD/_N_PIECE1_SPF"	; Appeler directement le sous-programme PIECE1 avec CALL.
…	

Appel indirect par le biais d'une variable :

Code de programme	Commentaire
…	
DEF STRING[100] NOM_PROG	; Définition d'une variable.
NOM_PROG="/_N_SCP_DIR/_N_SUBPROG_WPD/_N_PIECE1_SPF"	; Affectation du sous-programme PIECE1 à la variable NOM_PROG.
CALL NOM_PROG	; Appel indirect du sous-programme PIECE1 via CALL et variable NOM_PROG.
…	

3.2.3.6 Appel indirect d'un sous-programme avec indication de la section de programme à exécuter (CALL BLOCK ... TO ...)

Avec CALL et la combinaison de mots-clés BLOCK ... TO, un sous-programme est appelé indirectement et la section de programme repérée par repères de début et de fin est exécutée.

Syntaxe

```
CALL <nom de programme> BLOCK <repère de début> TO <repère de fin>
CALL BLOCK <repère de début> TO <repère de fin>
```

Signification

CALL :	Instruction pour l'appel indirect d'un sous-programme	
<nom du programme> :	Nom du sous-programme (variable ou constante) contenant la section de programme à exécuter (**indication optionnelle**).	
	Type :	STRING
	Remarque : Si aucun <nom de programme> n'est programmé, le système recherche et exécute la section de programme repérée par le <repère de début> et le <repère de fin> dans le programme actuel.	
BLOCK ... TO ... :	Combinaison de mots-clés pour l'exécution indirecte d'une section de programme	
<repère de début> :	Variable indiquant le début de la section de programme à exécuter.	
	Type :	STRING
<repère de fin> :	Variable indiquant la fin de la section de programme à exécuter.	
	Type :	STRING

Exemple

Programme principal :

Code de programme	Commentaire
...	

Code de programme	Commentaire
DEF STRING[20] STARTLABEL, ENDLABEL	; Définition des variables pour le repère de début et le repère de fin.
ETIQUETTE_DEBUT="LABEL_1"	
ENDLABEL="LABEL_2"	
...	
CALL "CONTUR_1" BLOCK STARTLABEL TO ENDLABEL	; Appel indirect d'un sous-programme et identification de la section de programme à exécuter.
...	

Sous-programme :

Code de programme	Commentaire
PROC CONTOUR_1 ...	
LABEL_1	; Repère de début : début de l'exécution de la section de programme
N1000 G1 ...	
...	
LABEL_2	; Repère de fin : fin de l'exécution de la section de programme
...	

3.2.3.7 Appel indirect d'un programme programmé en langage ISO (ISOCALL)

ISOCALL permet d'appeler de façon indirecte un programme programmé en langage ISO. L'appel active le mode ISO réglé dans les paramètres machine. Le mode d'usinage initial est rétabli à la fin du programme. En l'absence du mode ISO dans les paramètres machine, l'appel du sous-programme s'effectuera en mode Siemens.

Informations complémentaires sur le mode ISO : Description fonctionnelle Langage ISO

Syntaxe

```
ISOCALL <nom_programme>
```

Signification

ISOCALL :	Mot clé pour l'appel indirect du sous-programme activant le mode ISO réglé dans les paramètres machine
<nom du programme> :	Nom du programme programmé dans un langage ISO (variable ou constante de type STRING)

Exemple : Appel d'un contour avec programmation de cycles depuis le mode ISO

Code de programme	Commentaire
0122_SPF	; Description d'un contour en mode ISO
N1010 G1 X10 Z20	

Code de programme	Commentaire
N1020 X30 R5	
N1030 Z50 C10	
N1040 X50	
N1050 M99	
N0010 DEF STRING[5] NOM_PROG = "0122"	; Programme pièce (cycle) Siemens
...	
N2000 R11 = $AA_IW[X]	
N2010 ISOCALL NOM_PROG	
N2020 R10 = R10+1	; Exécution du programme 0122.spf en mode ISO
...	
N2400 M30	

3.2.3.8 Appel de sous-programme avec chemin d'accès et paramètres (PCALL)

PCALL permet d'appeler des sous-programmes avec chemin d'accès absolu et transfert de paramètres.

Syntaxe

```
PCALL <chemin/nom de programme>(<paramètre 1>, …,<paramètre n>)
```

Signification

PCALL :	Mot-clé pour l'appel d'un sous-programme avec chemin d'accès absolu.
<chemin/nom du programme> :	Chemin d'accès absolu, y compris nom du sous-programme. Pour les règles concernant le chemin d'accès, voir "Adressage des fichiers de la mémoire de programmes (Page 567)". Si aucune indication de chemin absolu n'est faite, PCALL se comporte comme un appel standard de sous-programme avec descripteur de programme. Le nom du programme est spécifié sans préfixe et sans extension de fichier. Si le nom du programme doit être programmé avec préfixe et extension de fichier, il convient de le déclarer de manière explicite avec préfixe et extension de fichier comme instruction EXTERN.
<paramètre 1>, ... :	Paramètres courants selon l'instruction PROC du sous-programme.

Exemple

Code de programme
PCALL/_N_WKS_DIR/_N_ARBRE_WPD/ARBRE(paramètre1,paramètre2, …)

3.2.3.9 Extension du chemin de recherche pour l'appel de sous-programmes (CALLPATH)

L'instruction `CALLPATH` permet d'étendre le chemin de recherche pour l'appel de sous-programmes. Ainsi il est également possible d'appeler des sous-programmes d'un répertoire pièce qui n'est pas sélectionné, sans indication complète du chemin d'accès absolu du sous-programme.

Une autre application est possible en mode de fonctionnement EES "EES sans GDIR", lorsqu'un répertoire sur une mémoire de programmes externe est utilisé pour stocker des sous-programmes globaux. Dans ce cas, l'instruction `CALLPATH` permet d'étendre le chemin de recherche à ce répertoire de sous-programmes.

L'extension du chemin de recherche s'effectue avant l'entrée des cycles utilisateur (_N_CUS_DIR).

Les événements suivants annulent de nouveau l'extension du chemin de recherche :

- `CALLPATH` avec espace
- `CALLPATH` sans paramètre
- Fin de programme pièce
- Reset

Syntaxe

```
CALLPATH("<chemin d'accès>")
```

Signification

`CALLPATH` :	Mot clé pour l'extension programmable du chemin de recherche. Programmation dans une ligne spécifique du programme pièce.
`<chemin d'accès>` :	Constante ou variable de type STRING Contient le chemin d'accès absolu du répertoire pour l'extension du chemin de recherche. Pour les règles concernant le chemin d'accès, voir "Adressage des fichiers de la mémoire de programmes (Page 567)".

Exemple

Le chemin de recherche doit être étendu à un répertoire pièce donné :

Code de programme

```
...
CALLPATH("/_N_WKS_DIR/_N_MYWPD_WPD")
...
```

Le réglage du chemin de recherche est alors le suivant (le point 5 est nouveau) :

1. Répertoire actuel/*nom*
2. Répertoire actuel/*nom_SPF*
3. Répertoire actuel/*nom_MPF*
4. //NC:/_N_SPF_DIR / *nom_SPF*
5. **/_N_WKS_DIR/_N_MYWPD_WPD/nom_SPF**
6. /N_CUS_DIR/*nom_SPF*
7. /_N_CMA_DIR/*nom_SPF*
8. /_N_CST_DIR/*nom_SPF*

Autres conditions à prendre en compte

- `CALLPATH` vérifie l'existence réelle du chemin d'accès programmé. En cas d'erreur, l'exécution du programme pièce est annulée avec l'alarme de correcteur 14009.
- `CALLPATH` peut également être programmé dans les fichiers INI. L'instruction est alors active pendant la durée d'exécution du fichier INI (fichier WPD-INI ou programme d'initialisation de données actives CN, par exemple frames dans le 1er canal _N_CH1_UFR_INI). Puis le chemin de recherche est de nouveau annulé.

3.2.3.10 Exécution de sous-programme externe (EXTCALL)

L'instruction `EXTCALL` permet de recharger un programme pièce à partir d'une mémoire externe et de l'exécuter.

Les mémoires externes disponibles sont les suivantes :

- Lecteur local
- Lecteur réseau
- Lecteur USB

Remarque

Seules les interfaces USB sur le pupitre opérateur ou TCU doivent être utilisées pour l'exécution d'un programme externe depuis un lecteur USB.

IMPORTANT
Endommagement de l'outil / de la pièce par une clé USB Pour exécuter un sous-programme externe, il est recommandé de ne pas utiliser de clé USB. En effet, toute coupure de communication avec la clé USB pendant l'exécution du programme pièce, résultant de mauvais contacts, d'une déconnexion ou d'un retrait accidentel pendant le fonctionnement, entraîne l'arrêt immédiat de l'usinage. Ceci pourrait endommager l'outil et/ou la pièce.

Préréglage du chemin vers le programme externe

Le chemin d'accès au répertoire de programmes externe peut être préréglé par la donnée de réglage :

SD42700 $SC_EXT_PROG_PATH

Avec le chemin vers le programme et le descripteur de programme spécifiés lors de l'appel `EXTCALL`, on obtient le chemin global du programme pièce à appeler.

Remarque

Si le chemin vers le programme doit uniquement être spécifié par l'appel `EXTCALL`, SD42700 doit être vide !

Remarque

Paramètres

Lors de l'appel d'un programme externe, aucun paramètre ne peut être transmis à celui-ci.

Syntaxe

```
EXTCALL("<chemin/><nom du programme>")
```

Signification

`EXTCALL` :	Instruction d'appel d'un sous-programme externe	
`''<chemin/><nom du programme>''` :	Constante/variable de type STRING	
	`<chemin/>` :	Chemin d'accès absolu ou relatif (**facultatif**)

	`<nom du programme>` :	le nom du programme est spécifié sans le préfixe "_N_".
		L'extension de fichier ("MPF", "SPF") peut être ajoutée au nom du programme avec le caractère "_" ou "." (**facultatif**).
		Exemple :
		`"ARBRE"`
		`"ARBRE_SPF"`
		`"ARBRE.SPF"`

Chemin d'accès : désignations abrégées

Les désignations abrégées ci-après sont utilisables pour le chemin d'accès :

- Lecteur local : "**LOCAL_DRIVE:**"
- Carte CF : "**CF_CARD:**"
- Lecteur USB (pupitre opérateur) : "**USB:**"

L'une ou l'autre des désignations abrégées "CF_CARD:" et "LOCAL_DRIVE:" peut être utilisée.

Exemple

Exécution depuis le lecteur local

Le programme principal "MAIN.MPF" se trouve dans la mémoire CN et est sélectionné pour être exécuté.

Sous-programme "SP_1"

Le sous-programme externe "SP_1.SPF" ou "SP_1.MPF" se trouve sur le lecteur local dans le répertoire "/user/sinumerik/data/prog/WKS.DIR/WST1.WPD".

Le chemin d'accès au répertoire du programme externe doit être réglé comme suit :

SD42700 $SC_EXT_PROG_PATH = LOCAL_DRIVE:WKS.DIR/WST1.WPD

Remarque

Chemin d'accès pour l'appel d'un sous-programme externe :

- Sans utiliser le préréglage : "LOCAL_DRIVE:WKS.DIR/WST1.WPD/SP_1"
- En utilisant le préréglage : "SP_1"

Sous-programme "SP_2"

Le sous-programme externe "SP_2.SPF" ou "SP_2.MPF" se trouve dans le répertoire WKS.DIR/WST1.WPD du lecteur USB. Le préréglage du chemin d'accès au répertoire de programmes externe est utilisé pour le chemin vers le sous-programme "SP_1" et n'est pas repris dans le programme principal. Le chemin complet doit donc être indiqué lors de l'appel du sous-programme "SP_2".

Programme principal "MAIN"

```
Code de programme
N010 PROC MAIN
N020 ...
N030 EXTCALL("SP_1")
N030 EXTCALL("USB:WKS.DIR/WST1.WPD/SP_2")
N050 ...
N060 M30
```

Informations complémentaires

Appel EXTCALL avec chemin d'accès absolu

Si le sous-programme existe sous le chemin spécifié, il est exécuté par l'appel EXTCALL. Si le sous-programme n'existe pas sous le chemin spécifié, l'exécution du programme par l'appel EXTCALL est annulée.

Appel EXTCALL avec chemin d'accès relatif / sans chemin d'accès

Lors d'un appel EXTCALL avec chemin d'accès relatif ou sans chemin d'accès, les mémoires de programme présentes sont parcourues selon le schéma suivant :

1. Lorsqu'un chemin d'accès est préréglé dans SD42700 $SC_EXT_PROG_PATH, la recherche s'effectue, dans un premier temps, à partir de cet endroit à l'aide de l'indication fournie dans l'appel EXTCALL (nom du programme avec chemin d'accès relatif, le cas échéant). Le chemin absolu est déterminé alors par concaténation de chaînes :
   - Chemin d'accès préréglé dans SD42700 $SC_EXT_PROG_PATH
   - Séparateur "/"
   - Chemin d'accès et nom de sous-programme dans l'instruction EXTCALL
2. Si le sous-programme indiqué au point 1. n'a pas été trouvé, tous les répertoires de la mémoire utilisateur sont parcourus.

La recherche est terminée dès que la première occurrence du sous-programme est trouvée. Si le sous-programme reste introuvable, l'exécution du programme par l'appel EXTCALL est annulée.

Mémoire de chargement réglable (tampon FIFO)

L'exécution d'un sous-programme externe requiert une mémoire de chargement. La taille de la mémoire de chargement est préréglée sur 30 Ko et ne peut être modifiée que par le constructeur de machines (via MD18360 MM_EXT_PROG_BUFFER_SIZE).

Remarque

Sous-programmes avec instructions de saut

Pour les sous-programmes externes qui contiennent des instructions de saut (GOTOF, GOTOB, CASE, FOR, LOOP, WHILE, REPEAT, IF, ELSE, ENDIF, etc.), les destinations doivent figurer dans la mémoire de chargement.

Remarque

Programmes ShopMill/ShopTurn

Les programmes ShopMill et ShopTurn doivent être stockés dans la mémoire de chargement dans leur intégralité du fait des descriptions de contour ajoutées à la fin du fichier.

Si plusieurs sous-programmes externes sont exécutés en parallèle, une mémoire de chargement propre à chacun d'eux est nécessaire.

Reset / Fin de programme / POWER ON

Avec Reset et POWER ON, les appels de sous-programme externes sont interrompus et les mémoires de chargement correspondantes supprimées.

Un programme sélectionné pour "exécution d'un programme externe" conserve cette sélection après un Reset, une fin de programme ou un POWER ON. Le comportement ne diffère pas par rapport aux programmes sélectionnés en interne tant que la mémoire de programmes externe reste disponible.

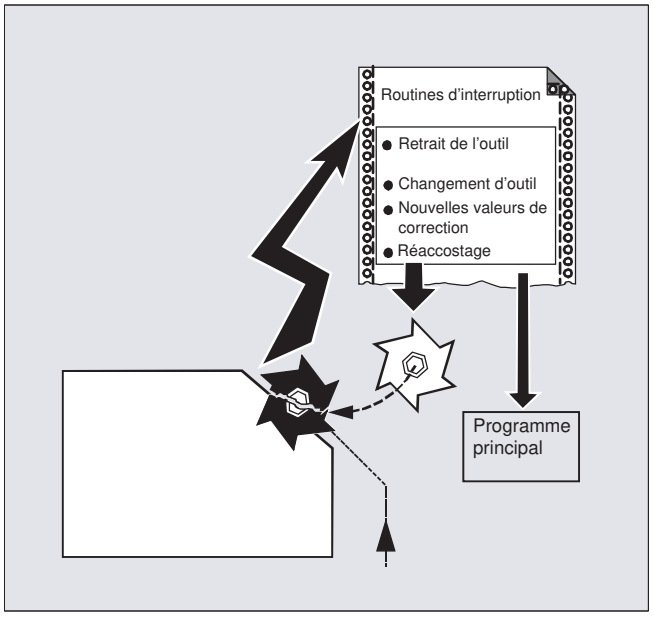
3.3 Routine d'interruption (ASUP)

3.3.1 Fonction d'une routine d'interruption

Remarque

Les termes "sous-programme asynchrone (ASUP)" et "routine d'interruption" utilisés en alternance dans la description suivante désignent la même fonctionnalité.

La fonction d'une routine d'interruption est explicitée à l'aide d'un exemple typique :



L'outil se brise pendant l'usinage. Ceci déclenche un signal qui arrête la phase d'usinage en cours et lance simultanément un sous-programme appelé "routine d'interruption". Ce sous-programme contient toutes les instructions qui doivent être exécutées dans ce cas.

Lorsque l'exécution du sous-programme est terminée (et la machine à nouveau prête), la commande retourne dans le programme principal et poursuit l'usinage – selon l'instruction REPOS – à l'endroit où il avait été interrompu (voir "Réaccostage du contour (Page 880)").

⚠ PRUDENCE

Risque de collision

Si aucune instruction REPOS n'est programmée dans le sous-programme, le positionnement a lieu au point final du bloc qui suit le bloc interrompu.

3.3.2 Création d'une routine d'interruption

Création d'une routine d'interruption en tant que sous-programme

La routine d'interruption est définie comme un sous-programme.

Exemple :

Code de programme	**Commentaire**
PROC RETR_Z	; Nom de programme "RELEV_Z"
N10 ...	; Puis suivent les blocs CN.
...	
N50 M17	; Pour terminer, fin du programme et retour au programme principal.

Sauvegarde des instructions M modales (SAVE)

Lors de la définition, la routine d'interruption peut être accompagnée de l'attribut SAVE.

Ce dernier entraîne la sauvegarde des instructions G modales actives avant l'appel de la routine d'interruption, et leur réactivation à la fin de la routine d'interruption (voir " Sous-programmes avec mécanisme de sauvegarde (SAVE) (Page 512) ").

Il est ainsi possible de reprendre l'usinage au point d'interruption après l'exécution de la routine d'interruption.

Exemple :

Code de programme
PROC RETR_Z SAVE
N10 ...
...
N50 M17

Affectation d'autres routines d'interruption (SETINT)

À l'intérieur de la routine d'interruption, il est possible de programmer des instructions SETINT (voir "Affectation et démarrage d'une routine d'interruption (SETINT)" (Page 552)) pour déclencher d'autres routines d'interruption. Le démarrage de ces dernières n'a lieu que lorsque l'entrée correspondante commute.

3.3.3 Affectation et démarrage d'une routine d'interruption (SETINT, PRIO, BLSYNC)

La commande dispose de plusieurs entrées rapides (entrée 1 à 8), qui déclenchent chacune une interruption (1 à 8). Une priorité et une routine d'interruption peuvent être affectées à chaque interruption par l'instruction SETINT. Si l'interruption est déclenchée par l'activation de l'entrée rapide, l'exécution actuelle dans le canal est interrompue et la routine d'interruption est lancée.

Priorité d'interruption

Si des interruptions sont affectées à plusieurs entrées dans un programme pièce, différentes priorités doivent être affectées aux interruptions.

Une valeur de priorité comprise entre 1 et 128 peut être affectée à une interruption. La valeur de priorité 1 correspond à la priorité la plus élevée, la valeur 128 à la priorité la plus faible.

Syntaxe

```
SETINT(<n>) <NOM>
SETINT(<n>) PRIO=<valeur> <NOM>
SETINT(<n>) PRIO=<valeur> <NOM> BLSYNC
SETINT(<n>) PRIO=<valeur> <NOM> LIFTFAST
```

Signification

SETINT(<n>) :	Le programme CN (ASUP) <nom> est affecté au signal d'interruption <n>. La routine d'interruption affectée est lancée dès que le signal d'interruption <n> == 1 est détecté. **Remarque :** Si une autre routine d'interruption est affectée à un signal d'interruption <n>, l'affectation précédente est désactivée.	
<n> :	Numéro du signal d'interruption	
	Type :	INT
	Plage de valeurs :	1 ... 32
PRIO= :	Priorité de l'interruption (facultatif)	
<valeur> :	(**facultatif**) Valeur de priorité	
	Type :	INT
	Plage de valeurs :	1 ... 128 (1 ⇒ priorité la plus élevée)
<NOM> :	Nom du programme CN (ASUP)	
BLSYNC :	(**facultatif**) BLSYNC entraîne, dans un premier temps après le déclenchement de l'interruption, l'attente de la fin de l'exécution du bloc actuel. La routine d'interruption n'est exécutée qu'après.	
LIFTFAST :	(**facultatif**) LIFTFAST entraîne, dans un premier temps après le déclenchement de l'interruption, un relèvement rapide (voir chapitre "Relèvement rapide du contour (SETINT LIFTFAST, ALF) (Page 556)"). La routine d'interruption n'est exécutée qu'après.	

Autres conditions à prendre en compte

Règles d'interruption

1. Pour chaque interruption ne pouvant pas être exécutée immédiatement, ou déjà en cours d'exécution, une demande d'interruption supplémentaire est enregistrée. Au-delà de cela, toute demande d'interruption supplémentaire pour cette interruption est perdue.
2. Si une interruption est en cours d'exécution quand une autre interruption de priorité supérieure est déclenchée, cette dernière interrompt l'interruption de priorité inférieure. Une fois l'interruption de priorité supérieure terminée, l'interruption de priorité inférieure se poursuit. Si, pendant l'exécution de l'interruption de priorité supérieure, d'autres demandes arrivent pour l'interruption de priorité inférieure, une demande est enregistrée. Toute demande supplémentaire est perdue.
3. Si une interruption est en cours d'exécution quand une autre interruption de priorité supérieure est déclenchée, cette dernière interrompt l'interruption de priorité inférieure. L'interruption de priorité supérieure est exécutée. Si une interruption de priorité supérieure est une nouvelle fois déclenchée, l'interruption actuelle est interrompue et l'interruption de priorité supérieure est exécutée. Jusqu'à six niveaux d'interruption actifs sont possibles : un niveau d'interruption en cours d'exécution et cinq niveaux d'interruption en attente. Pour chaque niveau d'interruption actif, une seule demande d'interruption supplémentaire est enregistrée. Toutes les autres demandes d'interruption sont perdues. De même, toute demande d'interruption est perdue lorsqu'elle concerne des niveaux d'interruption supplémentaires (niveau d'interruption ≥ 7).

Exemples

Exemple 1 : affectation des routines d'interruption et définition de priorité

Code de programme	Commentaire
N20 SETINT(3) PRIO=1 RELEV_Z	; IF entrée 3 == 1
	; THEN lancer la routine d'interruption "RELEV_Z"
N30 SETINT(2) PRIO=2 RELEV_X	; IF entrée 2 == 1
	; THEN lancer la routine d'interruption "RELEV_X"

Si les entrées commutent simultanément, les routines d'interruption sont exécutées successivement, dans l'ordre des valeurs de priorité : d'abord "RELEV_Z", puis "RELEV_X".

Exemple 2 : affectation d'une nouvelle routine d'interruption

Code de programme	Commentaire
N20 SETINT(3) PRIO=2 RELEV_Z	; IF entrée 3 == 1
	; THEN lancer la routine d'interruption **"RELEV_Z"**
...	
N80 SETINT(3) PRIO=1 RELEV_X	; IF entrée 3 == 1
	; THEN lancer la routine d'interruption **"RELEV_X"**

3.3.4 Désactivation/activation de l'affectation d'une routine d'interruption (DISABLE, ENABLE)

Une instruction SETINT se désactive avec DISABLE et se réactive avec ENABLE sans perdre l'affectation entrée → routine d'interruption.

Syntaxe

```
DISABLE(<n>)
ENABLE(<n>)
```

Signification

DISABLE(<n>) :	Instruction : **Désactivation** de l'affectation de la routine d'interruption à l'entrée <n>	
ENABLE(<n>) :	Instruction : **Réactivation** de l'affectation de la routine d'interruption à l'entrée <n>	
<n> :	Paramètre : Numéro du signal d'interruption	
	Type :	INT
	Plage de valeurs :	1 ... 32

Exemple

Code de programme	**Commentaire**
N20 SETINT(3) PRIO=1 RELEV_Z	; Sur commutation de l'entrée 3, la
	; routine d'interruption "RELEV_Z" doit démarrer.
...	
N90 DISABLE(3)	; Désactivation de l'instruction SETINT de N20.
...	
N130 ENABLE(3)	; Réactivation de l'instruction SETINT de N20.
...	

3.3.5 Suppression d'une affectation de routine d'interruption (CLRINT)

L'affectation d'un signal d'interruption à un programme CN (ASUP) définie avec SETINT peut être supprimée avec CLRINT.

Syntaxe

```
CLRINT(<n>)
```

Signification

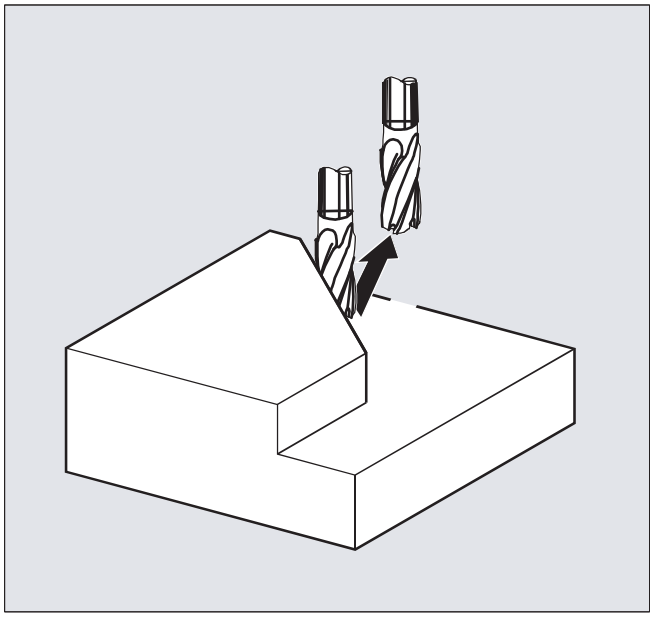
CLRINT(<n>) :	Instruction : Suppression de l'affectation du signal d'interruption <n> au programme CN (ASUP) <n> défini avec SETINT	
<n> :	Paramètre : Numéro du signal d'interruption	
	Type :	INT
	Plage de valeurs :	1 ... 32

Exemple

Code de programme	**Commentaire**
N20 SETINT(3) PRIO=2 RELEV_Z	
...	
N50 CLRINT(3)	; L'affectation de la routine d'interruption "RELEV_Z" à l'entrée "3" est supprimée.

3.3.6 Relèvement rapide du contour (SETINT LIFTFAST, ALF)

Dans le cas d'une instruction SETINT avec LIFTFAST, un relèvement rapide éloigne l'outil du contour de pièce lorsque l'entrée commute.



La suite dépend de l'instruction SETINT, c'est-à-dire si celle-ci contient ou non une routine d'interruption en plus de LIFTFAST :

Avec une routine d'interruption :	La routine d'interruption est exécutée **après** le relèvement rapide.
Sans routine d'interruption :	Après le relèvement rapide, l'usinage s'arrête avec une alarme.

Syntaxe

```
SETINT(<n>) PRIO=1 LIFTFAST
SETINT(<n>) PRIO=1 <NOM> LIFTFAST
```

Signification

SETINT(<n>) :	Instruction : affectation de l'entrée <n> à une routine d'interruption. La routine d'interruption affectée démarre lorsque l'entrée <n> commute.	
<n> :	Paramètre : Numéro de l'entrée	
	Type :	INT
	Plage de valeurs :	1 ... 8
PRIO= :	Définition de la priorité	
<valeur> :	Valeur de priorité	
	Plage de valeurs :	1 ... 128
	La priorité 1 correspond à la priorité maximale.	
<nom> :	Nom du sous-programme (routine d'interruption) qui doit être exécuté.	
LIFTFAST :	Instruction : Relèvement rapide du contour	
ALF=... :	Instruction : Sens de déplacement programmable (figure dans le bloc de déplacement) Pour les options de programmation avec ALF, voir chapitre "Sens de relèvement rapide du contour (Page 558)".	

Autres conditions à prendre en compte

Comportement en cas de frame actif avec fonction miroir

Lors de la détermination du sens de relèvement, il est vérifié si un frame avec fonction miroir est actif. Dans ce cas, la gauche et la droite sont inversées pour le sens de relèvement par rapport à la direction tangentielle. Les parts directionnelles dans le sens de l'outil ne seront pas prises en compte par la fonction miroir. Ce comportement est activé par le réglage du paramètre machine :

PM21202 $MC_LIFTFAST_WITH_MIRROR = TRUE

Exemple

Un outil brisé doit être remplacé automatiquement par un outil de rechange. L'usinage se poursuit alors avec le nouvel outil.

Programme principal :

Programme principal	**Commentaire**
N10 SETINT(1) PRIO=1 CHANG_O LIFTFAST	; Si l'entrée 1 commute, un relèvement rapide éloigne immédiatement l'outil du contour (code n° 7 pour la correction du rayon d'outil G41). Il est suivi de l'exécution de la routine d'interruption "CHANG_O".
N20 G0 Z100 G17 T1 ALF=7 D1	

Programme principal	**Commentaire**
N30 G0 X-5 Y-22 Z2 M3 S300	
N40 Z-7	
N50 G41 G1 X16 Y16 F200	
N60 Y35	
N70 X53 Y65	
N90 X71.5 Y16	
N100 X16	
N110 G40 G0 Z100 M30	

Sous-programme :

Sous-programme	**Commentaire**
PROC CHANG_O SAVE	; Sous-programme avec mise en mémoire de l'état courant
N10 G0 Z100 M5	;position de changement de l'outil, arrêt de la broche
N20 T11 M6 D1 G41	; Changement d'outil
N30 REPOSL RMBBL M3	; Réaccostage du contour et retour au programme principal (programmation dans un bloc)

3.3.7 Sens de retrait rapide du contour

Déplacement de retrait

Le plan du déplacement de retrait est déterminé par les instructions G suivants :

- LFTXT
  Le plan du déplacement de retrait est déterminé par la tangente de la trajectoire et la direction de l'outil (réglage par défaut).
- LFWP
  Le plan du déplacement de retrait est le plan de travail actif sélectionné avec les instructions G17, G18 ou G19. Le sens du déplacement de retrait est indépendant de la tangente à la trajectoire. Ceci permet de programmer un relèvement rapide paraxial.
- LFPOS
  Retrait de l'axe introduit avec POLFMASK / POLFMLIN à la position d'axe absolue programmée avec POLF.
  ALF n'a aucune influence sur la direction du relèvement pour plusieurs axes ainsi que pour plusieurs axes dans un contexte linéaire.

Sens de déplacement programmable (ALF=...)

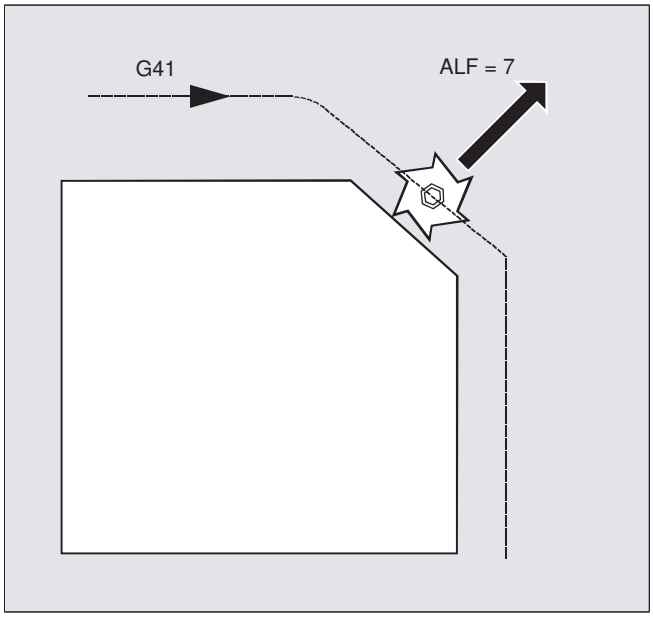
Dans le plan du mouvement de retrait, la direction est programmée avec ALF par pas discrets de 45 degrés.

Les sens de déplacement possibles sont enregistrés dans la commande sous des n° de code spéciaux et peuvent être appelés par le biais de ces numéros.

Exemple :

```
Code de programme
N10 SETINT(2) PRIO=1 RETR_Z LIFTFAST
ALF=7
```

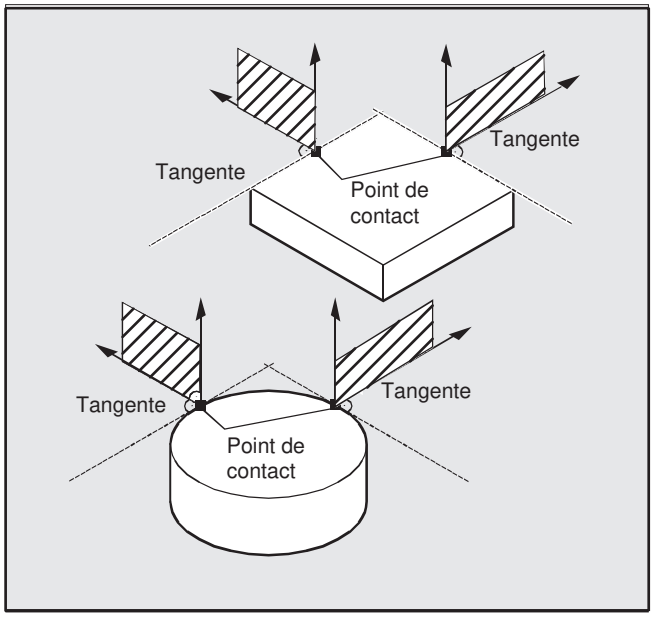
Lorsque G41 est activé (sens d'usinage à gauche du contour), le retrait de l'outil est perpendiculaire au contour.



Plan de référence pour la description du sens de déplacement avec LFTXT

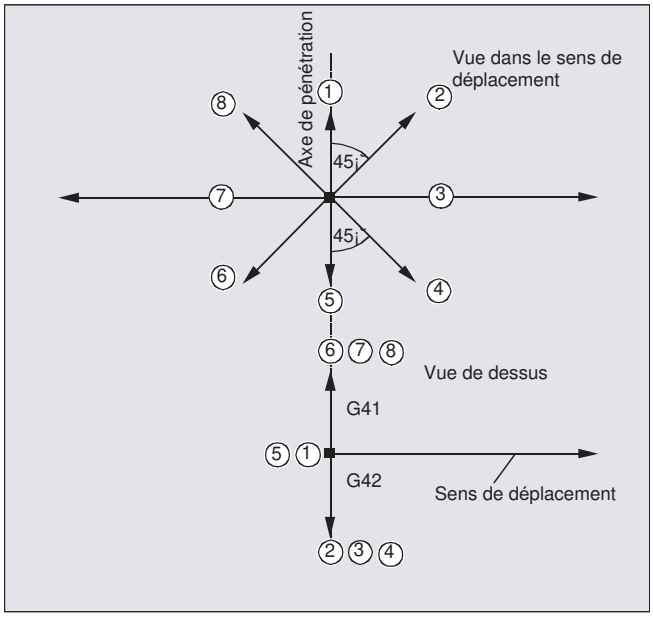
Un plan est défini au point de contact de l'outil avec le contour programmé ; ce plan sert de référence pour préciser le mouvement de relèvement avec le numéro de code correspondant.

Le plan de référence est défini à partir de l'axe longitudinal de l'outil (direction de pénétration) et d'un vecteur perpendiculaire à ce dernier et à la tangente au point de contact de l'outil avec le contour.



Numéros de code pour le sens de déplacement avec LFTXT

La figure ci-dessous contient les numéros de code avec les sens de déplacement en partant du plan de référence.



Avec `ALF=1`, le retrait a lieu dans la direction de l'outil.

Avec ALF=0, la fonction de relèvement rapide est désactivée.

⚠ PRUDENCE

Risque de collision

Lorsque la correction du rayon d'outil est activée, il est recommandé de ne pas utiliser :

- les codes 2, 3, 4 pour G41
- les codes 6, 7, 8 pour G42

car, dans ces cas, l'outil se déplace vers le contour et entre en collision avec la pièce.

Numéros de code pour le sens de déplacement avec LFWP

Avec LFWP, la direction dans le plan de travail est tributaire des configurations suivantes :

- G17 : Plan X/Y (X/Y plane)
  ALF=1: retrait en direction X
  ALF=3 : retrait en direction Y
- G18 : Plan Z / X
  ALF=1: retrait en direction Z
  ALF=3 : retrait en direction X
- G19 : Plan Y/Z (Y/Z plane)
  ALF=1: retrait en direction Y
  ALF=3 : retrait en direction Z

Voir aussi

Retrait rapide pendant un filetage (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN) (Page 235)

3.3.8 Séquence de déplacement avec des routines d'interruption

Routine d'interruption sans LIFTFAST

Les déplacements d'axe sont freinés sur la trajectoire jusqu'à l'arrêt. Puis la routine d'interruption démarre.

La position d'arrêt est enregistrée comme position d'interruption et est accostée à la fin de la routine d'interruption en présence de REPOS avec RMIBL.

Routine d'interruption avec LIFTFAST

Les déplacements d'axe sont freinés sur la trajectoire. Simultanément, le déplacement LIFTFAST est exécuté comme déplacement superposé. Lorsque le déplacement avec interpolation et le déplacement LIFTFAST se sont immobilisés, la routine d'interruption démarre.

La position où le déplacement `LIFTFAST` a démarré sur le contour et où l'outil a quitté la trajectoire est enregistrée comme position d'interruption.

Avec `LIFTFAST` et `ALF=0`, le comportement de la routine d'interruption est identique à celui de la routine sans `LIFTFAST`.

Remarque

La distance de relèvement rapide des axes géométriques par rapport au contour est réglable dans un paramètre machine.

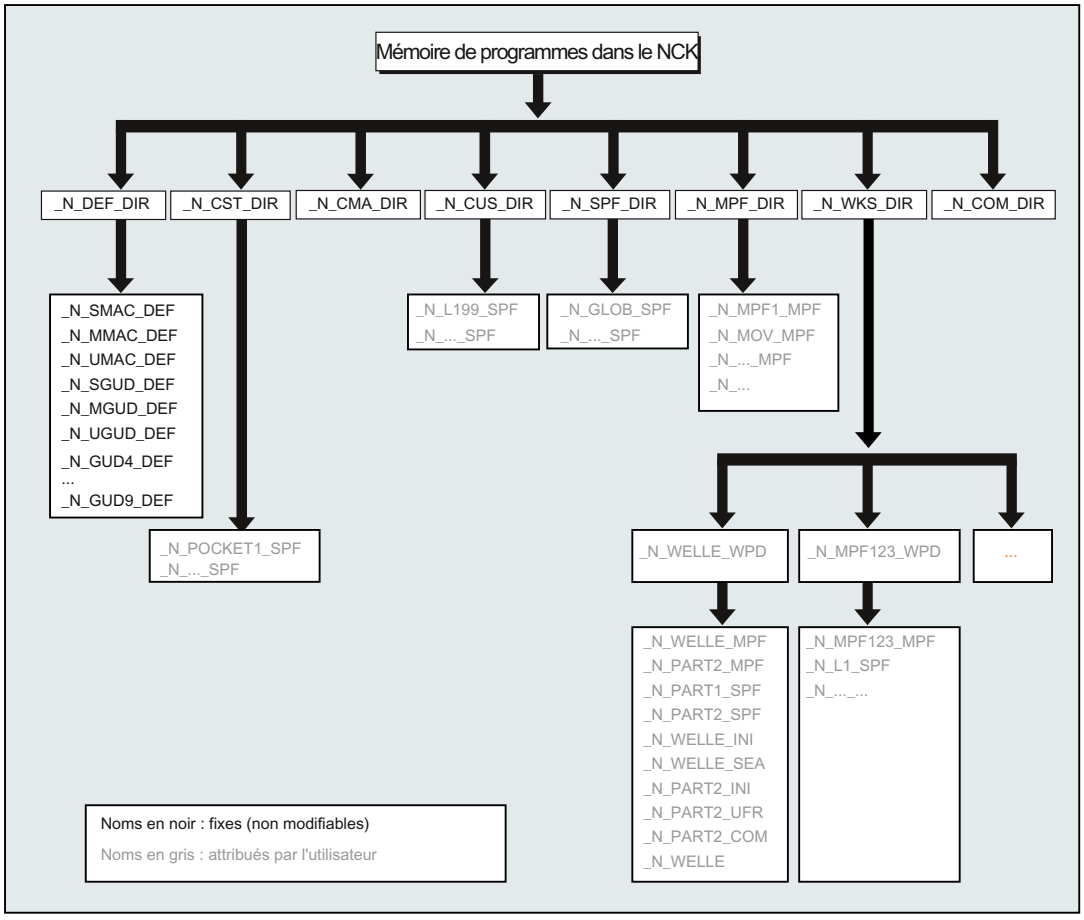
3.4 Gestion des fichiers et programmes

3.4.1 Mémoire de programmes

3.4.1.1 Mémoire de programmes dans le NCK

Les fichiers et programmes (p. ex. programmes principaux et sous-programmes, définitions de macros) sont stockés de manière permanente dans la mémoire de programme (→ système de fichiers passif).

Il existe en outre un certain nombre de types de fichiers qui peuvent y être stockés de façon provisoire pour être transférés sur demande (par exemple lors de la réalisation d'une certaine pièce) dans la mémoire de travail (par exemple pour des raisons d'initialisation).



Répertoires standard

Les répertoires suivants sont créés en version standard :

Répertoire	Contenu
_N_DEF_DIR	Blocs de données et blocs de macros
_N_CST_DIR	Cycles standard
_N_CMA_DIR	Cycles constructeur
_N_CUS_DIR	Cycles utilisateur
_N_WKS_DIR	Pièces
_N_SPF_DIR	Sous-programmes globaux
_N_MPF_DIR	Programmes principaux
_N_COM_DIR	Commentaires

Types de fichiers

Les types de fichiers suivants peuvent être introduits dans la mémoire de programmes :

Type de fichier	Description
<Nom>_MPF	Programme principal
<Nom>_SPF	Sous-programme
<Nom>_TEA	Paramètres machine
<Nom>_SEA	Données de réglage
<Nom>_TOA	Corrections d'outil
<Nom>_UFR	Décalages d'origine / frames
<Nom>_INI	Fichier d'initialisation
<Nom>_GUD	Données utilisateur globales
<Nom>_RPA	Paramètres R
<Nom>_COM	Commentaire
<Nom>_DEF	Définitions pour données utilisateur globales et macro-instructions

Catalogue des pièces (_N_WKS_DIR)

Le catalogue des pièces est configuré par défaut dans la mémoire de programmes sous la désignation _N_WKS_DIR. Il contient les répertoires pièce pour toutes les pièces qui ont été programmées.

Catalogues des pièces (..._WPD)

Un répertoire pièce contient tous les fichiers nécessaires pour l'usinage d'une pièce. Ceux-ci peuvent être des programmes principaux, des sous-programmes, tous programmes d'initialisation et fichiers de commentaires.

Les programmes d'initialisation sont exécutés une seule fois après la sélection du programme avec le premier lancement du programme pièce (selon le paramètre machine MD11280 $MN_WPD_INI_MODE).

Exemple :

Le répertoire pièce _N_ARBRE_WPD, qui a été créé pour la pièce ARBRE, contient les fichiers suivants :

Fichier	Description
_N_ARBRE_MPF	Programme principal
_N_PART2_MPF	Programme principal
_N_PART1_SPF	Sous-programme
_N_PART2_SPF	Sous-programme
_N_ARBRE_INI	Programme général d'initialisation des données pour la pièce
_N_ARBRE_SEA	Programme d'initialisation des données de réglage
_N_PART2_INI	Programme général d'initialisation des données pour le programme partie 2
_N_PART2_UFR	Programme d'initialisation des données frame pour le programme partie 2
_N_ARBRE_COM	Fichier de commentaires

En outre, des données qui ne sont pas directement nécessaires pour l'usinage par la CN peuvent également être stockées dans un répertoire pièce. Il peut s'agir de fichiers ASCII mais aussi de fichiers binaires comme, par exemple, des images au format JPG ou des descriptions au format PDF. Pour que la CN puisse les interpréter en tant que fichiers binaires, les extensions de fichier doivent être connues dans la CN (réglage lors de la mise en service via MD17000 $MN_EXTENSIONS_OF_BIN_FILES ; dans le réglage de base, les extensions de fichier suivantes sont prédéfinies : JPG, GIF, PNG, BMP, PDF, ICO, HTM).

Sélection d'une pièce pour exécution

Un répertoire pièce peut être sélectionné dans un canal en vue de son exécution. S'il existe dans ce répertoire un programme principal **de même nom** ou un seul programme principal (_MPF), il sera sélectionné automatiquement pour l'exécution.

3.4.1.2 Mémoires de programmes externes

Outre le système de fichiers passif dans la CN, des mémoires de programmes externes peuvent également être présentes sur une machine (par ex. sur le lecteur local ou sur un lecteur réseau).

Les fonctions "Exécution d'un programme externe" et "EES (Execution from External Storage)" permettent d'exécuter des programmes pièce **directement** depuis des mémoires de programmes externes.

Informations complémentaires : Description fonctionnelle Fonctions de base

Mémoire globale de programmes pièce (GDIR)

Lors de la déclaration des lecteurs, l'un des lecteurs peut être configuré en tant que mémoire globale de programmes pièce (GDIR).

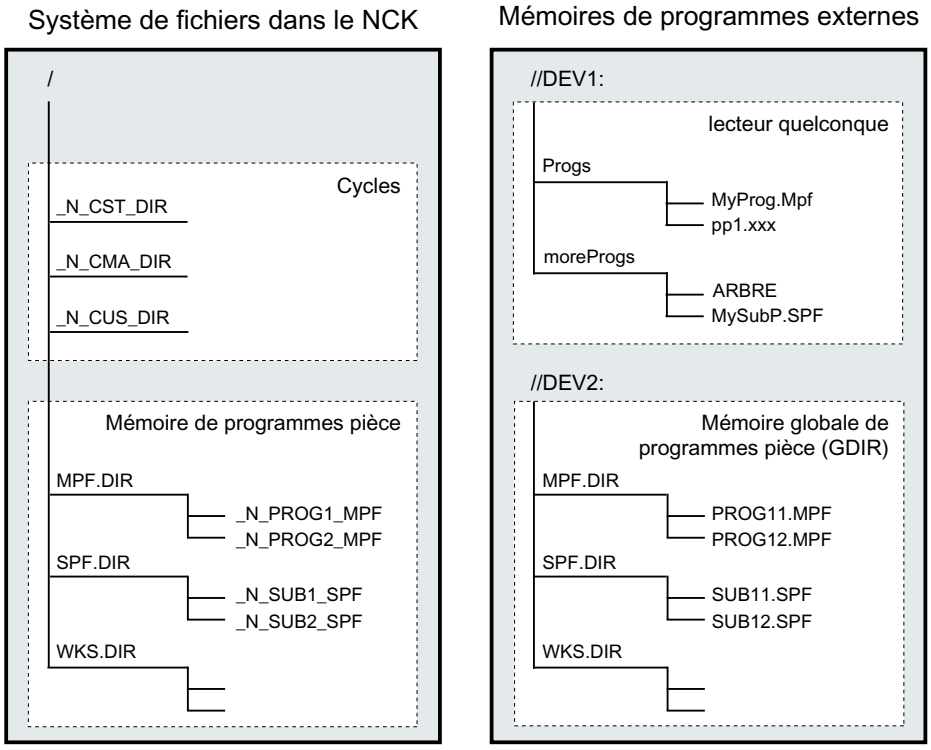
Le système crée alors automatiquement les répertoires MPF.DIR, SPF.DIR et WKS.DIR sur le lecteur. Ces trois répertoires constituent la GDIR.

La GDIR ne joue un rôle que pour la fonction EES. Selon la configuration du lecteur, la GDIR remplace ou étend la mémoire de programmes pièce CN. La configuration d'une GDIR n'est toutefois pas nécessaire pour le fonctionnement de l'EES.

Les répertoires et fichiers de la GDIR peuvent être adressés dans le programme pièce comme dans le système de fichiers passif. Il est ainsi possible d'effectuer un transfert compatible un programme CN avec chemins d'accès à partir du système de fichiers passif dans la GDIR. Le répertoire SPF.DIR de la GDIR se trouve dans le chemin de recherche des sous-programmes.

Organisation de programmes

La figure suivante met en évidence l'organisation des programmes dans les mémoires de programmes externes :



Systèmes de fichiers insensibles à la casse

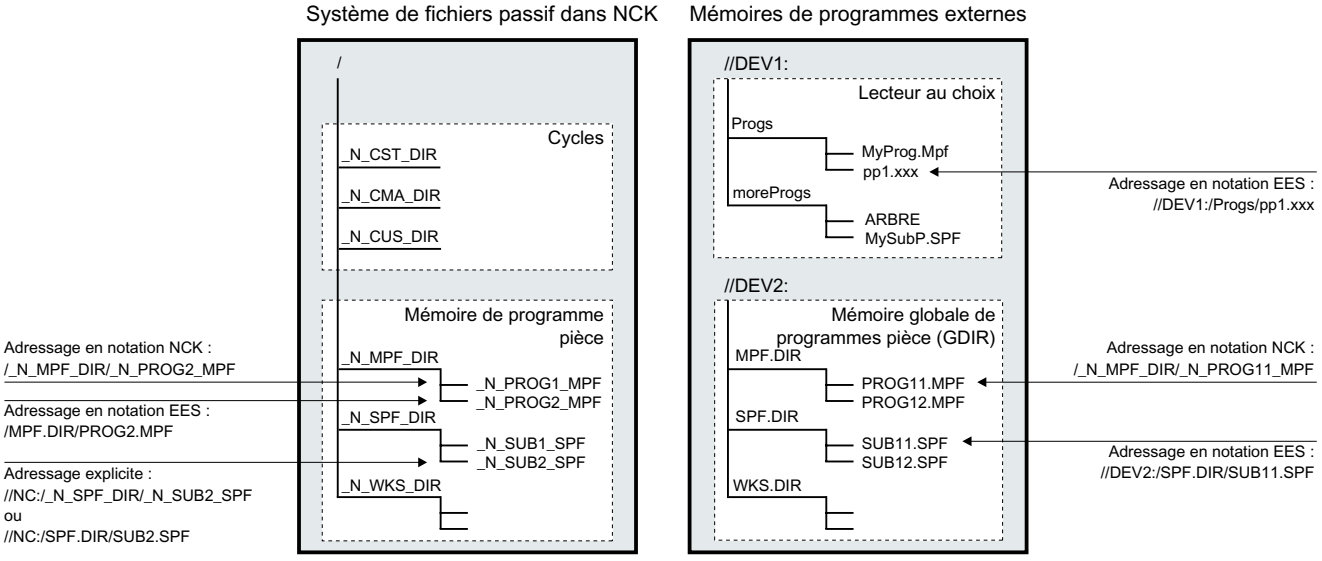
Remarque

Pour éviter les problèmes liés aux majuscules/minuscules lors de l'adressage des fichiers (voir "Adressage des fichiers de la mémoire de programmes (Page 567)"), des systèmes de fichiers **insensibles à la casse** doivent être utilisés comme mémoires de programmes externes.

3.4.1.3 Adressage des fichiers de la mémoire de programmes

Un fichier de la mémoire de programmes, adressé par une instruction de gestion de fichier (p. ex. WRITE, DELETE, READ, ISFILE, FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO), est référencé soit par un chemin absolu et le nom de fichier, soit par le nom de fichier uniquement. Dans le deuxième cas, le chemin du programme sélectionné est utilisé comme chemin d'accès au fichier.

Adressage en notation CN/EES



Adressage des fichiers du système de fichiers passif

L'adressage des fichiers du système de fichiers passif s'effectue généralement en **notation CN** (les noms des répertoires et des fichiers commencent par l'identificateur de domaine "_N_", le séparateur pour l'extension de répertoire/fichier est "_") sans spécification du nom de lecteur. Un adressage en **notation EES** (sans l'identificateur de domaine "_N_", le séparateur pour l'extension de répertoire/fichier est ".") est également possible.

Exemple :

- Notation CN : "/_N_SPF_DIR/_N_SUB1_SPF"
- Notation EES : "/SPF.DIR/SUB1.SPF"

Remarque

L'adressage des fichiers du système de fichiers passif en notation EES est converti en interne en notation CN selon les règles suivantes :

- les noms de répertoires et de fichiers sont complétés par l'identificateur de domaine "_N_".
- Si le quatrième caractère en partant de la fin dans les noms de répertoires ou de fichiers est un point ("."), il est converti en tiret bas ("_").

Le nom de lecteur prédéfini "//NC:" permet également un adressage ciblé du système de fichiers passif.

Exemple :

- Notation CN : "//NC:/_N_SPF_DIR/_N_SUB1_SPF"
- Notation EES : "//NC:/SPF.DIR/SUB1.SPF"

Adressage des fichiers d'une mémoire de programmes externe

L'adressage des fichiers d'une mémoire de programmes externe non configurée en tant que GDIR doit être effectué en notation EES. Le nom de lecteur (p. ex. "//DEV1:") doit être spécifié au début du chemin d'adressage. Tous les noms symboliques de périphériques configurés dans /user/sinumerik/hmi/cfg/logdrive.ini sont autorisés.

Exemple :

- Notation EES : "//DEV1:/MyProgDir/pp1.xxx"
- Notation CN : non admissible

Adressage des fichiers de la mémoire globale de programmes pièce (GDIR)

Lors de l'adressage des fichiers de la GDIR, un chemin d'accès en notation CN est également autorisé en plus du chemin d'accès en notation EES.

Exemple :

- Notation EES : "//DEV2:/MPF.DIR/PROG11.MPF"
- Notation CN : "/_N_MPF_DIR/_N_PROG11_MPF"

Remarque

L'adressage des fichiers de la GDIR en notation CN est converti en interne en notation EES selon les règles suivantes :

- L'identificateur de domaine "_N_" est supprimé dans les noms de répertoires et de domaines.
- Si le quatrième caractère en partant de la fin dans les noms de répertoires ou de fichiers est un tiret bas ("_"), il est converti en point (".").

Règles concernant le chemin d'accès

L'indication complète d'un chemin comprend le nom du lecteur, le chemin du répertoire et le nom du fichier.

Nom du lecteur

Les règles suivantes s'appliquent pour l'indication du nom du lecteur :

- Tous les noms symboliques de périphériques configurés dans /user/sinumerik/hmi/cfg/logdrive.ini sont autorisés.
- Il commence par les caractères "//", suivi d'au moins une lettre ou un chiffre.
- Quant aux caractères qui suivent, il peut s'agir d'une combinaison quelconque de lettres, de chiffres, de traits de soulignement ("_") et d'espaces.
- Le nom se termine par une lettre ou un chiffre, suivi d'un ":".
- Aucun autre caractère spécial n'est autorisé.

Remarque

Pour le système de fichiers passif, le nom du lecteur prédéfini est "//NC:".

Exemples :

- Mémoire de programmes externe :
  - //Drive1:
  - //Drive_1:
  - //Drive 1:
  - //A B:
  - //1 B C 2:

Chemin de répertoire

Les règles suivantes s'appliquent pour l'indication du chemin de répertoire :

- Le caractère "/" se trouve au début et à la fin du chemin de répertoire, et il sert de séparateur pour les différentes parties du chemin.

  Remarque

  Un double trait oblique ("//") au sein du chemin de répertoire n'est **pas** autorisé !

- Noms de répertoires :
  - les noms de répertoires doivent commencer par une lettre ou par un chiffre. Les noms de répertoires ne commencent par l'identificateur de domaine "_N_" que dans le cas de l'adressage en notation CN.
  - Quant aux caractères qui suivent, il peut s'agir d'une combinaison quelconque de lettres, de chiffres et de "_".

    Remarque

    Les espaces sont également autorisés dans les noms de répertoires pour les mémoires de programmes externes. Cela n'est toutefois pas le cas si la mémoire de programmes externe est configurée en tant que mémoire globale de programmes pièce (GDIR).

  - Aucun autre caractère spécial n'est autorisé.
- Extensions de répertoires :
  - les extensions de répertoires doivent comprendre exactement trois lettres/chiffres.
  - Elles sont séparées du nom du répertoire par "_" (notation CN) ou "." (notation EES).

  Remarque

  Dans le système de fichiers passif, les seules extensions de répertoires sont _DIR et _WPD.

Exemples :

- Système de fichiers passif ou GDIR :
  - Notation CN : _N_WKS_DIR/_N_MYNCPROGS_WPD/...
  - Notation EES : WKS.DIR/MYPROGS.WPD/...
- Mémoire de programmes externe :
  - /abc
  - /ab_c.def
  - /ab c1.def
  - /a b c .d11
  - /abc.def/ghi.klm

Nom de fichier

Les règles suivantes s'appliquent pour le nom de fichier :

- Les noms de fichiers ne commencent par l'identificateur de domaine "_N_" que dans le cas de l'adressage en notation CN.
- Les deux caractères suivants doivent être deux lettres, ou bien un tiret bas suivi d'une lettre.

  Remarque

  Si cette condition est remplie, un programme CN peut être appelé, par simple indication du nom du programme, comme sous-programme depuis un autre programme. En revanche, si le nom du programme commence par des chiffres, l'appel en tant que sous-programme est possible uniquement avec l'instruction CALL.

- Quant aux caractères qui suivent, il peut s'agir d'une combinaison quelconque de lettres, de chiffres et de "_".
- Extension du nom de fichier :
  - l'extension du nom de fichier doit comprendre exactement trois lettres/chiffres.

    Remarque

    Pour les extensions du nom de fichier autorisées dans le système de fichiers passif, voir "Mémoire de programmes dans le NCK (Page 563)".

  - L'extension est séparée du nom du fichier par "_" (notation CN) ou "." (notation EES).

Exemples :

- Système de fichiers passif ou GDIR :
  - Notation CN : _N_SUB1_SPF
  - Notation EES : SUB1.SPF
- Mémoire de programmes externe :
  - Partie 1
  - _pièce1
  - Pièce_1.spf
  - Pièce1.mpf

Nom de sous-programme DIN

Les règles suivantes s'appliquent pour le nom de sous-programme DIN :

- Le premier caractère est la lettre "L".
- Les caractères suivants sont des chiffres (au moins un).
- Extension du nom de fichier :
  - L'extension du nom de fichier doit comprendre exactement trois lettres.
  - L'extension est séparée du nom du fichier par "_" (notation CN) ou "." (notation EES).

Exemples :

- L123
- L1_SPF (notation CN) ou L1.SPF (notation EES)

Longueur de chemin maximale

Pour indiquer le nom du lecteur et le chemin du répertoire, vous disposez au maximum de 128 octets, et la longueur du nom du fichier ne doit pas dépasser 31 octets. Pour l'ensemble du chemin, cela donne une longueur maximale de 159 octets.

3.4.1.4 Chemin de recherche pour l'appel de sous-programmes

Lors d'appels de sous-programmes sans chemin d'accès, le chemin absolu est déterminé par l'exécution d'un chemin de recherche fixe.

La recherche dans les mémoires de programmes s'effectue dans l'ordre suivant :

		Répertoire	Description
1		répertoire actuel / *nom*	Le répertoire actuel est le répertoire à partir duquel le programme principal est appelé. A savoir : • un répertoire pièce ou le répertoire standard _N_MPF_DIR dans la mémoire de programmes pièce du CN ou la mémoire globale de programmes pièce ou • un répertoire quelconque d'une mémoire de programmes externe
2		répertoire actuel / *nom_SPF*	
3		répertoire actuel / *nom_MPF*	
4	a	//NC:/_N_SPF_DIR / *nom_SPF*	Répertoire de sous-programmes dans la mémoire de programmes pièce de la CN
	b	//DEV2:/_N_SPF_DIR / name_SPF [1]	Répertoire de sous-programmes dans la mémoire globale de programmes pièce **Remarque :** Cette étape de recherche est omise lorsque aucune mémoire globale de programmes pièce n'est configurée ou que l'appel de programme a eu lieu dans la mémoire de programme pièce de la CN.
5		Extension du chemin de recherche programmée avec `CALLPATH` (voir "Extension du chemin de recherche pour l'appel de sous-programmes (CALLPATH) (Page 546)"). **Remarque :** Cette étape de recherche est omise lorsque aucun `CALLPATH` n'est programmé.	
6		/_N_CUS_DIR / *nom_SPF*	Répertoire de cycles utilisateur
7		/_N_CMA_DIR / *nom_SPF*	Répertoire de cycles constructeur
8		/_N_CST_DIR / *nom_SPF*	Répertoire de cycles standard

[1] //DEV2:" représente par exemple le lecteur sur lequel la mémoire de programme pièce globale se trouve.

Les règles suivantes s'appliquent à la recherche :

- Le chemin de recherche est parcouru pour chaque appel de sous-programme individuel, c'est-à-dire que l'emplacement du programme de niveau supérieur est sans importance.
- Les types de fichiers pris en compte varient en fonction du répertoire.
- En règle générale, aucune recherche n'est effectuée dans les répertoires de niveau inférieur, c.-à-d. imbriqués, d'un répertoire.

3.4.1.5 Interrogation du chemin et du nom de fichier

Les variables système suivantes, pouvant être lues dans le programme pièce, sont disponibles pour l'interrogation du chemin et du nom de fichier d'un programme CN :

Variable système	Type	Signification
$P_STACK	INT	Fournit le niveau de programme dans lequel le programme CN actuel est exécuté.
$P_PATH[<n>]	STRING	Fournit le chemin du programme CN qui est exécuté dans le niveau de programme sélectionné par l'indice de tableau <n>. Exemples : $P_PATH[0] fournit le chemin pour le programme principal, p. ex. "/_N_WKS_DIR/_N_ARBRE_WPD/". $P_PATH[$P_STACK - 1] fournit le chemin du programme appelant. Si le chemin se rapporte à un programme CN stocké dans le système de fichiers passif de la CN ou dans la mémoire globale de programmes pièce (GDIR), le chemin est fourni en notation CN. Si le chemin se rapporte à un programme CN exécuté depuis une mémoire de programmes externe autre que la mémoire globale de programmes pièce, $P_PATH fournit le chemin en notation EES.
$P_PROG[<n>]	STRING	Fournit le nom du programme CN qui est exécuté dans le niveau de programme sélectionné par l'indice de tableau <n>. Si le programme CN est stocké dans le système de fichiers passif de la CN ou dans la mémoire globale de programmes pièce, le nom du programme est fourni en notation CN. Si le programme CN est exécuté depuis un lecteur externe autre que la mémoire globale de programmes pièce, $P_PROG fournit le nom en notation EES.
$P_PROGPATH	STRING	Fournit le chemin du programme CN en cours d'exécution. L'appel de $P_PROGPATH est identique à $P_PATH[$P_STACK].

Variable système	Type	Signification	
$P_IS_EES_PATH[<n>]	BOOL	Interrogation pour savoir si le chemin fourni par $P_PATH[<n>] ou le nom de programme fourni par $P_PROG[<n>] correspond à la notation CN ou à la notation EES.	
		= FALSE	$P_PATH[<n>] et $P_PROG[<n>] fournissent la notation CN, c.-à-d. qu'un préfixe "_N_" précède chaque descripteur. Le caractère de séparation pour l'extension de fichier est "_". Exemples : • Chemin en notation CN : "/_N_WKS_DIR/_N_MYWPD_WPD/" • Nom du programme en notation CN : "_N_MYPROG_MPF" Un chemin en notation CN peut se rapporter aussi bien au système de fichiers passif dans la CN qu'à la mémoire globale de programmes pièce.
		= TRUE	$P_PATH[<n>] et $P_PROG[<n>] fournissent la notation EES, c.-à-d. qu'aucun préfixe "_N_" ne précède le descripteur. Le caractère de séparation pour l'extension de fichier est ".". Exemples : • Chemin en notation EES : "//DEV1:/WKS.DIR/MYWPD.WPD/" • Nom du programme en notation EES : "MYPROG.MPF"

<n> : l'indice <n> définit le niveau de programme depuis lequel les informations de chemin doivent être lues (plage de valeurs : 0 ... 17)

Remarque

En mode de fonctionnement EES, en dehors de la mémoire globale de programmes pièce (GDIR), les chemins d'accès sont fournis en notation EES par les variables système $P_PROG, $P_PATH et $P_PROGPATH Les programmes utilisateur qui exploitent et traitent ces chemins d'accès doivent par conséquent, pour le fonctionnement EES, être étendus de telle sorte qu'ils puissent traiter également les chemins d'accès en notation EES.

3.4.2 Mémoire de travail (CHANDATA, COMPLETE, INITIAL)

Fonction

La mémoire de travail contient les données système et les données utilisateur courantes qui permettent à la commande de fonctionner (système actif de fichiers). Par exemple :

- Paramètres machine actifs
- Données de correction d'outil

- Décalages d'origine
- ...

programmes d'initialisation

Les programmes d'initialisation servent à l'initialisation des données de la mémoire de travail. Les types de fichier suivants peuvent être utilisés :

Type de fichier	Description
Nom_TEA	Paramètres machine
Nom_SEA	Données de réglage
Nom_TOA	Corrections d'outils
Nom_UFR	Décalages d'origine / frames
Nom_INI	Fichier d'initialisation
Nom_GUD	Données utilisateur globales
Nom_RPA	Paramètres R

Zones de données

Vous pouvez intégrer les données dans différentes zones au sein desquelles elles doivent être valides. Ainsi, une commande peut disposer de plusieurs canaux ou, d'une manière générale, également de plusieurs axes.

La répartition est la suivante :

identificateur	Zones de données
NC	Données spécifiques à la CN
CH<n>	Données spécifiques à un canal (<n> indique le numéro de canal)
AX<n>	Données spécifiques à un axe (<n> indique le numéro de l'axe machine)
TO	Données d'outil
COMPLETE	Toutes les données

Création d'un programme d'initialisation sur PC externe

A l'aide des identificateurs des zones de données et des types de données, vous pouvez déterminer les zones à considérer comme un ensemble lors de la sauvegarde des données :

_N_AX5_TEA_INI	Paramètres machine pour l'axe 5
_N_CH2_UFR_INI	Frames du canal 2
_N_COMPLETE_TEA_INI	Tous les paramètres machine

Après la mise en service de la commande, un bloc de données garantissant un fonctionnement correct de cette dernière est présent dans la mémoire de travail.

Procédure dans le cas des commandes multicanaux (CHANDATA)

CHANDATA(<numéro de canal>) pour plusieurs canaux est uniquement autorisé dans le fichier _N_INITIAL_INI. C'est le fichier de mise en service avec lequel toutes les données de la commande sont initialisées.

Code de programme	Commentaire
%_N_INITIAL_INI	
CHANDATA(1)	
	; Affectation des axes machine au canal 1 :
$MC_AXCONF_MACHAX_USED(0) = 1	
$MC_AXCONF_MACHAX_USED[1]=2	
$MC_AXCONF_MACHAX_USED[2]=3	
CHANDATA(2)	
	; Affectation des axes machine au canal 2 :
$MC_AXCONF_MACHAX_USED[0]=4	
$MC_AXCONF_MACHAX_USED[1]=5	
CHANDATA(1)	
	; Paramètres machine axiaux :
	; Fenêtre d'arrêt précis grossier :
$MA_STOP_LIMIT_COARSE[AX1]=0.2	; Axe 1
$MA_STOP_LIMIT_COARSE[AX2]=0.2	; Axe 2
	; Fenêtre d'arrêt précis fin :
$MA_STOP_LIMIT_FINE[AX1]=0.01	; Axe 1
$MA_STOP_LIMIT_FINE[AX1]=0.01	; Axe 2

IMPORTANT

Instruction CHANDATA

Dans le programme pièce, l'instruction CHANDATA ne peut être utilisée que pour le canal dans lequel le programme CN est exécuté ; c'est-à-dire que l'instruction peut être utilisée pour protéger le programme CN contre l'exécution dans un canal non prévu à cet effet.

En cas d'erreur, l'exécution du programme est abandonnée.

Remarque

Les fichiers INI dans des listes de tâches ne contiennent pas d'instruction CHANDATA.

Sauvegarde des programmes d'initialisation (COMPLETE, INITIAL)

Les fichiers de la mémoire de travail peuvent être sauvegardés sur un PC externe, puis être rechargés depuis celui-ci.

- La sauvegarde des fichiers s'effectue avec COMPLETE.
- INITIAL génère un fichier INI (_N_INITIAL_INI) pour toutes les zones.

Chargement de programmes d'initialisation

IMPORTANT
Perte de données Après le chargement du fichier "INITIAL_INI", tous les paramètres qui ne sont pas définis dans le fichier sont initialisés avec des paramètres par défaut, excepté les paramètres machine. Les **données de réglage, données d'outil, décalages d'origine, valeurs GUD, ...** sont donc réglés à des valeurs par défaut (normalement "ZERO").

Le fichier COMPLETE_TEA_INI convient par exemple au chargement de paramètres machine individuels. Dans ce fichier, la commande n'attend que des paramètres machine. Les autres zones de données restent donc inchangées dans ce cas.

Chargement de programmes d'initialisation

Les programmes INI peuvent également être sélectionnés et appelés comme programme pièce s'ils utilisent uniquement des données d'un canal. Il est ainsi également possible d'initialiser des données commandées par programme.

3.5 Gestion de fichiers

3.5.1 Ecriture d'un fichier (WRITE)

L'instruction `WRITE` permet d'écrire des blocs/données du programme CN à la fin d'un fichier se trouvant dans le système de fichiers passif ou sur une mémoire de programmes externe (fichier journal). Cette écriture peut également concerner le programme en cours d'exécution.

Remarque

Si le fichier à décrire avec l'instruction `WRITE` n'existe pas dans la mémoire de programmes, il est créé.

Condition

Le niveau de protection réglé doit être supérieur ou égal au droit WRITE du fichier. Si ce n'est pas le cas, l'accès est refusé avec émission du message d'erreur (valeur de retour de la variable d'erreur = 13).

Syntaxe

```
DEF INT <erreur>
...
WRITE(<erreur>,"<nom de fichier>"/"PériphExt","<bloc/données>")
```

Signification

`WRITE` :	Instruction d'insertion d'un bloc ou de données à la fin du fichier spécifié		
`<erreur>` :	**Paramètre 1 :** variable pour signalisation en retour de la valeur d'erreur		
	Type :	INT	
	Valeur :	0	Aucune erreur
		1	Chemin d'accès non autorisé
		2	Chemin d'accès introuvable
		3	Fichier introuvable
		4	Type de fichier incorrect
		10	Fichier plein
		11	Fichier en cours d'utilisation
		12	Pas de ressources disponibles
		13	Pas de droits d'accès
		14	Absence ou échec d'EXTOPEN pour le périphérique de sortie
		15	Erreur lors de l'écriture sur le périphérique de sortie
		16	Chemin externe programmé non valide

<nom de fichier> :	**Paramètre 2 :** nom du fichier dans lequel le bloc ou les données indiqués doivent être ajoutés	
	Type :	STRING
	Le chemin absolu peut être indiqué avant le nom de fichier proprement dit. En l'absence de chemin d'accès, le fichier est recherché dans le répertoire courant (= répertoire du programme sélectionné). Pour les règles concernant le chemin d'accès, voir "Adressage des fichiers de la mémoire de programmes (Page 567)".	
<PériphExt> :	Si des données doivent être transmises à un périphérique/fichier externe via la fonction "Process DataShare", le descripteur symbolique du périphérique/fichier externe à ouvrir doit être spécifié à la place du nom de fichier.	
	Type :	STRING
	Pour plus d'informations, voir "Process DataShare - Sortie sur un appareil/fichier externe (EXTOPEN, WRITE, EXTCLOSE) (Page 1033)". **Remarque :** Le descripteur doit être identique à celui spécifié dans l'instruction EXTOPEN.	
<bloc/ données> :	**Paramètre 3 :** bloc ou données à ajouter dans le fichier spécifié.	
	Type :	STRING

Remarque

Lors de l'écriture dans le système de fichiers passif ou sur une mémoire de programmes externe, l'instruction WRITE insère implicitement un caractère "LF" (LINE FEED = nouvelle ligne) à la fin de la chaîne de sortie.

Ce comportement ne s'applique pas à la sortie sur un périphérique/fichier externe via la fonction "Process DataShare". Si un "LF" doit être transmis en même temps, ceci doit être indiqué explicitement dans la chaîne de sortie.

→ Sur ce point, voir exemple 3 : "LF" implicite/explicite.

Autres conditions à prendre en compte

- **Taille maximale du fichier (→ constructeur de machine !)**
  La taille maximale possible des fichiers journaux dans le système de fichiers passif est réglée avec le paramètre machine :
  PM11420 $MN_LEN_PROTOCOL_FILE
  La taille maximale s'applique à tous les fichiers créés dans le système de fichiers passif avec l'instruction WRITE. En cas de dépassement, un message d'erreur est généré et l'enregistrement du bloc ou des données ne s'effectue pas. Si la capacité mémoire est suffisante, un nouveau fichier peut être créé.

Exemples

Exemple 1 : instruction WRITE dans le système de fichiers passif sans chemin d'accès absolu

Code de programme	Commentaire
N10 DEF INT ERROR	; Définition des variables d'erreur.

Code de programme	Commentaire
N20 WRITE(ERROR,"PROT","JOURNAL DU 07/02/97")	; Ecrire le texte "JOURNAL DU 07/02/97" dans le fichier _N_PROT_MPF.
N30 IF ERROR	; Analyse des erreurs.
N40 MSG ("Erreur lors instruction WRITE:" << ERROR)	
N50 M0	
N60 ENDIF	
...	

Exemple 2 : Instruction WRITE dans le système de fichiers passif avec chemin d'accès absolu

```
Code de programme
...
WRITE(ERROR,"/_N_WKS_DIR/_N_PROT_WPD/_N_PROT_MPF","JOURNAL DU
07/02/97")
...
```

Exemple 3 : "LF" implicite/explicite

a, écriture dans le système de fichiers passif avec le "LF" généré de façon implicite

```
Code de programme
...
N110 DEF INT ERROR
N120 WRITE(ERROR,"/_N_MPF_DIR/_N_MYPROTFILE_MPF","MY_STRING")
N130 WRITE(ERROR,"/_N_MPF_DIR/_N_MYPROTFILE_MPF","MY_STRING")
N140 M30
```

Résultat en sortie :

MY_STRING

MY_STRING

b, écriture dans un fichier externe sans "LF" généré de façon implicite

```
Code de programme
...
N200 DEF STRING[30] DEV_1
N210 DEF INT ERROR
N220 DEV_1="LOCAL_DRIVE/myprotfile.mpf"
N230 EXTOPEN(ERROR,DEV_1)
N240 WRITE(ERROR,DEV_1,"MY_STRING")
N250 WRITE(ERROR,DEV_1,"MY_STRING")
N260 EXTCLOSE(ERROR,DEV_1)
```

Code de programme

```
N270 M30
```

Résultat en sortie :

MY_STRINGMY_STRING

c, écriture dans un fichier externe avec un "LF" programmé de façon explicite

Pour obtenir le même résultat que pour a, la programmation suivante doit être effectuée :

Code de programme

```
...
N200 DEF STRING[30] DEV_1
N210 DEF INT ERROR
N220 DEV_1="LOCAL_DRIVE/myprotfile.mpf"
N230 EXTOPEN(ERROR,DEV_1)
N240 WRITE(ERROR,DEV_1,"MY_STRING'H0A'")
N250 WRITE(ERROR,DEV_1,"MY_STRING'H0A'")
N260 EXTCLOSE(ERROR,DEV_1)
N270 M30
```

Résultat en sortie :

MY_STRING

MY_STRING

3.5.2 Suppression d'un fichier (DELETE)

L'instruction `DELETE` permet de supprimer tous les fichiers, qu'ils aient été créés avec l'instruction `WRITE` ou non. `DELETE` permet également de supprimer les fichiers ayant été créés avec des niveaux d'accès supérieurs.

Syntaxe

```
DEF INT <erreur>
DELETE(<erreur>,"<nom de fichier>")
```

Signification

DELETE :	Instruction de suppression du fichier spécifié		
<erreur> :	Variable pour la valeur d'erreur en retour		
	type.	INT	
	Valeur :	0	Aucune erreur
		1	Chemin d'accès non autorisé
		2	Chemin d'accès introuvable
		3	Fichier introuvable
		4	Type de fichier incorrect
		11	Fichier en cours d'utilisation
		12	Pas de ressources disponibles
		20	Autre erreur
<nom de fichier> :	Nom du fichier à supprimer		
	Type :	STRING	
	Le chemin absolu peut être indiqué avant le nom de fichier proprement dit. En l'absence de chemin d'accès, le fichier est recherché dans le répertoire courant (= répertoire du programme sélectionné). Pour les règles concernant le chemin d'accès, voir "Adressage des fichiers de la mémoire de programmes (Page 567)".		

Exemple

Code de programme	**Commentaire**
N10 DEF INT ERROR	; Définition des variables d'erreur.
N15 STOPRE	; Arrêt du prétraitement des blocs.
N20 DELETE(ERROR,"/_N_SPF_DIR/_N_TEST1_SPF")	; Supprimer le fichier TEST1 du répertoire du sous-programme.
N30 IF ERROR	; Analyse des erreurs.
N40 MSG ("Erreur lors instruction DELETE :" <<ERROR)	
N50 M0	
N60 ENDIF	

3.5.3 Lecture des lignes d'un fichier (READ)

L'instruction READ permet de lire une ou plusieurs lignes dans le fichier spécifié et d'enregistrer les informations lues dans un tableau de type STRING. Chaque ligne lue occupe un élément de ce tableau.

Condition

Le niveau de protection réglé doit être supérieur ou égal au droit READ du fichier. Si ce n'est pas le cas, l'accès est refusé avec émission du message d'erreur (valeur de retour de la variable d'erreur = 13).

Syntaxe

```
DEF INT <erreur>
DEF STRING[<longueur de chaîne>] <résultat>[<n>,<m>]
READ(<erreur>,"<nom de fichier>",<ligne de début>,<nombre de
lignes>,<résultat>)
```

Signification

`READ` :	Instruction de lecture des lignes du fichier spécifié et d'enregistrement de ces lignes dans un tableau de variables.		
`<erreur>` :	Variable pour la valeur d'erreur en retour (paramètre Call-By-Reference)		
	type.	INT	
	Valeur :	0	Aucune erreur
		1	Chemin d'accès non autorisé
		2	Chemin d'accès introuvable
		3	Fichier introuvable
		4	Type de fichier incorrect
		11	Fichier en cours d'utilisation
		13	Droits d'accès insuffisants
		21	Ligne inexistante (paramètres `<ligne de début>` ou `<nombre de lignes>` supérieurs au nombre de lignes du fichier spécifié).
		22	Longueur de tableau de la variable de résultat (`<résultat>`) trop petite.
		23	Zone de lignes trop grande (paramètre `<nombre de lignes>` trop grand, si bien que la lecture se poursuit au-delà de la fin du fichier).
`<nom de fichier>` :	Nom du fichier à lire (paramètre Call-By-Value)		
	Type :	STRING	
	Le chemin absolu peut être indiqué avant le nom de fichier proprement dit. En l'absence de chemin d'accès, le fichier est recherché dans le répertoire courant (= répertoire du programme sélectionné). Pour les règles concernant le chemin d'accès, voir "Adressage des fichiers de la mémoire de programmes (Page 567)".		
`<ligne de début>` :	Ligne de début de la zone de fichier à lire (paramètre Call-By-Value)		
	Type :	INT	
	Valeur :	0	Le nombre de lignes, avant la fin du fichier, indiqué par le paramètre `<nombre de lignes>` est lu.
		1 ... n	Numéro de la première ligne à lire.
`<nombre de lignes>` :	Nombre de lignes à lire (paramètre Call-By-Value)		
	Type :	INT	

<résultat> :	Variable de résultat (paramètre Call-By-Reference) Tableau de variables dans lequel le texte lu est enregistré.	
	Type :	STRING (longueur max. : 255)
	Si le nombre de lignes indiqué dans le paramètre <nombre de lignes> est inférieur à la dimension du tableau [<n>,<m>] de la variable de résultat, les autres éléments du tableau ne sont pas modifiés. Les caractères de commande de fin de ligne "LF" (Line Feed) ou "CR LF" (Carrige Return Line Feed) ne sont **pas** enregistrés dans la variable de résultat. Si la longueur de la ligne est supérieure à la longueur de chaîne définie, les lignes lues sont tronquées. Il n'y a pas de signalisation d'erreur.	

Remarque

La lecture dans un fichier binaire est impossible. L'erreur "Type de fichier incorrect" (valeur de retour de la variable d'erreur = 4) est signalée. Les types de fichiers suivants ne sont pas lisibles : _BIN, _EXE, _OBJ, _LIB, _BOT, _TRC, _ACC, _CYC, _NCK.

Exemple

Code de programme	Commentaire
N10 DEF INT ERROR	; Définition des variables d'erreur.
N20 DEF STRING[255] RESULT[5]	; Définition des variables de résultat.
N30 READ(ERROR,"/_N_CST_DIR/_N_TESTFILE_MPF",1,5,RESULT)	; Nom du fichier avec identificateur de domaine, extension de fichier et chemin d'accès.
N40 IF ERROR <>0	; Analyse des erreurs.
N50 MSG("ERREUR"<<ERROR<<"POUR INSTRUCTION READ")	
N60 M0	
N70 ENDIF	
...	

3.5.4 Vérification de l'existence d'un fichier (ISFILE)

L'instruction ISFILE permet de vérifier si un fichier existe dans la mémoire de programmes.

Syntaxe

```
<Résultat>=ISFILE("<nom de fichier>")
```

Signification

ISFILE:	Instruction pour vérifier l'existence d'un fichier		
<nom de fichier>:	Nom du fichier dont l'existence doit être vérifiée.		
	Type :	STRING	
	Le chemin absolu peut être indiqué avant le nom de fichier proprement dit. En l'absence de chemin d'accès, le fichier est recherché dans le répertoire courant (= répertoire du programme sélectionné). Pour les règles concernant le chemin d'accès, voir "Adressage des fichiers de la mémoire de programmes (Page 567)".		
<résultat>:	Variable de résultat pour l'acquisition du résultat de la vérification		
	type.	BOOL	
	Valeur :	TRUE	Fichier existant
		FALSE	Fichier inexistant

Exemples

Exemple 1

```
Code de programme                          Commentaire
N10 DEF BOOL RESULT                        ; Définition des variables de résultat.
N20 RESULT=ISFILE("TESTFILE")
N30 IF(RESULT==FALSE)
N40    MSG("FICHIER INEXISTANT")
N50    M0
N60 ENDIF
...
```

Exemple 2

```
Code de programme                          Commentaire
N10 DEF BOOL RESULT                        ; Définition des variables de résultat.
N20 RESULT=ISFILE("TESTFILE")
N30 IF(NOT ISFILE("TESTFILE"))
N40    MSG("FICHIER INEXISTANT")
N50    M0
N60 ENDIF
...
```

3.5.5 Lecture des informations fichier (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO)

Les instructions FILEDATE, FILETIME, FILESIZE, FILESTAT et FILEINFO permettent de lire des informations fichier telles que date/heure du dernier accès en écriture, taille actuelle du fichier, état du fichier ou l'ensemble de ces informations.

Condition

Le niveau de protection actuellement paramétré doit être égal ou supérieur au droit show du répertoire principal. Si ce n'est pas le cas, l'accès est refusé avec émission du message d'erreur (valeur de retour de la variable d'erreur = 13).

Syntaxe

```
FILE....(<erreur>,"<nom de fichier>",<résultat>)
```

Signification

FILEDATE:	Fournit la **date** du dernier accès en écriture à un fichier		
FILETIME:	Fournit l'**heure** du dernier accès en écriture à un fichier		
FILESIZE:	Fournit la **taille actuelle** d'un fichier		
FILESTAT:	Fournit l'**état** d'un fichier se rapportant aux **droits** suivants : • Lecture (r : read) • Ecriture (w : write) • Exécution (x : execute) • Affichage (s : show) • Suppression (d : delete) **Remarque :** Ces niveaux de protection sont des propriétés spéciales du système de fichiers passif. Ainsi, lors de l'accès à des mémoires de programmes externes, FILESTAT ne fournit que les droits d'accès par défaut (77777).		
FILEINFO:	Fournit, pour un fichier, l'**ensemble des informations** lisibles via FILEDATE, FILETIME, FILESIZE et FILESTAT.		
<erreur> :	Variable pour la valeur d'erreur en retour (paramètre Call-By-Reference)		
	type.	VAR INT	
	Valeur :	0	Aucune erreur
		1	Chemin d'accès non autorisé
		2	Chemin d'accès introuvable
		3	Fichier introuvable
		4	Type de fichier incorrect
		13	Droits d'accès insuffisants
		22	Longueur de chaîne de la variable de résultat (<résultat>) trop petite.
<nom de fichier> :	Nom du fichier dont les informations de fichier doivent être lues		
	Type :	CHAR[160]	
	Le chemin absolu peut être indiqué avant le nom de fichier proprement dit. En l'absence de chemin d'accès, le fichier est recherché dans le répertoire courant (= répertoire du programme sélectionné). Pour les règles concernant le chemin d'accès, voir "Adressage des fichiers de la mémoire de programmes (Page 567)".		

<résultat> :	Variable de résultat (paramètre Call-By-Reference) Variable dans laquelle l'information de fichier interrogée est enregistrée.			
	Type :	VAR CHAR[8]	pour	FILEDATE Format : "dd.mm.yy"
		VAR CHAR[8]	pour	FILETIME Format : "hh:mm.ss"
		VAR INT	pour	FILESIZE La taille du fichier est indiquée en octets.
		VAR CHAR[5]	pour	FILESTAT Format : "rwxsd" (r : **r**ead, w : **w**rite, x : e**x**ecute, s : **s**how, d : **d**elete)
		VAR CHAR[32]	pour	FILEINFO Format : "rwxsd nnnnnnnn dd.mm.yy hh:mm:ss"

Exemple

Code de programme	**Commentaire**
N10 DEF INT ERROR	; Définition des variables d'erreur.
N20 STRING[32] RESULT	; Définition des variables de résultat.
N30 FILEINFO(ERROR,"/_N_MPF_DIR/_N_TESTFILE_MPF",RESULT)	; Nom du fichier avec identificateur de domaine, extension de fichier et chemin d'accès.
N40 IF ERROR <> 0	; Analyse des erreurs
N50 MSG("ERREUR"<<ERROR<<"SUR L'INSTRUCTION FILEINFO")	
N60 M0	
N70 ENDIF	
...	

L'exemple pourrait p. ex. fournir le résultat suivant dans la variable de résultat RESULT :

"77777 12345678 26.05.00 13:51:30"

3.6 Zones de protection

3.6.1 Définition des zones de protection (CPROTDEF, NPROTDEF)

Les zones de protection qui doivent protéger des éléments de machine des collisions sont définies en blocs dans le programme pièce. Ceux-ci contiennent les éléments suivants :

1. Définition du plan de travail
   Avant de définir la zone de protection à proprement parler, il convient de sélectionner le plan de travail auquel doit se rapporter la description de contour de la zone de protection.
2. Début de la définition
   Selon l'instruction CN, c'est soit une zone de protection spécifique au canal, soit une zone de protection spécifique à la machine qui est créée.
3. Description du contour de la zone de protection
   Le contour d'une zone de protection est décrit à l'aide de mouvements de déplacement. Ceux-ci ne sont pas exécutés et n'ont aucun lien avec les descriptions géométriques précédentes ou suivantes. Ils définissent exclusivement la zone de protection.
4. Fin de la définition

Syntaxe

```
DEF INT <Var>
G17/G18/G19
CPROTDEF/NPROTDEF(<n>,<t>,<AppLim>,<AppPlus>,<AppMinus>)
G0/G1/... X/Y/Z...
...
EXECUTE(<Var>)
```

Signification

`DEF INT <Var>` :	Définition d'une variable auxiliaire locale du type de données INTEGER	
`<Var>` :	Nom de la variable auxiliaire	
`G17/G18/G19` :	Plan de travail **Remarque :** Le plan de travail ne doit pas être modifié avant la fin de la définition. La programmation de la cote entre le début et la fin de la définition n'est pas admissible.	
`CPROTDEF()` :	Procédure prédéfinie pour la définition d'une zone de protection spécifique au **canal**	
`NPROTDEF()` :	Procédure prédéfinie pour la définition d'une zone de protection spécifique à la **machine**	
`<n>` :	Numéro de la zone de protection définie	
	Type de données :	INT

<t> :	Type de la zone de protection		
	Type de données :	BOOL	
	Valeur :	TRUE	zone de protection orientée **outil**
		FALSE	zone de protection orientée **pièce**
<AppLim> :	Type de limitation dans la 3e dimension		
	Type de données :	INT	
	Valeur :	0	aucune limitation
		1	limitation dans le sens plus
		2	limitation dans le sens moins
		3	limitation dans le sens positif et négatif
<AppPlus> :	Valeur de la limitation dans le sens positif de la 3e dimension		
	Type de données :	REAL	
<AppMinus> :	Valeur de la limitation dans le sens négatif de la 3e dimension		
	Type de données :	REAL	

`G0/G1/... X/Y/Z... ...:`	On peut décrire le contour d'une zone de protection avec 11 déplacements au maximum dans le plan de travail sélectionné. Le premier déplacement est constitué par l'accostage du contour. Le dernier point et le premier point de la description du contour doivent toujours coïncider. La zone se trouvant à gauche du contour correspond à la zone de protection. • Zone de protection interne Le contour pour une zone de protection interne doit être décrit dans le sens antihoraire. • Zone de protection externe (seulement admissible pour les zones de protection orientées pièce !) Le contour pour une zone de protection externe doit être décrit dans le sens horaire. Les éléments de contour suivants sont admissibles : • `G0` et `G1` pour les parties rectilignes du contour • `G2` pour les arcs de cercle dans le sens horaire Seulement admissible pour les zones de protection orientées pièce ! Non admissible pour les zones de protection orientées pièce, compte tenu qu'elles doivent être que convexes. • `G3` pour les arcs de cercle dans le sens antihoraire **Remarque :** Une zone de protection ne peut pas être décrite par un cercle complet. Un cercle complet doit être segmenté en deux cercles partiels. **Remarque :** La séquence `G2` → `G3` ou `G3` → `G2` n'est pas admissible ! Un bloc `G1` court doit être inséré entre les deux blocs à interpolation circulaire.
`EXECUTE(<Var>) :`	Procédure prédéfinie qui marque la fin de la définition. `EXECUTE` permet de revenir à l'exécution normale du programme.

Exemple

Voir exemple sous "Activation/désactivation des zones de protection (CPROT, NPROT) (Page 592)".

informations complémentaires

Zones de protection spécifiques à la machine

Une zone de protection spécifique à la machine (soit son contour) est définie au moyen des axes géométriques, c'est-à-dire par rapport au système de coordonnées de base (SCB) d'un canal. Pour assurer une surveillance correcte de la zone de protection dans tous les canaux où la zone de protection spécifique à la machine est active, le système de coordonnées de base (SCB) de tous les canaux concernés doit être identique :

- Position de l'origine système de coordonnées par rapport à l'origine machine
- Orientation de l'axe de coordonnées

Point de référence de la description du contour

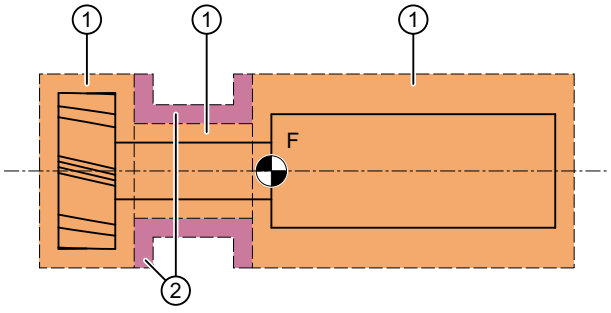
- Zones de protection orientées outil
  Les coordonnées pour les zones de protection **orientées outil** doivent être spécifiées sous forme de coordonnées absolues, rapportées au **point de référence F du porte-outil**.
- Zones de protection orientées pièce
  Les coordonnées pour les zones de protection **orientées pièce** doivent être spécifiées sous forme de coordonnées absolues, rapportées à l'origine du **système de coordonnées de base (SCB)**.

Zones de protection à symétrie de révolution

Dans le cas de zones de protection à symétrie de révolution (par ex. pour un mandrin), l'ensemble du contour doit être décrit, et pas seulement le contour jusqu'au centre de rotation.

Zones de protection orientées outil

Les zones de protection orientées outil doivent toujours être convexes. S'il faut une zone de protection concave, il faut la fractionner en plusieurs zones de protection convexes.



① Zones de protection convexes
② Zones de protection concaves (**non admissibles !**)
F Point de référence du porte-outil

Autres conditions

Pendant la définition d'une zone de protection, les fonctions suivantes ne doivent pas être activés ou utilisées :

- Correction du rayon d'outil (correction du rayon de fraise, correction du rayon du tranchant)
- Transformation
- Accostage de point de référence (`G74`)
- Accostage de point fixe (`G75`)
- Arrêt temporisé (`G4`)
- Arrêt de recherche de blocs (`STOPRE`)
- Fin du programme (`M17`, `M30`)
- Fonctions M : `M0`, `M1`, `M2`

3.6.2 Activation/désactivation des zones de protection (CPROT, NPROT)

Les zones de protection préalablement définies dans le programme pièce peuvent être activées à tout moment ou préactivées en vue d'une activation ultérieure par le programme utilisateur de l'AP. Les zones de protection actives peuvent être désactivées à tout moment.

Lors de l'activation ou de la pré-activation, il est possible en outre de décaler le point de référence de la zone de protection de manière relative.

Remarque

Une zone de protection n'est prise en compte qu'après le référencement de tous les axes géométriques du canal dans lequel elle a été activée.

Remarque

Surveillance des zones de protection

Si aucune zone de protection orientée outil n'est active, la trajectoire de l'outil fera l'objet d'une surveillance rapportée aux zones de protection orientées pièce.

Si aucune zone de protection orientée pièce n'est active, il n'y aura aucune surveillance de la zone de protection.

Syntaxe

```
CPROT(<n>,<Status>,<XMov>,<YMov>,<ZMov>)
NPROT(<n>,<Status>,<XMov>,<YMov>,<ZMov>)
```

Signification

`CPROT` :	Procédure prédéfinie pour l'activation d'une zone de protection spécifique à un **canal**		
`NPROT` :	Procédure prédéfinie pour l'activation d'une zone de protection spécifique à une **machine**		
`<n>` :	Numéro de la zone de protection		
	Type de données :	INT	
`<Status>` :	Ce paramètre permet de définir l'état d'activation spécifique au canal		
	Type de données :	INT	
	Valeur :	`0`	Désactivation de la zone de protection
		`1`	Préactivation de la zone de protection
		`2`	Activation de la zone de protection
		`3`	Préactivation de la zone de protection avec arrêt conditionnel

<XMov>,<YMov>,<ZMov> :	Valeurs de décalage additives dans le sens X/Y/Z Le décalage peut être réalisé dans 1, 2 ou 3 dimensions. Les valeurs de décalage se rapportent : • à l'origine machine dans le cas d'une zone de protection orientée **pièce** • au point de référence F du porte-outil dans le cas d'une zone de protection orientée **outil**.	
	Type de données :	REAL

Exemple

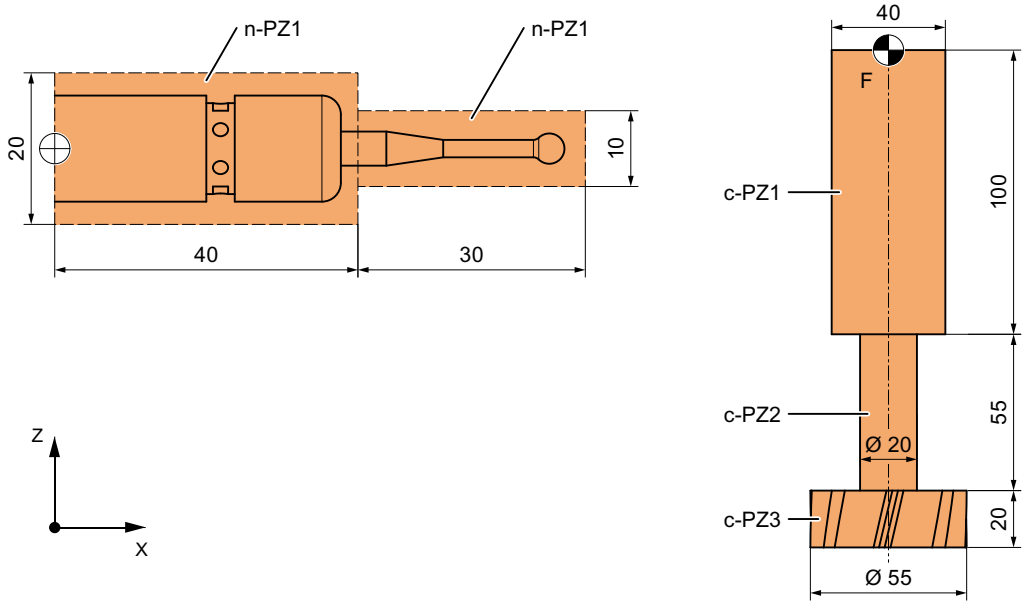
Sur une fraiseuse, il convient d'éviter une collision entre la fraise et le palpeur de mesure. La position du palpeur doit être indiquée par un décalage lors de l'activation.

A cet effet, on définit les zones de protection suivantes :

- Une zone de protection spécifique à la machine et orientée pièce pour le support du palpeur (n-PZ1) et pour le palpeur lui-même (n-PZ2).
- Une zone de protection spécifique au canal et orientée outil pour l'organe porte-fraise (c-PZ1), la queue de la fraise (c-PZ2) et la fraise elle-même (c-PZ3).

Toutes les zones de protection sont orientées en Z.

Le point de référence du palpeur de mesure doit se trouver en X = -120, Y = 60 et Z = 80 lors de l'activation.



① Point de référence pour la zone de protection du palpeur

F Point de référence du porte-outil

Code de programme	Commentaire
DEF INT PROTZONE	; Définition d'une variable auxiliaire

Code de programme	**Commentaire**
G17	; Plan de travail XY
; Définition des zones de protection :	
NPROTDEF(1,FALSE,3,10,-10)	; Zone de protection n-PZ1
G01 X0 Y-10	
X40	
Y10	
X0	
Y-10	
EXECUTE(PROTZONE)	
NPROTDEF(2,FALSE,3,5,-5)	; Zone de protection n-PZ2
G01 X40 Y-5	
X70	
Y5	
X40	
Y-5	
EXECUTE(PROTZONE)	
CPROTDEF(1,TRUE,3,0,-100)	; Zone de protection c-PZ1
G01 X-20 Y-20	
X20	
Y20	
X-20	
Y-20	
EXECUTE(PROTZONE)	
CPROTDEF(2,TRUE,3,-100,-150)	; Zone de protection c-PZ2
G01 X0 Y-10	
G03 X0 Y10 J10	
X0 Y-10 J-10	
EXECUTE(PROTZONE)	
CPROTDEF(3,TRUE,3,-150,-170)	; Zone de protection c-PZ3
G01 X0 Y-27,5	
G03 X0 Y27,5 J27,5	
X0 Y27,5 J-27,5	
EXECUTE(PROTZONE)	
; Activation des zones de protection :	
NPROT(1,2,-120,60,80)	; Activer la zone de protection n-PZ1 avec décalage
NPROT(2,2,-120,60,80)	; Activer la zone de protection n-PZ2 avec décalage
CPROT(1,2,0,0,0)	; Activer la zone de protection c-PZ1
CPROT(2,2,0,0,0)	; Activer la zone de protection c-PZ2
CPROT(3,2,0,0,0)	; Activer la zone de protection c-PZ3

Informations complémentaires

État d'activation après démarrage de la commande

Une zone de protection peut être active tout de suite après le démarrage de la commande et le référencement des axes. C'est le cas lorsque la variable système suivante est définie sur TRUE pour la zone de protection :

- $SN_PA_ACTIV_IMMED[<n>] (pour une zone de protection spécifique à la machine) ou
- $SC_PA_ACTIV_IMMED[<n>] (pour une zone de protection spécifique au canal)
  L'indice "<n>" correspond au numéro de la zone de protection : 0 = 1re zone de protection

La zone de protection est activée avec l'état = 2 et sans décalage.

Activation multiple d'une zone de protection

Une zone de protection spécifique à la machine peut être activée dans plusieurs canaux à la fois (par ex. : la zone de protection d'un fourreau de contre-poupée quand deux chariots se font face). La surveillance des zones de protection se fait uniquement si la prise de référence a été effectuée dans tous les axes géométriques.

Une zone de protection ne peut pas être activée simultanément avec différents décalages dans un même canal.

Surveillance de zone de protection avec correction du rayon d'outil active

Lorsque la correction du rayon d'outil est active, une surveillance de zone de protection opérationnelle n'est possible que si le plan de la correction du rayon d'outil est identique au plan des définitions de zones de protection.

3.6.3 Contrôle des dépassements de zone de protection, des limitations de la zone de travail et des fins de course logicielles (CALCPOSI)

Fonction

La fonction CALCPOSI vérifie dans le système de coordonnées pièce (SCP) si, en partant de la position de départ, les **axes géométriques** peuvent parcourir une trajectoire donnée sans dépasser les limitations actives. Dans le cas où le déplacement ne pourrait pas être exécuté sur l'ensemble de la distance à parcourir en raison de limitations, une valeur d'état positive codée en décimal et la distance maximale à parcourir possible sont retournées.

Définition

```
INT CALCPOSI(VAR REAL[3] <Start>, VAR REAL[3] <Dist>, VAR REAL[5]
<Limit>, VAR REAL[3] <MaxDist>, BOOL <MeasSys>, INT <TestLim>)
```

Syntaxe

```
<Status> = CALCPOSI(VAR <Start>, VAR <Dist>, VAR <Limit>, VAR
<MaxDist>, <MeasSys>, <TestLim>)
```

Signification

`CALCPOSI(...)` :	Fonction prédéfinie pour le test portant sur les dépassements de limitations par rapport aux axes géométriques		
	Arrêt du prétraitement des blocs :		non
	Uniquement dans le bloc :		oui
`<Status>` : (Partie 1)	Valeur de retour de la fonction. Les valeurs négatives indiquent des états d'erreur.		
	Type de données :		INT
	Plage de valeurs :		-8 ≤ x ≤ 100 000
	Valeur :	0	Le déplacement peut être exécuté sur l'ensemble de la distance à parcourir
		-1	Au moins une composante est négative dans <Limit>.
		-2	Erreur lors d'un calcul de transformation. Exemple : La distance à parcourir comprend un point singulier sur lequel les positions d'axe ne sont pas définies.
		-3	La distance à parcourir spécifiée <Dist> et la distance à parcourir maximale possible <MaxDist> sont linéairement dépendantes. **Remarque** Ne peut se produire qu'en relation avec <TestLim>, bit 4 == 1.
		-4	La projection du sens de déplacement contenu dans <Dist> sur la surface de délimitation est un vecteur nul, autrement dit le sens de déplacement est perpendiculaire à la surface de limitation dépassée. **Remarque** Ne peut se produire qu'en relation avec <TestLim>, bit 5 == 1.
		-5	Dans <TestLim>, bit 4 == 1 ET bit 5 == 1
		-6	Au moins un axe machine, qui doit être pris en compte pour la vérification des limites de déplacement, n'est pas référencé.
		-7	Fonction de prévention des collisions : Définition non valide de la chaîne cinématique ou des zones de protection.
		-8	Fonction de prévention des collisions : La fonction ne peut pas être exécutée à cause d'une capacité de mémoire insuffisante.

<Status> : (Partie 2)	Position des unités **Remarque** Si plusieurs limites sont dépassées simultanément, celle entraînant la plus forte restriction de la distance à parcourir spécifiée est signalée.		
	Valeur :	1	Distance à parcourir restreinte par des fins de course logicielles
		2	Distance à parcourir restreinte par une limitation de la zone de travail
		3	Distance à parcourir restreinte par des zones de protection
		4	Fonction de prévention des collisions : Distance à parcourir restreinte par des zones de protection
	Position des dizaines		
	Valeur :	1x	Limite dépassée par la valeur initiale
		2x	Limite dépassée par la droite spécifiée La valeur de retour correspond également à cette valeur lorsque le point final en soi ne dépasse aucune limite, mais que le déplacement du point de départ au point final provoquerait le dépassement d'une limite (par exemple la traversée d'une zone de protection ou des fins de course logicielles courbés dans le SCP dans le cas de transformations non linéaires telles que Transmit).
<Status> : (Partie 3)	Position des centaines		
	Valeur :	1xx	ET position des unités == 1 ou 2 : Dépassement de la valeur limite positive.
			ET position des unités == 3[1] : Dépassement d'une zone de protection spécifique au canal.
		2xx	ET position des unités == 1 ou 2 : Dépassement de la valeur limite négative.
			ET position des unités == 3[1] : Violation d'une zone de protection spécifique au canal.
<Status> : (Partie 4)	Position des milliers		
	Valeur :	1xxx	ET position des unités == 1 ou 2 : Facteur multipliant le numéro de l'axe qui dépasse la limite. Le comptage des axes commence par 1. Référence : • Fin de course logiciel : axes machine • Limitation de la zone de travail : axes géométriques ET position des unités == 3[1] : Facteur multipliant le numéro de la zone de protection dépassée.

<table>
<tr><td rowspan="4"><Status> :
(Partie 5)</td><td colspan="3">Position des centaines de milliers</td></tr>
<tr><td rowspan="3">Valeur :</td><td>0xxxxx</td><td>Position des centaines de milliers == 0 : <Dist> reste inchangé</td></tr>
<tr><td>1xxxxx</td><td>Un vecteur de direction définissant le sens de la poursuite du déplacement sur la surface de limitation est retourné dans <Dist>.
Ne peut se produire qu'avec les autres conditions suivantes :
• Dépassement de fin de course logiciel ou de limitation de la zone de travail (pas au point de départ)
• Une transformation n'est pas active
• <TestID>, bit 4 ou bit 5 == 1</td></tr>
<tr><td></td><td></td></tr>
<tr><td><Start> :</td><td colspan="3">Référence à un vecteur avec les positions de départ :
• <Start> [0] : 1. axe géométrique
• <Start> [1] : 2. axe géométrique
• <Start> [2] : 3. axe géométrique</td></tr>
<tr><td></td><td colspan="2">Type de paramètre :</td><td>Entrée</td></tr>
<tr><td></td><td colspan="2">Type de données :</td><td>VAR REAL[3]</td></tr>
<tr><td></td><td colspan="2">Plage de valeurs :</td><td>Valeur Valeur REAL ≤ x[<n>] ≤ +max. REAL +max.</td></tr>
<tr><td><Dist> :</td><td colspan="3">Référence à un vecteur.</td></tr>
<tr><td></td><td colspan="3">Entrée : distance à parcourir relative
• <Dist> [0] : 1. axe géométrique
• <Dist> [1] : 2. axe géométrique
• <Dist> [2] : 3. axe géométrique</td></tr>
<tr><td></td><td colspan="3">Sortie (uniquement avec position des centaines de milliers activée dans <Status>) :</td></tr>
<tr><td></td><td></td><td colspan="2"><Dist> contient un vecteur unité v en tant que valeur de sortie, qui définit la poursuite du sens de déplacement dans le SCP.</td></tr>
<tr><td></td><td></td><td colspan="2">Cas 1 : formation du vecteur v pour <TestID>, bit 4 == 1
Les vecteurs d'entrée <Dist> et <MaxDist> définissent le plan de déplacement. Ce plan est coupé par la surface de délimitation dépassée. La droite d'intersection des deux plans définit la direction du vecteur v. L'orientation (le signe) est alors choisie de telle sorte que l'angle entre le vecteur d'entrée <MaxDist> et v ne dépasse pas 90 degrés.</td></tr>
<tr><td></td><td></td><td colspan="2">Cas 2 : formation du vecteur v pour <TestID>, bit 5 == 1
Le vecteur v est le vecteur unité en direction de la projection du vecteur de déplacement contenu dans <Dist> sur la surface de délimitation. Une erreur est retournée si la projection du vecteur de déplacement sur la surface de délimitation est un vecteur nul.</td></tr>
<tr><td></td><td colspan="2">Type de paramètre :</td><td>Entrée/sortie</td></tr>
<tr><td></td><td colspan="2">Type de données :</td><td>VAR REAL[3]</td></tr>
<tr><td></td><td colspan="2">Plage de valeurs :</td><td>Valeur Valeur REAL ≤ x[<n>] ≤ +max. REAL +max.</td></tr>
</table>

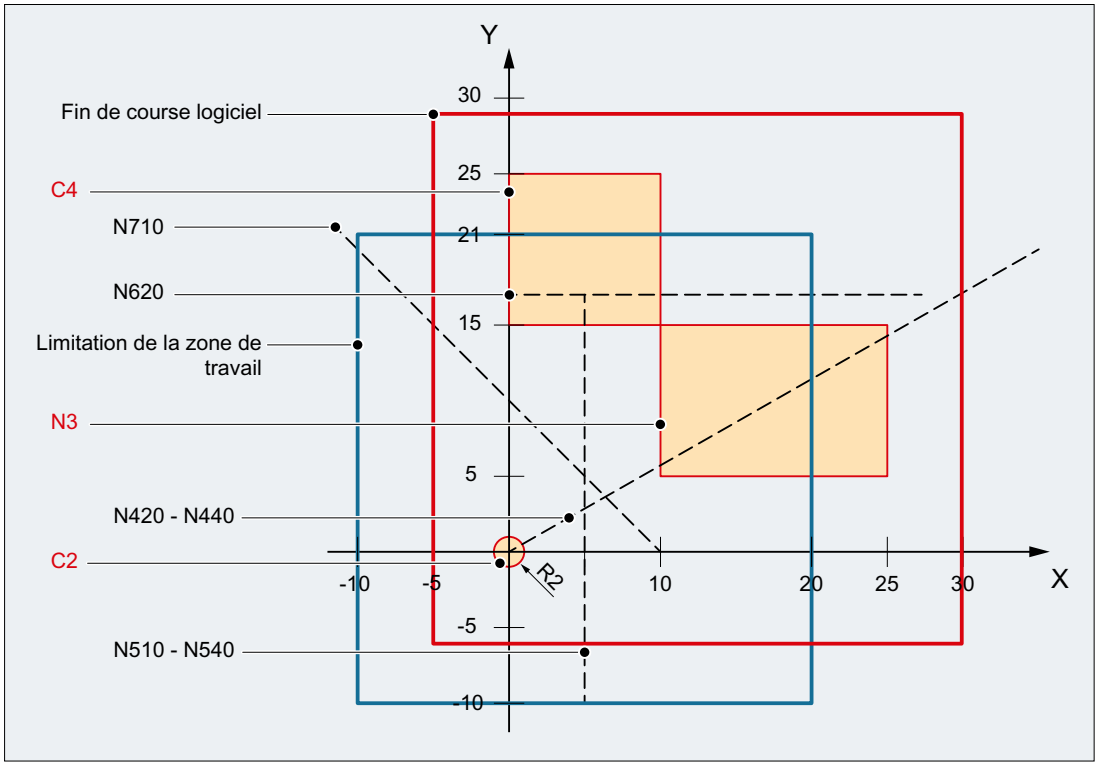
`<Limit>` :	Référence à une zone de longueur 5. • <Limit> [0 - 2] : distances minimales des axes géométriques aux limitations : – <Limit> [0] : 1. axe géométrique – <Limit> [1] : 2. axe géométrique – <Limit> [2] : 3. axe géométrique Les distances minimales sont respectées pour : – Limitation de la zone de travail : Sans restriction – Fin de course logiciel : Si aucune transformation n'est active, ou si une transformation est active pour laquelle une affectation univoque des axes géométriques à des axes machine linéaires est possible, par ex. transformations à 5 axes. • <Limit> [3] : Contient la distance minimale pour les axes machine linéaires qui ne peuvent être affectés à aucun axe géométrique à cause d'une transformation non linéaire, par exemple. Cette valeur est en outre utilisée en tant que valeur limite pour la surveillance des zones de protection conventionnelles et des zones de protection de la prévention des collisions. • <Limit> [4] : Contient la distance minimale pour les axes machine rotatifs qui ne peuvent être affectés à aucun axe géométrique en raison d'une transformation non linéaire, par exemple. **Remarque** Cette valeur prend effet uniquement lors de la surveillance des fins de course logicielles de transformations spécifiques.	
	Type de paramètre :	Entrée
	Type de données :	VAR REAL[5]
	Plage de valeurs :	Valeur REAL -max. ≤ x[n] ≤ valeur REAL +max.
`<MaxDist>` :	Référence à un vecteur avec la distance à parcourir relative pour lequel la distance minimale spécifiée d'une limite d'axe n'est dépassée par aucun de l'ensemble des axes machine concernés : • <Dist> [0] : 1. axe géométrique • <Dist> [1] : 2. axe géométrique • <Dist> [2] : 3. axe géométrique Si la distance à parcourir n'est pas limitée, le contenu de ce paramètre de retour est identique à celui de <Dist>.	
	Pour <TestID>, bit 4 == 1 : <Dist> et <MaxDist> <MaxDist> et <Dist> doivent contenir en tant que valeurs d'entrée des vecteurs qui définissent un plan de déplacement. Les deux vecteurs doivent être linéairement dépendants l'un de l'autre. La valeur absolue de <MaxDist> est quelconque. Pour le calcul du sens de déplacement, voir la description de <Dist>.	
	Type de paramètre :	Sortie
	Type de données :	VAR REAL[3]
	Plage de valeurs :	Valeur Valeur REAL ≤ x[<n>] ≤ +max. REAL +max.

<MeasSys> :	Système de mesure (anglo-saxon / métrique) pour les indications de position et de longueur (**facultatif**)		
	Type de données :		BOOL
	Valeur :	FALSE (Par défaut)	Système de mesure correspondant à l'instruction G actuellement active du groupe G 13 (G70, G71, G700, G710). **Remarque** Lorsque G70 est active et que le système de base est métrique ou que G71 est active et que le système de base est anglo-saxon, les variables système $AA_IW et $AA_MW sont fournies dans le système de base et doivent être converties le cas échéant lorsqu'elles sont utilisées pour CALCPOSI.
		TRUE	Système de mesure selon le système de base réglé : PM52806 $MN_ISO_SCALING_SYSTEM
<TestLim> :	Sélection codée sur bits des limitations à surveiller (**facultatif**)		
	Type de données :		INT
	Valeur par défaut :		Bit 0, 1, 2, 3, 6, 7 == 1 (207)
	Bit	**Décimale**	**Signification**
	0	1	Fin de course logiciel
	1	2	Limitation de la zone de travail
	2	4	Zones de protection conventionnelles activées
	3	8	Zones de protection conventionnelles pré-activées
	4	16	En cas de dépassement de fins de course logicielles ou de limitations de zones de travail dans <Dist>, fournir en retour le sens de déplacement selon le **cas 1** (voir ci-dessus).
	5	32	En cas de dépassement de fins de course logicielles ou de limitations de zones de travail dans <Dist>, fournir en retour le sens de déplacement selon le **cas 2** (voir ci-dessus).
	6	64	Zones de protection de la prévention des collisions activées
	7	128	Zones de protection de la prévention des collisions pré-activées
	8	256	Paires de zones de protection activées et pré-activées de la prévention des collisions

[1] En cas de dépassement de plusieurs zones de protection, la zone de protection signalée est celle entraînant la plus forte restriction de la distance à parcourir spécifiée.

Exemple

Limitations



L'exemple représente les fins de course logicielles et les limitations de zone de travail actives dans le plan X-Y ainsi que les trois zones de protection suivantes :

- C2 : Zone de protection spécifique au canal orientée outil, active, circulaire, rayon = 2 mm
- C4 : Zone de protection orientée pièce spécifique au canal, pré-activée, carrée, longueur des côtés = 10 mm
- N3 : Zone de protection spécifique à la machine, active, rectangulaire, longueur des côtés = 10 mm x 15 mm

Programme CN

Dans un premier temps, les zones de protection et les limitations de zone de travail sont définies dans le programme CN. La fonction `CALCPOSI()` est ensuite appelée avec différents paramétrages.

Code de programme

```
N10 DEF REAL _START[3]
N20 DEF REAL _DIST[3]
N30 DEF REAL _LIMIT[5]
N40 DEF REAL _MAXDIST[3]
N50 DEF INT _PA
N60 DEF INT _STATUS
```

Code de programme

```
; Zone de protection C2 relative à l'outil
N70 CPROTDEF(2, TRUE, 0)
N80 G17 G1 X-2 Y0
N90 G3 I2 X2
N100 I-2 X-2
N110 EXECUTE(_PA)
; Zone de protection C4 relative à la pièce
N120 CPROTDEF(4, FALSE, 0)
N130 G17 G1 X0 Y15
N140 X10
N150 Y25
N160 X0
N170 Y15
N180 EXECUTE(_PA)
; Zone de protection N3 spécifique à la machine
N190 NPROTDEF(3, FALSE, 0)
N200 G17 G1 X10 Y5
N210 X25
N220 Y15
N230 X10
N240 Y5
N250 EXECUTE(_PA)
; Activation ou pré-activation des zones de protection
N260 CPROT(2, 2, 0, 0, 0)
N270 CPROT(4, 1, 0, 0, 0)
N280 NPROT(3, 2, 0, 0, 0)
; Définition des limitations de la zone de travail
N290 G25 XX=-10 YY=-10
N300 G26 XX=20 YY=21
N310 _START[0] = 0.
N320 _START[1] = 0.
N330 _START[2] = 0.
N340 _DIST[0] = 35.
N350 _DIST[1] = 20.
N360 _DIST[2] =  0.
N370 _LIMIT[0] = 0.
N380 _LIMIT[1] = 0.
N390 _LIMIT[2] = 0.
N400 _LIMIT[3] = 0.
N410 _LIMIT[4] = 0.
N420 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST)
N430 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,3)
N440 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,1)
N450 _START[0] =  5.
N460 _START[1] = 17.
N470 _START[2] =  0.
N480 _DIST[0] =  0.
N490 _DIST[1] =-27.
N500 _DIST[2] =  0.
N510 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,14)
N520 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,, 6)
N530 _LIMIT[1] = 2.
```

Code de programme

```
N540 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,, 6)
N550 _START[0] = 27.
N560 _START[1] = 17,1
N570 _START[2] =  0.
N580 _DIST[0] =-27.
N590 _DIST[1] =  0.
N600 _DIST[2] =  0.
N610 _LIMIT[3] = 2.
N620 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,12)
N630 _START[0] = 0.
N640 _START[1] = 0.
N650 _START[2] = 0.
N660 _DIST[0] =  0.
N670 _DIST[1] = 30.
N680 _DIST[2] =  0.
N690 TRANS X10
N700 AROT Z45
N710 _STATUS = CALCPOSI(_START,_DIST, _LIMIT, _MAXDIST)
; Suppression des frames de N690 et N700
N720 TRANS
N730 _START[0] =  0.
N740 _START[1] = 10.
N750 _START[2] =  0.
; Les vecteurs _DIST et _MAXDIST définissent le plan de déplacement
N760 _DIST[0] = 30.
N770 _DIST[1] = 30.
N780 _DIST[2] =  0.
N790 _MAXDIST[0] = 1.
N800 _MAXDIST[1] = 0.
N810 _MAXDIST[2] = 1.
N820 _STATUS = CALCPOSI(_START, _DIST, _LIMIT, _MAXDIST,,17)
N830 M30
```

Résultats de CALCPOSI()

N...	<Status>	<MaxDist>[0] ≙ X	<MaxDist>[1] ≙ Y	Observations
420	3123	8,040	4,594	Dépassement de N3
430	1122	20,000	11,429	Aucune surveillance des zones de protection, la limitation de la zone de travail est dépassée.
440	1121	30,000	17,143	Seule la surveillance des fins de course logicielles est encore active
510	4213	0,000	0,000	Dépassement du point de départ C4
520	0000	0,000	-27,000	La zone de protection pré-activée C4 n'est pas surveillée. Le déplacement peut être exécuté sur l'ensemble de la distance spécifiée.

N...	<Status>	<MaxDist>[0] ≙ X	<MaxDist>[1] ≙ Y	Observations
540	2222	0,000	-25,000	Comme _LIMIT[1] = 2, la distance à parcourir est restreinte par la limitation de la zone de travail.
620	4223	-13,000	0,000	Du fait de C2 et de _LIMIT[3], la distance totale par rapport à C4 est de 4 mm. La distance C2 → N3 de 0,1 mm ne restreint pas la distance à parcourir.
710	1221	0,000	21,213	Frame actif avec translation et rotation. La distance à parcourir admissible dans _DIST s'applique au SCP décalé et tourné.
820	102121	18,000	18,000	Dépassement du fin de course logiciel de l'axe Y. Le réglage <_TESTLIM> = 17 permet de demander le calcul d'un sens de poursuite du déplacement. Ce sens de déplacement figure dans <_DIST> (0,707 ; 0,0 ; 0,707). Il est valide car la position des centaines de milliers est activée dans <_STATUS>.

Informations complémentaires

État de l'axe "référencé"

Tous les axes machine pris en compte par `CALCPOSI()` doivent être référencés.

Indications de distance relatives à un cercle

Toutes les indications de distance se rapportant à un cercle sont **toujours** interprétées en tant qu'indication de rayon. Ceci doit particulièrement être pris en compte pour les axes transversaux pour lesquels la programmation du diamètre est activée (`DIAMON` / `DIAM90`).

Réduction de la distance à parcourir

Si la distance à parcourir spécifiée d'un axe est limitée, la distance à parcourir des autres axes est également réduite proportionnellement dans la valeur de retour `<MaxDist>`. Le point final obtenu se trouve par conséquent toujours sur la trajectoire spécifiée.

Axes rotatifs

Les axes rotatifs sont surveillés uniquement si ce ne sont pas des axes rotatifs modulo.

L'absence de définition de fins de course logicielles, de limitations de la zone de travail ou de zones de protection pour un ou plusieurs des axes impliqués est admissible.

État des fins de course logicielles et des limitations de la zone de travail

Les fins de course logicielles et les limitations de la zone de travail sont pris en compte uniquement s'ils sont actifs lors de l'exécution de `CALCPOSI()`. L'état peut être influencé par exemple par :

- Paramètres machine : PM21020 $MC_WORKAREA_WITH_TOOL_RADIUS
- Données de réglage : $AC_WORKAREA_CS_...
- Signaux d'interface CN/AP : DB31, ... DBX12.2 / 3
- Instructions : `WALIMON` / `WALIMOF`

Fins de course logicielles et transformations

Pour CALCPOSI() en présence de différentes transformations cinématiques (par ex. TRANSMIT), les positions des axes machine (SCM) ne peuvent pas toujours être déterminées de manière univoque à partir des positions des axes géométriques (SCP) en raison d'ambiguïtés à certains endroits de la plage de déplacement. En mode de déplacement normal, l'univocité résulte en règle générale des précédents et de la condition selon laquelle un mouvement continu des axes machine dans le SCP doit correspondre à un mouvement continu dans le SCM. C'est pourquoi, lors de la surveillance des fins de course logicielles, la position de la machine présente au moment de l'exécution de CALCPOSI() est utilisée dans ce type de cas pour résoudre l'ambiguïté.

Remarque

Arrêt du prétraitement des blocs

Lorsque CALCPOSI() est utilisée en relation avec des transformations, l'utilisateur assume l'entière responsabilité quant à la programmation d'un arrêt du prétraitement des blocs (STOPRE) pour la synchronisation des positions d'axes machine avec le prétraitement des blocs avant CALCPOSI().

Distance de la zone de protection et zones de protection conventionnelles

Pour les zones de protection conventionnelles, il n'est **pas** garanti que lors d'un mouvement de déplacement sur la distance à parcourir spécifiée, la distance de sécurité renseignée dans <Limit>[3] est respectée par rapport à toutes les zones de protection. Il est seulement garanti que lors d'un prolongement du point final fourni en retour dans <Dist> correspondant à la distance de sécurité dans le sens de déplacement, aucune zone de protection n'est dépassée. Le parcours de la droite peut cependant passer plus ou moins près d'une zone de protection.

Distance de la zone de protection et zones de protection de la prévention des collisions

Pour les zones de protection de la prévention des collisions, il est garanti que lors d'un mouvement de déplacement sur la distance à parcourir spécifiée, la distance de sécurité renseignée dans le paramètre <Limit>[3] est respectée par rapport à toutes les zones de protection.

La distance de sécurité renseignée dans le paramètre <Limit>[3] prend uniquement effet lorsque :

<Limit>[3] > (PM10619 $MN_COLLISION_TOLERANCE)

Si dans le paramètre <TestLim> le bit 4 est mis à 1 (calcul du sens de la poursuite du déplacement), le vecteur de direction contenu dans <DIST> n'est valide que lorsque la position des centaines de milliers est activée dans la valeur de retour de la fonction (<Status>). Si un tel sens ne peut pas être déterminé, soit en raison d'un dépassement des zones de protection, soit parce qu'une transformation est active, la valeur d'entrée dans <DIST> reste inchangée. Aucun autre message d'erreur n'est généré.

3.7 Instructions de déplacement spéciales

3.7.1 Accostage de positions codées (CAC, CIC, CDC, CACP, CACN)

Avec les instructions de déplacement pour "l'accostage de positions codées", il est possible de déplacer des axes linéaires et rotatifs en indiquant des numéros de position sur des positions d'axe définies enregistrées dans des tableaux de paramètres machine.

Syntaxe

```
CAC(<n>)
CIC(<n>)
CACP(<n>)
CACN(<n>)
```

Signification

`CAC(<n>)` :	Accostage de la position codée du numéro de position n
`CIC(<n>)` :	Accostage de la position codée n positions en avant (+n) ou en arrière (–n), à partir du numéro de position courant
`CDC(<n>)` :	Accostage de la position codée sur le chemin le plus court depuis le numéro de position n (uniquement pour les axes rotatifs)
`CACP(<n>)` :	Accostage de la position codée dans le sens positif depuis le numéro de position n (uniquement pour les axes rotatifs)
`CACN(<n>)` :	Accostage de la position codée dans le sens négatif depuis le numéro de position n (uniquement pour les axes rotatifs)
`<n>` :	Numéro de position dans le tableau de paramètres machine Plage de valeurs : 0, 1, ... (nombre maximal de positions dans le tableau - 1)

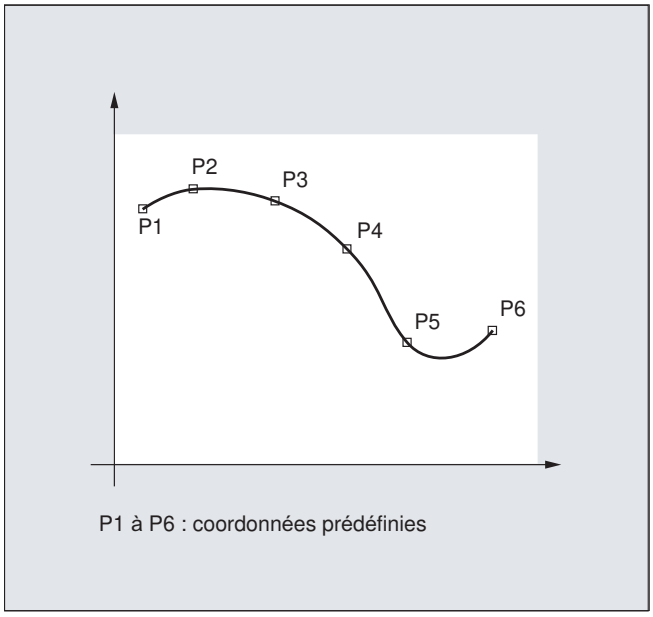
Exemple : Accostage des positions codées d'un axe de positionnement

Code de programme	**Commentaire**
`N10 FA[B]=300`	`; Avance de l'axe de positionnement B`
`N20 POS[B]=CAC(10)`	`; Accostage de la position codée du numéro de position 10`
`N30 POS[B]=CIC(-4)`	`; Accostage de la position codée du "numéro de position courant" - 4`

3.7.2 Interpolation de type spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Pour les contours de pièce de forme courbe quelconque, une description analytique exacte est impossible. C'est pourquoi de tels contours sont décrits approximativement par un nombre limité de points intermédiaires, notamment lors de la numérisation des surfaces. Pour générer une surface de pièce numérisée, il est nécessaire de relier les points intermédiaires et d'en faire une description du contour. C'est ce que permet l'interpolation de type spline.

Un spline définit une courbe composée de polynômes du 2e ou 3e degré. Les propriétés des points intermédiaires d'un spline peuvent être définis **selon le type de spline utilisé**.



SINUMERIK solution line met à disposition des types de spline suivants :

- Spline A
- Spline B
- Spline C

Syntaxe

Généralités :

```
ASPLINE X... Y... Z... A... B... C...
BSPLINE X... Y... Z... A... B... C...
CSPLINE X... Y... Z... A... B... C...
```

Le spline B permet en outre de programmer :

```
PW=<n>
SD=2
PL=<valeur>
```

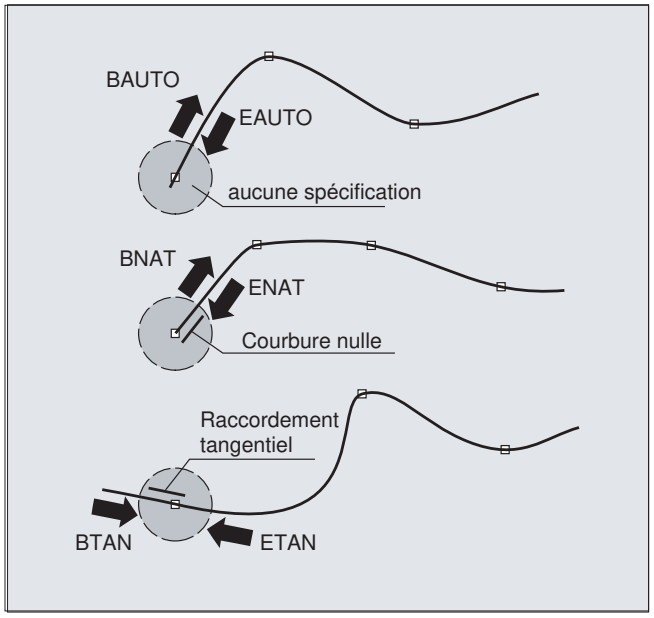
Les splines A et C permettent en outre de programmer :

`BAUTO` / `BNAT` / `BTAN`

EAUTO / ENAT / ETAN

Signification

Type d'interpolation spline :			
ASPLINE :	Instruction d'activation de l'interpolation de type spline A		
BSPLINE :	Instruction d'activation de l'interpolation de type spline B		
CSPLINE :	Instruction d'activation de l'interpolation de type spline C		
	Les instructions ASPLINE, BSPLINE et CSPLINE ont un effet modal et font partie du groupe des instructions de déplacement.		
Points intermédiaires et points de contrôle :			
X... Y... Z... A... B... C...	Positions en coordonnées cartésiennes		
Poids de point (uniquement spline B) :			
PW :	L'instruction PW permet de programmer un "poids de point" pour chaque point intermédiaire.		
<n> :	"Poids de point"		
	Plage de valeurs :	0 ≤ n ≤ 3	
	Pas :	0.0001	
	Effet :	n > 1	La courbe tend plus fortement.
		n < 1	La courbe tend moins fortement.
Degré spline (uniquement spline B) :			
SD :	Un polygone du troisième degré est appliqué de façon standard. Mais la programmation de SD=2 permet également d'utiliser un polygone du deuxième degré.		
Distance entre les pôles (uniquement spline B) :			
PL :	Les distances entre les pôles sont calculées par le système, Mais la commande peut également traiter les distances prescrites en tant que "Paramètre - Intervalle - Longueur" avec l'instruction PL.		
<valeur> :	Paramètre-Intervalle-Longueur		
	Plage de valeurs :	comme pour le positionnement non indexé	
Comportement de transition au début de la courbe spline (splines A et C uniquement) :			
BAUTO :	Aucune prescription pour le comportement de transition. Le début résulte de la position du premier point.		
BNAT :	Courbure nulle		
BTAN :	Raccordement tangentiel avec bloc précédent (fonction initialisée)		
Comportement de transition à la fin de la courbe spline (splines A et C uniquement) :			
EAUTO :	Aucune prescription pour le comportement de transition. La fin résulte de la position du dernier point.		
ENAT :	Courbure nulle		

ETAN :	Raccordement tangentiel avec bloc précédent (fonction initialisée)
	

Remarque

Le comportement de transition programmable n'influence pas le spline B. Au point de départ et au point final, le spline B est toujours tangent au polygone de contrôle.

Autres conditions à prendre en compte

- La correction du rayon d'outil est utilisable.
- La surveillance anticollision s'effectue par projection dans le plan.

Exemples

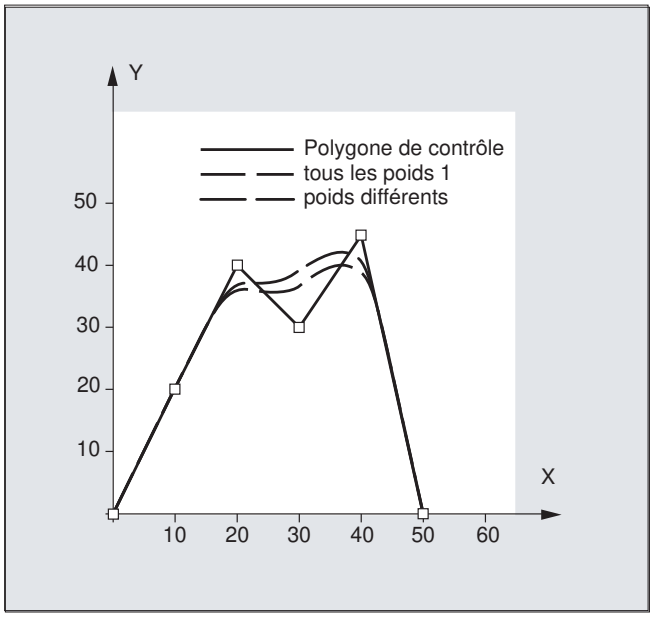
Exemple 1 : Spline B

```
Code de programme 1 (tous les poids 1)
N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20
N40 X20 Y40
N50 X30 Y30
N60 X40 Y45
N70 X50 Y0
```

Code de programme 2 (différents poids)

```
N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20 PW=2
N40 X20 Y40
N50 X30 Y30 PW=0.5
N60 X40 Y45
N70 X50 Y0
```

Code de programme 3 (polygone de contrôle)	**Commentaire**
N10 G1 X0 Y0 F300 G64	
N20	; ne s'applique pas
N30 X10 Y20	
N40 X20 Y40	
N50 X30 Y30	
N60 X40 Y45	
N70 X50 Y0	

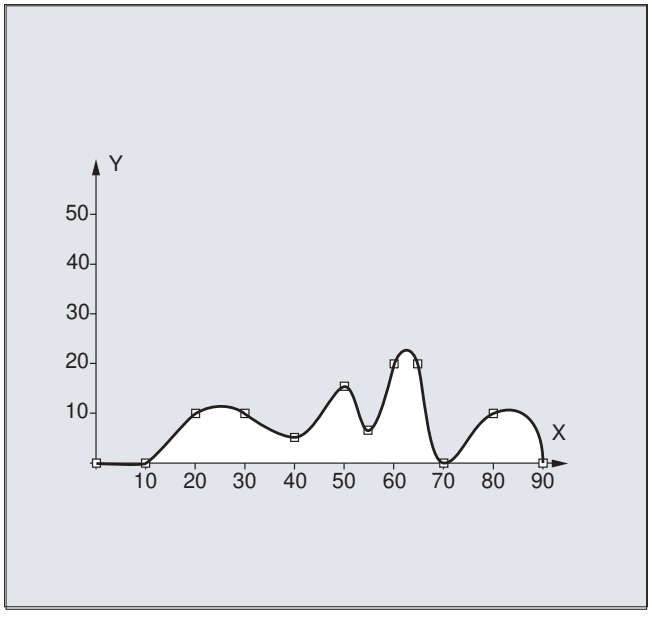


Exemple 2 : Spline C, courbure nulle au début et à la fin

Code de programme

```
N10 G1 X0 Y0 F300
N15 X10
N20 BNAT ENAT
N30 CSPLINE X20 Y10
```

```
Code de programme
N40 X30
N50 X40 Y5
N60 X50 Y15
N70 X55 Y7
N80 X60 Y20
N90 X65 Y20
N100 X70 Y0
N110 X80 Y10
N120 X90 Y0
N130 M30
```



Exemple 3 : Interpolation spline (spline A) et transformation de coordonnées (ROT)

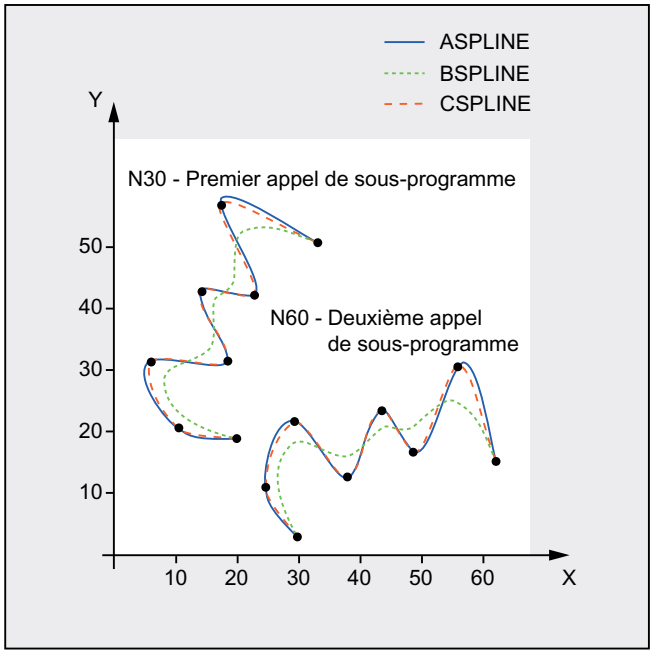
Programme principal :

Code de programme	Commentaire
N10 G00 X20 Y18 F300 G64	; Accostage du point de départ
N20 ASPLINE	; Activation de l'interpolation de type spline A.
N30 CONTOUR	; Premier appel du sous-programme.
N40 ROT Z-45	; Transformation de coordonnées : rotation du SCP de -45° autour de l'axe Z.
N50 G00 X20 Y18	; Accostage du point de départ du contour.
N60 CONTOUR	; Deuxième appel du sous-programme.
N70 M30	; Fin du programme

Sous-programme "Contour" (contenant les coordonnées des points intermédiaires) :

Code de programme
N10 X20 Y18
N20 X10 Y21
N30 X6 Y31
N40 X18 Y31
N50 X13 Y43
N60 X22 Y42
N70 X16 Y58
N80 X33 Y51
N90 M1

La figure suivante contient la courbe spline résultante de l'exemple de programme (ASPLINE), ainsi que les courbes splines qui auraient résulté d'une interpolation de type spline B ou C (BSPLINE, CSPLINE) :



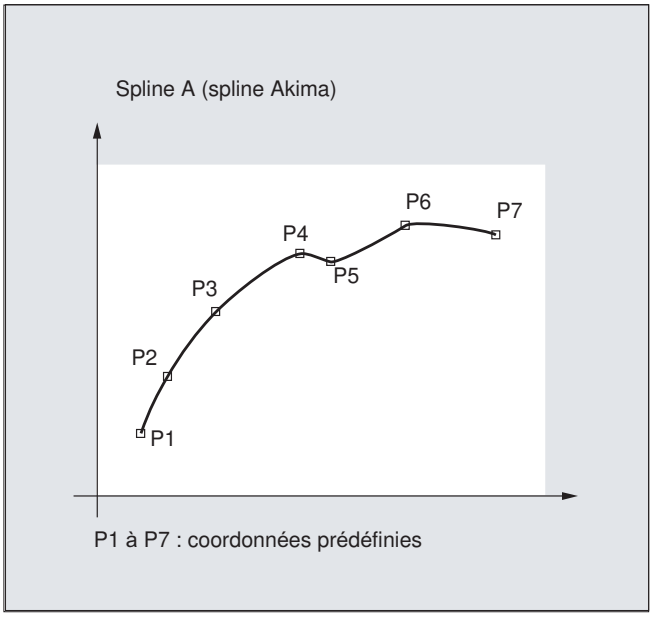
Informations complémentaires

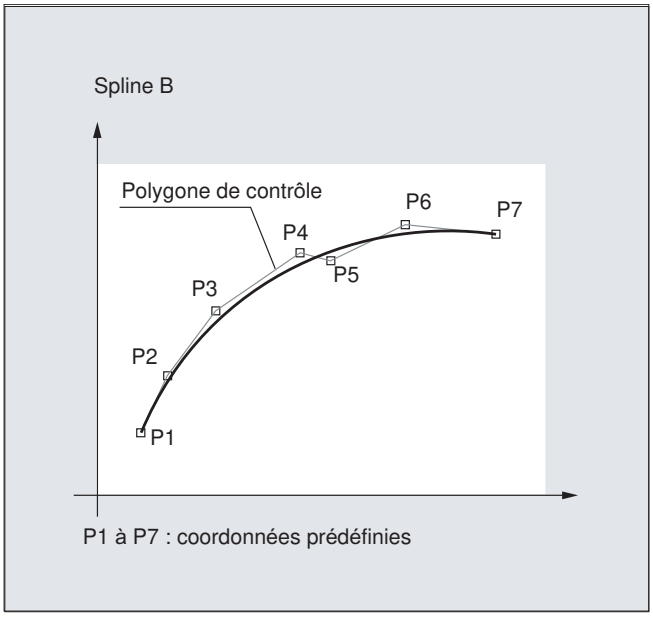
Avantages de l'interpolation de type spline

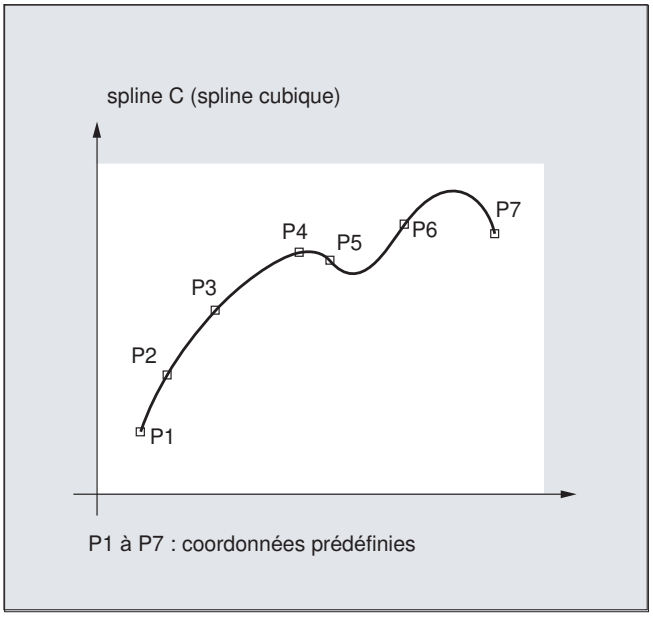
Contrairement à l'utilisation de blocs linéaires G01, l'interpolation de type spline présente les avantages suivants :

- Réduction de nombre de blocs requis dans le programme pièce pour la description du contour
- Courbes douces ménageant la mécanique lors des transitions entre les blocs du programme pièce

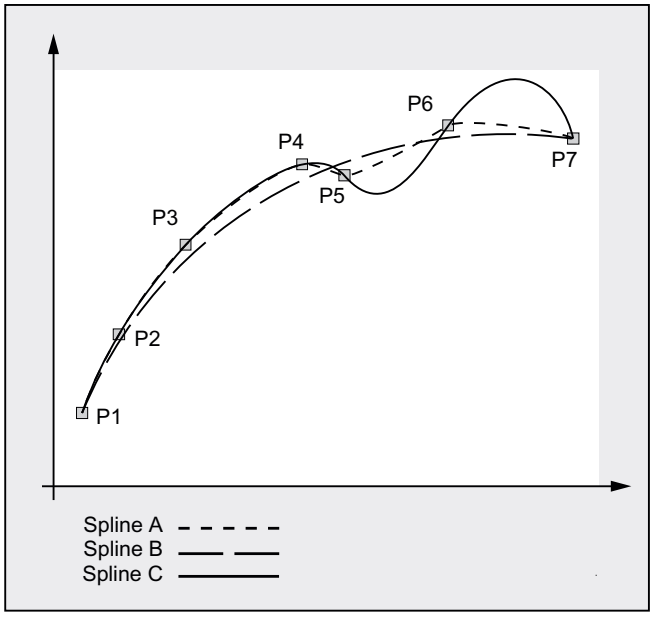
Propriétés et applications des différents types de spline

Type spline	Propriétés et applications
Spline A	 **Propriétés :** • Courbe passant exactement par les points intermédiaires indiqués. • La courbe est caractérisée par la continuité de la tangente, mais pas du rayon de courbure. • Elle ne crée presque pas de vibrations indésirables. • La modification d'un point intermédiaire a un effet local, ce qui signifie qu'elle n'influence pas plus de 6 points intermédiaires voisins. **Application :** Le spline A convient avant tout à l'interpolation de courbes avec de grandes variations de pente (par exemple les courbes en forme d'escalier).

Type spline	Propriétés et applications
Spline B	 **Propriétés :** • Courbe ne passant pas par les points intermédiaires indiqués, mais à proximité de ceux-ci. La courbe tend vers les points intermédiaires. L'utilisation d'un facteur de pondération des points intermédiaires permet en outre d'influencer le tracé de la courbe. • La courbe est caractérisée par la continuité de la tangente et du rayon de courbure. • Elle ne crée aucune vibration indésirable. • La modification d'un point intermédiaire a un effet local, ce qui signifie qu'elle n'influence pas plus de 6 points intermédiaires voisins. **Application :** Le spline B est destiné en premier lieu à servir d'interface avec les systèmes de CAO.

Type spline	Propriétés et applications
Spline C	 **Propriétés :** • Courbe passant exactement par les points intermédiaires indiqués. • La courbe est caractérisée par la continuité de la tangente et du rayon de courbure. • Elle crée fréquemment des vibrations indésirables, en particulier aux endroits avec de grandes variations de pente. • La modification d'un point intermédiaire a un effet global, ce qui signifie qu'elle influence l'ensemble du tracé de la courbe. **Application :** Le spline C convient particulièrement bien lorsque les points se situent sur une courbe analytique connue (cercle, parabole, hyperbole).

Comparaison des trois types de splines avec des points intermédiaires identiques



Nombre minimum de blocs spline

Les instructions G ASPLINE, BSPLINE et CSPLINE relient les points finaux de blocs par des courbes de type spline. Pour ce faire, la commande doit effectuer le décodage simultané d'une série de blocs (points finaux) pendant le prétraitement. En version standard, la taille du tampon pour le décodage vaut 10 blocs. Toutes les informations de bloc ne sont pas forcément des fins de bloc spline. La commande nécessite néanmoins un nombre précis de fins de bloc spline provenant de 10 blocs :

Type spline	Nombre minimum de blocs spline
spline A :	Sur 10 blocs, **4** au moins doivent être des blocs spline. Les blocs de commentaire et les calculs de paramètres ne sont pas pris en compte.
spline B :	Sur 10 blocs, **6** au moins doivent être des blocs spline. Les blocs de commentaire et les calculs de paramètres ne sont pas pris en compte.
spline C :	Le nombre minimum de blocs spline nécessaire se calcule de la manière suivante : valeur de MD20160 $MC_CUBIC_SPLINE_BLOCKS + 1 Le nombre de points à l'aide desquels la section de spline est calculée est renseigné dans MD20160. Le paramétrage par défaut est 8. Sur 10 blocs, **9** au moins doivent être des blocs spline dans le cas standard.

Remarque

Si la valeur est inférieure à la valeur tolérée, une alarme est émise, de même que si un axe participant à l'interpolation de type spline est programmé comme axe de positionnement.

Groupement de blocs spline courts

Lors de l'interpolation de type spline des blocs spline courts peuvent être créés, qui contribue inutilement à la réduction de la vitesse tangentielle. La fonction "groupement de blocs spline courts" permet de grouper ces blocs de manière que la longueur de bloc en résultant soit assez longue et à ne pas réduire la vitesse tangentielle.

La fonction est activée via le paramètre machine spécifique au canal :

MD20488 $MC_SPLINE_MODE (réglage pour interpolation de type spline)

Informations complémentaires : Description fonctionnelle Fonctions de base

3.7.3 Groupe spline (SPLINEPATH)

Les axes à interpoler dans le groupe spline sont sélectionnés avec l'instruction `SPLINEPATH`. L'interpolation de type spline admet jusqu'à 8 axes à interpolation.

Remarque

Si l'instruction `SPLINEPATH` n'est pas programmée de façon explicite, ce sont les trois premiers axes du canal qui seront déplacés en tant que groupe spline.

Syntaxe

La définition du groupe spline se fait dans un bloc séparé :

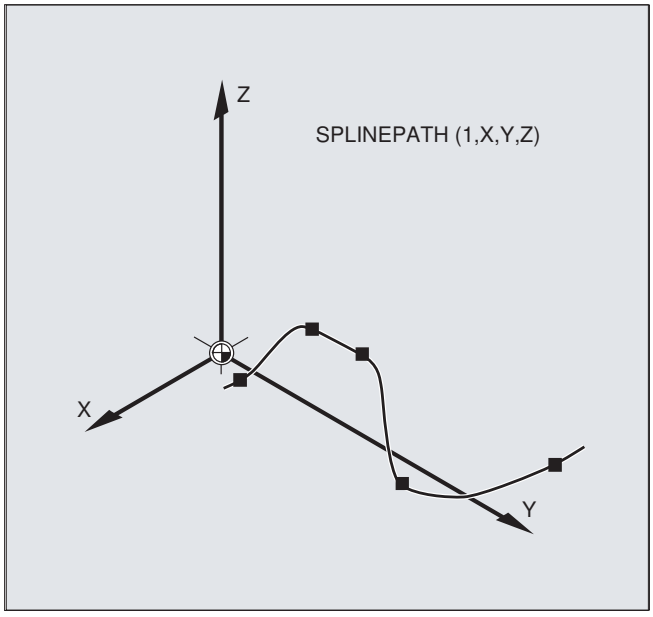
```
SPLINEPATH(n,X,Y,Z,…)
```

Signification

`SPLINEPATH` :	Instruction de définition du groupe spline
`n` :	=1 (valeur fixe)
`X,Y,Z,…` :	Descripteurs des axes à interpoler dans le groupe spline

Exemple : Groupe spline avec trois axes à interpolation

```
Code de programme                                   Commentaire
N10 G1 X10 Y20 Z30 A40 B50 F350
N11 SPLINEPATH(1,X,Y,Z)                             ; Groupe spline
N13 CSPLINE BAUTO EAUTO X20 Y30 Z40 A50 B60         ; Spline C
N14 X30 Y40 Z50 A60 B70                             ; Points intermédiaires
…
N100 G1 X… Y…                                       ; Désactivation de l'interpo-
                                                    lation de type spline
```



3.7.4 Activer/désactiver le compactage de bloc CN (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF)

Les fonctions de compactage de blocs linéaires (et aussi de blocs circulaires et/ou à vitesse rapide selon le paramétrage) sont activées/désactivées avec les instructions G du groupe G 30. Les instructions sont des fonctions à effet modal.

Syntaxe

```
COMPON / COMPCURV / COMPCAD / COMPSURF
...
COMPOF
```

Signification

COMPON :	Activation de la fonction de compresseur COMPON
COMPCURV :	Activation de la fonction de compresseur COMPCURV
COMPCAD :	Activation de la fonction de compresseur COMPCAD
COMPSURF:	Activation de la fonction de compresseur COMPSURF
COMPOF :	Désactivation de la fonction de compresseur actuellement active

Remarque

Pour une amélioration supplémentaire de l'état de surface, il est possible d'utiliser la fonction d'arrondissement G642 et la limitation des à-coups SOFT. Ces instructions doivent figurer en début de programme.

Exemple : COMPCAD

```
Code de programme              Commentaire
N10 G00 X30 Y6 Z40
N20 G1 F10000 G642             ; Activation : fonction d'arrondissement G642
N30 SOFT                       ; Activation : limitation des à-coups SOFT
N40 COMPCAD                    ; Activation : fonction de compresseur COMPCAD
N50 FIFOCTRL
N24050 Z32.499                 ; 1er bloc de déplacement
N24051 X41.365 Z32.500         ; 2ème bloc de déplacement
...
N99999 X... Z...               ; Dernier bloc de déplacement
COMPOF                         ; Fonction de compresseur désactivée.
...
```

3.7.5 Interpolation polynomiale (POLY, POLYPATH, PO, PL)

L'interpolation polynomiale (POLY) n'est en fait pas une interpolation de type spline. Elle est destinée avant tout à servir d'interface pour la programmation de courbes spline générées de manière externe. Les sections spline peuvent être programmées directement.

Ce mode d'interpolation évite à la CN de calculer les coefficients de polynôme. Elle peut être utilisée de manière optimale lorsque les coefficients sont issus directement d'un système de CAO ou d'un postprocesseur.

Syntaxe

Polynôme du 3e degré :
```
POLY PO[X]=(xe,a2,a3) PO[Y]=(ye,b2,b3) PO[Z]=(ze,c2,c3) PL=n
```

Polynôme du 5e degré et nouvelle syntaxe polynomiale :
```
POLY X=PO(xe,a2,a3,a4,a5) Y=PO(ye,b2,b3,b4,b5) Z=PO(ze,c2,c3,c4,c5)
PL=n
POLYPATH("AXES","VECT")
```

Remarque

La somme des coefficients de polynôme et des axes programmés dans un bloc CN ne doit pas dépasser le nombre maximal d'axes autorisés par bloc.

Signification

POLY :	Activation de l'interpolation polynomiale par un bloc avec POLY.
POLYPATH :	Interpolation polynomiale sélectionnable pour les deux groupes d'axes AXIS ou VECT
PO[descripteur d'axe/variable] :	Points finals et coefficients de polynôme

`X, Y, Z:`	Descripteur d'axe
`xe, ye, ze :`	Indication de la position finale de l'axe correspondant ; plage de valeurs comme pour le positionnement non indexé
`a2, a3, a4, a5:`	Les coefficients a_2, a_3, a_4, et a_5 sont écrits avec leur valeur ; plage de valeurs comme pour le positionnement non indexé. Le dernier coefficient peut être omis si sa valeur est égale à zéro.
`PL:`	Longueur de l'intervalle de paramètre sur lequel le polynôme est défini (domaine de définition de la fonction f(p)). L'intervalle commence toujours à 0 et la valeur de p peut aller de 0 à PL. Plage de valeurs théorique de PL : 0,0001 ... 99 999,9999 **Remarque :** La valeur de PL est valable pour le bloc dans lequel elle figure. Si PL n'est pas programmé, PL=1.

Activation/désactivation de l'interpolation polynomiale

L'interpolation polynomiale est activée dans le programme pièce au moyen de l'instruction G `POLY`.

L'instruction G `POLY` ainsi que `G0`, `G1`, `G2`, `G3`, `ASPLINE`, `BSPLINE` et `CSPLINE` font partie du 1er groupe G.

Etant uniquement programmés avec un nom et un point final (par ex. `X10`) les axes sont déplacés linéairement. Après avoir programmé tous les axes d'un bloc CN de la sorte, la commande se comporte comme pour G1.

L'interpolation polynomiale est de nouveau désactivée de manière implicite en programmant une autre instruction du 1er groupe G (par ex. `G0`, `G1`).

Coefficient du polynôme

La valeur `(PO[]=)` ou `...=PO(...)` indique tous les coefficients de polynôme d'un axe. Selon le degré du polynôme, plusieurs valeurs séparées par des virgules sont spécifiées. Dans un bloc, différents degrés de polynôme de différents axes sont possibles.

Sous-programme POLYPATH

`POLYPATH(...)` permet de débloquer l'interpolation polynomiale de manière sélective pour certains groupes d'axes :

Uniquement axes à interpolation et axes supplémentaires :	`POLYPATH("AXES")`
Uniquement axes d'orientation : (lors du déplacement avec transformation d'orientation)	`POLYPATH("VECT")`

Les axes respectifs non débloqués sont déplacés linéairement.

Par défaut, l'interpolation polynomiale est débloquée pour les deux groupes d'axes.

Une programmation sans paramètre `POLYPATH( )` permet de désactiver l'interpolation polynomiale pour tous les axes.

Exemple

Code de programme	**Commentaire**
`N10 G1 X… Y… Z… F600`	
`N11 POLY PO[X]=(1,2.5,0.7) PO[Y]=(0.3,1,3.2) PL=1.5`	`; Interpolation polynomiale activée`
`N12 PO[X]=(0,2.5,1.7) PO[Y]=(2.3,1.7) PL=3`	
`...`	
`N20 M8 H126 …`	
`N25 X70 PO[Y]=(9.3,1,7.67) PL=5`	`; Indications mixtes pour les axes`
`N27 PO[X]=(10,2.5) PO[Y]=(2.3)`	`; PL n'est pas programmé, donc PL=1.`
`N30 G1 X… Y… Z.`	`; Interpolation polynomiale désactivée`
…	

Exemple : Nouvelle syntaxe polynomiale

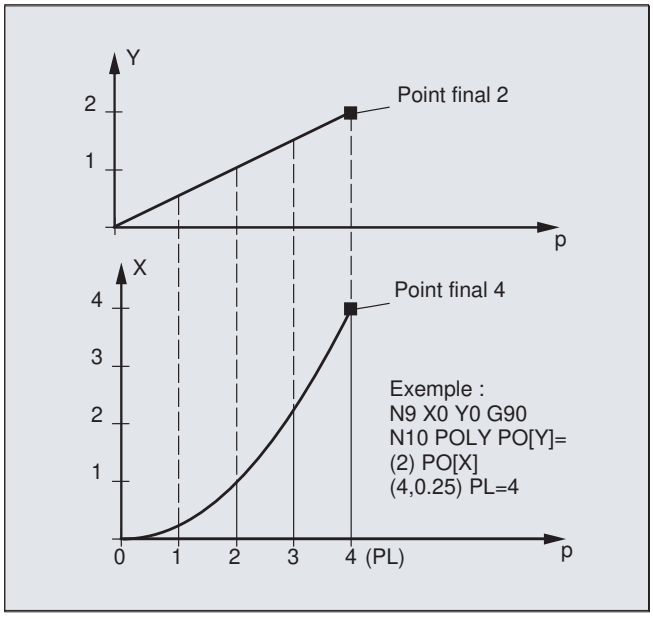
Syntaxe polynomiale encore valable	Nouvelle syntaxe polynomiale
`PO[descripteur d'axe]=(.. , ..)`	`Descripteur d'axe=PO(.. , ..)`
`PO[PHI]=(.. , ..)`	`PHI=PO(.. , ..)`
`PO[PSI]=(.. , ..)`	`PSI=PO(.. , ..)`
`PO[THT]=(.. , ..)`	`THT=PO(.. , ..)`
`PO[]=(.. , ..)`	`PO(.. , ..)`
`PO[variable]=IC(.. , ..)`	`Variable=PO IC(.. , ..)`

Exemple : Cercle dans le plan X/Y

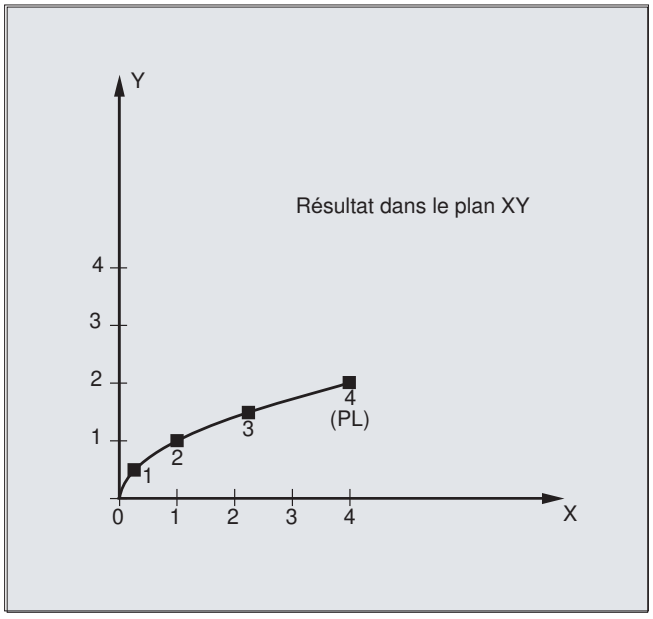
Programmation

Code de programme
`N9 X0 Y0 G90 F100`
`N10 POLY PO[Y]=(2) PO[X]=(4,0.25) PL=4`

Evolution des courbes X(p) et Y(p)



Evolution de la courbe dans le plan XY



Description

La forme générale de la fonction polynomiale est la suivante :

$f(p)= a_0 + a_1p + a_2p^2 + . . . + a_np^n$

où : a_i : coefficients constants (i = 0, 1, ..., n)
p : paramètres

Le nombre maximum de polynômes du 5e degré pouvant être programmés dans la commande est le suivant :

$$f(p)= a_0 + a_1p + a_2p^2 + a_3p^3 + a_4p^4 + a_5p^5$$

En affectant des valeurs concrètes aux coefficients, il est possible de générer différentes courbes : droites, paraboles et fonctions puissance.

Une courbe droite est créée par $a_2 = a_3 = a_4 = a_5 = 0$:

$$f(p)= a_0 + a_1p$$

En outre :

a_0 : position de l'axe à la fin du bloc précédent
$p = PL$
$a_1 = (x_E - a_0 - a_2*p^2 - a_3*p^3) / p$

Des polynômes peuvent être programmés
sans avoir que l'interpolation polynomiale n'ait été activée au moyen de l'instruction G `POLY`. Dans ce cas, les polynômes programmés ne seront pas interpolés, mais les points finaux programmés pour les axes seront accostés linéairement (`G1`). Les polynômes programmés ne sont déplacés en tant que tels qu'après avoir explicitement activé l'interpolation polynomiale dans le programme pièce (`POLY`).

Particularité : polynôme dénominateur

Pour les axes géométriques, `PO[]=(...)` permet aussi de programmer un polynôme dénominateur commun sans indiquer le nom d'un axe, c'est-à-dire que le déplacement des axes géométriques sera interpolé en tant que quotient de deux polynômes.

Ceci permet, par exemple, de représenter avec exactitude des coniques (cercle, ellipse, parabole, hyperbole).

Exemple :

Code de programme	**Commentaire**
`POLY G90 X10 Y0 F100`	`; Déplacement linéaire des axes géométriques à la position X10 Y0.`
`PO[X]=(0,-10) PO[Y]=(10) PO[]=(2,1)`	`; Déplacement des axes géométriques en quart de cercle sur position X0 Y10.`

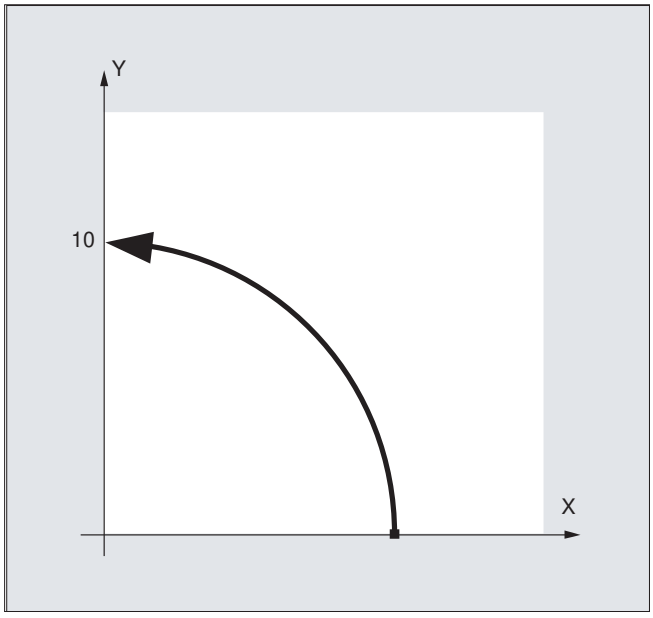
Le coefficient constant (a_0) du polynôme dénominateur est toujours considéré comme étant égal à 1. Le point final programmé est indépendant de `G90` / `G91`.

X(p) et Y(p) sont calculés comme suit à partir des valeurs programmées :

$X(p) = (10 - 10 * p^2) / (1 + p^2)$
$Y(p) = 20 * p / (1 + p^2)$
avec $0 \le p \le 1$

Compte tenu de la programmation des points de départ, des points finaux, du coefficient a_2 et de PL=1, les résultats intermédiaires suivants sont obtenus :

Numérateur (X) = $10 + 0 * p - 10 * p^2$
Numérateur (Y) = $0 + 20 * p + 0 * p^2$
Dénominateur = $1 + p^2$



Lorsque l'interpolation polynomiale est activée, la programmation d'un polynôme dénominateur possédant des racines à l'intérieur de l'intervalle `[0,PL]` est rejetée avec une alarme. Le polynôme dénominateur n'a aucune influence sur les déplacements des axes supplémentaires.

Remarque

Dans le cas de l'interpolation polynomiale, on peut activer une correction de rayon d'outil avec `G41`, `G42` et l'utiliser comme pour l'interpolation linéaire ou circulaire.

3.7.6 Référence de trajectoire réglable (SPATH, UPATH)

Pour l'interpolation polynomiale (`POLY`, `ASPLINE`, `BSPLINE`, `CSPLINE`, `COMP...`), les positions des axes à interpolation i sont spécifiées par des polynômes $p_i(U)$. Dans un bloc CN, le paramètre de courbe U varie de 0 à 1.

`FGROUP` permet de sélectionner les axes (axes FGROUP) auxquels se rapporte l'avance tangentielle `F`. Toutefois, une interpolation à vitesse constante sur la trajectoire S des axes FGROUP signifie généralement une variation non constante du paramètre de courbe U pendant

l'interpolation polynomiale. Pour les axes non contenus dans FGROUP, il est ainsi possible de choisir entre deux possibilités de poursuite en fonction des axes FGROUP :

- de manière synchrone avec la trajectoire S (SPATH)
- de manière synchrone avec le paramètre de courbe U (UPATH)

Syntaxe

```
SPATH
UPATH
```

Signification

SPATH :	Les axes non contenus dans FGROUP sont déplacés en fonction de la trajectoire S
UPATH :	Les axes non contenus dans FGROUP sont déplacés en fonction du paramètre de courbe U

Remarque

UPATH et SPATH déterminent également la relation entre le polynôme décrivant le mot F (FPOLY, FCUB, FLIN) et le déplacement le long de la trajectoire.

Autres conditions à prendre en compte

SPATH ou UPATH n'ont aucune signification pour :

- interpolation linéaire (G1)
- interpolation circulaire (G2, G3)
- blocs de filetage (G33, G34, G35, G33x, G63)
- Tous les axes en interpolation sont contenus dans FGROUP

Exemple

L'exemple suivant illustre la différence entre les deux modes de pilotage du déplacement.

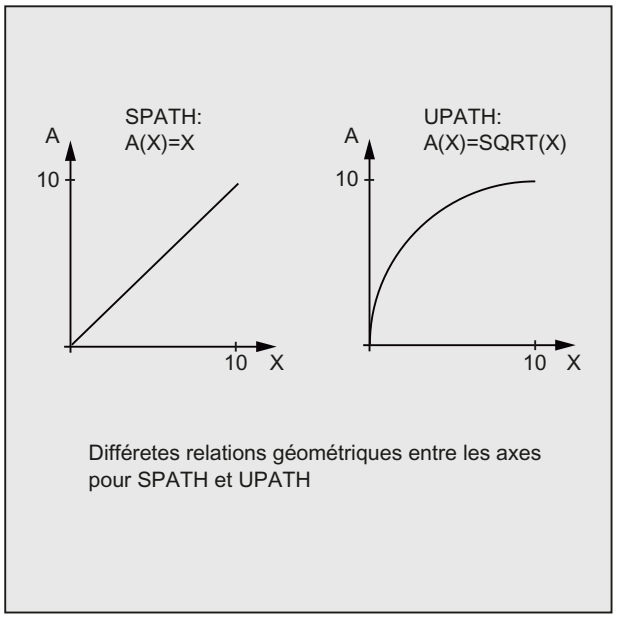
Code de programme

```
N10 FGROUP(X,Y,Z)
N15 G1 X0 A0 F1000 SPATH                ; SPATH
N20 POLY PO[X]=(10,10) A10
```

Code de programme

```
N10 FGROUP(X,Y,Z)
N15 G1 X0 A0 F1000 UPATH                ; UPATH
N20 POLY PO[X]=(10,10) A10
```

Dans les deux sections de programme, la trajectoire S dans N20 des axes FGROUP dépend du carré du paramètre de courbe U. C'est pourquoi on obtient des positions différentes de l'axe synchrone A en fonction du déplacement de X, selon que SPATH ou UPATH est actif :



Différetes relations géométriques entre les axes pour SPATH et UPATH

Informations complémentaires

Comportement de la commande en cas de Reset et paramètres machine/d'option

Après un Reset, l'instruction G définie par PM20150 $MC_GCODE_RESET_VALUES[44] est active (45e groupe G).

La valeur de position initiale pour le type d'arrondissement est définie par PM20150 $MC_GCODE_RESET_VALUES[9] (10e groupe G).

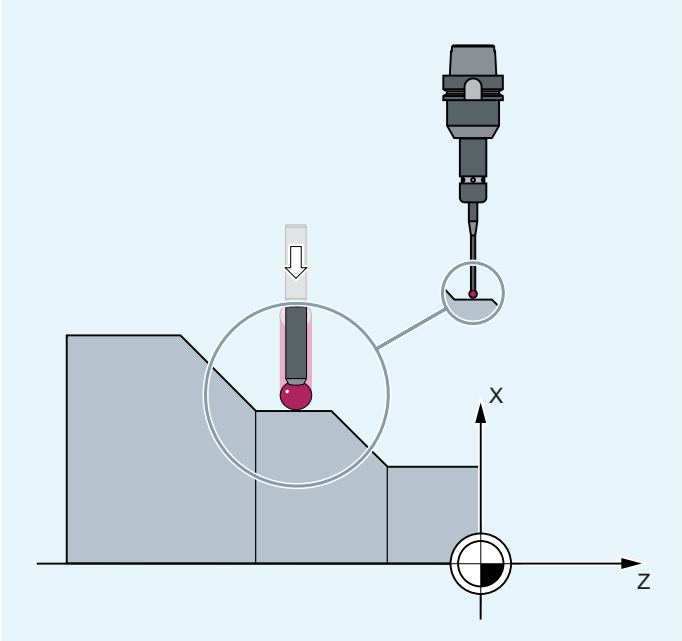
Le paramètre machine spécifique à l'axe PM33100 $MA_COMPRESS_POS_TOL[<n>] a une signification étendue : il contient les tolérances pour la fonction compresseur et l'arrondissement avec G642.

3.7.7 Mesure spécifique au canal (MEAS, MEAW)

Lors de la mesure spécifique au canal, l'activation du processus de mesure pour un canal CN s'effectue toujours à partir du programme pièce en cours dans le canal concerné. Dans un bloc de mesure, **un** événement de déclenchement (front montant ou descendant du palpeur) et **un** mode de mesure avec effacement de la distance restant à parcourir (MEAS) ou sans effacement de la distance restant à parcourir (MEAW) sont programmés. **Tous** les axes programmés dans le bloc de mesure participent alors au processus de mesure.

Dès qu'un bloc de mesure est actif, le palpeur accoste la pièce. Au front de déclenchement du palpeur, les positions sont mesurées pour tous les axes programmés dans le bloc de mesure et écrites pour chaque axe dans la cellule mémoire correspondante.

Les résultats de mesure peuvent être lus dans le programme pièce ou à l'aide d'actions synchrones soit dans le système de coordonnées machine, soit dans le système de coordonnées pièce.



Syntaxe

```
MEAS=<TE> G... X... Y... Z...
MEAW=<TE> G... X... Y... Z...
```

Signification

MEAS :	Mesure **avec** effacement de la distance restant à parcourir		
	Prise d'effet :		Bloc par bloc (non modale)
MEAW :	Mesure **sans** effacement de la distance restant à parcourir		
	Prise d'effet :		Bloc par bloc (non modale)
	Application :		Pour les tâches de mesure où la position programmée doit impérativement être accostée.
<TE>:	Evénement déclencheur de la mesure		
	Type :	INT	
	Plage de valeurs :		-2, -1, 1, 2
	Valeur :	(+)1	Front montant du palpeur 1 (à l'entrée de mesure 1)
		-1	Front descendant du palpeur 1 (à l'entrée de mesure 1)
		(+)2	Front montant du palpeur 2 (à l'entrée de mesure 2)
		-2	Front descendant du palpeur 2 (à l'entrée de mesure 2)
	Remarque : Il existe au maximum 2 palpeurs (selon le niveau d'extension).		

`G... :`	Mode d'interpolation, p. ex. `G0`, `G1`, `G2` ou `G3`
`X... Y... Z... :`	Points finaux en coordonnées cartésiennes

Remarque

MEAS et MEAW sont à effet non modal et peuvent être programmés avec des instructions de déplacement. L'avance et le type d'interpolation (G0, G1...), de même que le nombre d'axes, doivent être adaptés au problème de mesure concerné.

Exemple

Code de programme	**Commentaire**
`...`	
`N10 MEAS=1 G1 F1000 X100 Y730 Z40`	`; Bloc de mesure avec palpeur de la première entrée de mesure et interpolation linéaire. Un arrêt du prétraitement est automatiquement créé.`
`...`	

Informations complémentaires

Interrogation de l'état

S'il est nécessaire, dans le programme, de savoir si un palpeur a été actionné ou s'est déclenché, l'état peut être interrogé dans les variables système suivantes :

Variable système	Signification	Type de données	Valeur	
$A_PROBE[<n>]	État du palpeur	INT	0	Palpeur non actionné.
			1	Palpeur actionné.
$AC_MEA[<n>]	État de déclenchement du palpeur $AC_MEA[<n>] est automatiquement remis à zéro au début d'une mesure.	INT	0	Le palpeur ne s'est pas déclenché.
			1	Le palpeur s'est déclenché.

<n> = numéro du palpeur

Enregistrement des valeurs de mesure

Lors de la mesure spécifique au canal, les positions sont mesurées pour tous les axes à interpolation et axes de positionnement d'un bloc, qui sont déplacés (le nombre maximal d'axes

dépend de la configuration de la commande). Avec MEAS, le déclenchement du palpeur entraîne un freinage défini du déplacement.

Remarque

Si un axe géométrique est programmé dans un bloc de mesure, les valeurs de mesure sont enregistrées pour tous les axes géométriques courants.

Si un axe concerné par une transformation est programmé dans un bloc de mesure, les valeurs de mesure sont enregistrées pour tous les axes concernés par cette transformation.

Lecture des résultats de mesure

Les valeurs de mesure du palpeur des axes enregistrés peuvent être lues à l'aide des variables système suivantes dans le programme pièce et dans les actions synchrones :

Variable système	Signification
$AA_MM[<Axis>]	Valeur de mesure du palpeur dans le système de coordonnées machine.
$AA_MW[<Axis>]	Valeur de mesure du palpeur dans le système de coordonnées pièce.

3.7.8 Mesure spécifique à l'axe (MEASA, MEAWA, MEAC) (option)

Pour la mesure spécifique à l'axe, l'activation peut avoir lieu dans le programme pièce **ou** dans les actions synchrones. Si deux systèmes de mesure sont disponibles pour les axes, ils peuvent tous deux être utilisés pour la mesure.

Les méthodes de mesure suivantes sont disponibles :

- Mesure avec effacement de la distance restant à parcourir (MEASA)
- Mesure sans effacement de la distance restant à parcourir (MEAWA)
- Mesure continue sans effacement de la distance restant à parcourir (MEAC)

MEASA et MEAWA permettent de saisir à chaque mesure jusqu'à quatre valeurs de mesure pour l'axe programmé et de les stocker dans des variables système en fonction de l'événement déclencheur.

Pour la mesure continue avec MEAC, les résultats de mesure sont stockés dans des variables FIFO.

Syntaxe

```
MEASA[<Axis>]=(<Mode>,<TE1>,...,<TE4>)
MEAWA[<Axis>]=(<Mode>,<TE1>,...,<TE4>)
MEAC[<Axis>]=(<Mode>,<MeasMem>,<TE1>,...,<TE4>)
```

Remarque

MEASA et MEAWA sont à effet non modal et peuvent être programmés ensemble dans un bloc. Par contre, si MEASA/MEAWA sont programmés avec MEAS/MEAW dans un même bloc, un message d'erreur est émis.

Signification

MEASA :	Mesure spécifique à l'axe avec effacement de la distance restant à parcourir	
	Prise d'effet :	bloc par bloc (non modale)
MEAWA :	Mesure spécifique à l'axe sans effacement de la distance restant à parcourir	
	Prise d'effet :	bloc par bloc (non modale)
MEAC :	Mesure continue spécifique à l'axe sans effacement de la distance restant à parcourir	
	Prise d'effet :	Bloc par bloc (non modale)
<Axis> :	Nom de l'axe de canal utilisé pour la mesure	
<Mode> :	Nombre à deux chiffres (xx) pour l'indication du mode de fonctionnement (mode de mesure et système de mesure)	
	Unité : Mode de mesure Définit si les événements déclencheurs doivent être activés dans l'ordre chronologique ou de programmation.	
	x**0**	Annulation de la tâche de mesure.
	x**1**	Jusqu'à 4 événements déclencheurs **différents**, activables simultanément.
	x**2**	Jusqu'à 4 événements déclencheurs différents, activables successivement.
	x**3**	Jusqu'à 4 événements déclencheurs activables successivement, cependant sans surveillance de l'événement déclencheur 1 au démarrage (les alarmes 21700/21703 sont inhibées). **Remarque :** Ce mode n'est pas possible avec MEAC.
	Dizaine : Système de mesure Définit le système de mesure avec lequel la mesure doit être effectuée.	
	0x (ou aucune indication)	Système de mesure actif
	1x	Système de mesure 1
	2x	Système de mesure 2
	3x	Les deux systèmes de mesure
<TE> :	Evénement déclencheur de la mesure	
	Type :	INT
	Plage de valeurs :	-2, -1, 1, 2
	(+)1	Front montant du palpeur 1
	-1	Front descendant du palpeur 1
	(+)2	Front montant du palpeur 2
	-2	Front descendant du palpeur 2
	Remarque : Si la mesure est effectuée avec deux systèmes de mesure, maximum deux événements déclencheurs sont programmables (front montant ou descendant). Pour chacun des deux événements déclencheurs, les valeurs de mesure des deux systèmes de mesure sont saisies.	
<MeasMem> :	Numéro de la FIFO (mémoire à file d'attente)	

Exemples

Exemple 1 : Mesure spécifique à l'axe avec effacement de la distance restant à parcourir en mode 1 (évaluation dans l'ordre chronologique)

a) Mesure avec un système de mesure

Code de programme	Commentaire
...	
N100 **MEASA[X]=(1,1,-1)** G01 X100 F100	; Mesure en mode 1 avec système de mesure actif. Attente du signal de mesure avec front montant/descendant du palpeur 1 sur le trajet vers X=100.
N110 IF $AC_MEA[1]==FALSE GOTOF ENDE	; Vérification du succès de la mesure.
N120 R10=$AA_MM1[X]	; Mémorisation de la valeur de mesure correspondant au premier événement déclencheur programmé (front montant).
N130 R11=$AA_MM2[X]	; Mémorisation de la valeur de mesure correspondant au deuxième événement déclencheur programmé (front descendant).
N140 FIN :	

b) Mesure avec deux systèmes de mesure

Code de programme	Commentaire
...	
N200 **MEASA[X]=(31,1,-1)** G01 X100 F100	; Mesure en mode 1 avec les deux systèmes de mesure. Attente du signal de mesure avec front montant/descendant du palpeur 1 sur le trajet vers X=100.
N210 IF $AC_MEA[1]==FALSE GOTOF ENDE	; Vérification du succès de la mesure.
N220 R10=$AA_MM1[X]	; Mémorisation de la valeur de mesure du système de mesure 1 sur front montant.
N230 R11=$AA_MM2[X]	; Mémorisation de la valeur de mesure du système de mesure 2 sur front montant.
N240 R12=$AA_MM3[X]	; Mémorisation de la valeur de mesure du système de mesure 1 sur front descendant.
N250 R13=$AA_MM4[X]	; Mémorisation de la valeur de mesure du système de mesure 2 sur front descendant.
N260 FIN :	

Exemple 2 : Mesure spécifique à l'axe avec effacement de la distance restant à parcourir en mode 2 (évaluation dans l'ordre de programmation)

Code de programme	Commentaire
...	

Code de programme	Commentaire
N100 **MEASA[X]=(2,1,-1,2,-2)** G01 X100 F100	; Mesure en mode 2 avec système de mesure actif. Attente du signal de mesure dans l'ordre suivant : front montant du palpeur 1, front descendant du palpeur 1, front montant du palpeur 2, front descendant du palpeur 2 sur le trajet vers X=100.
N110 IF $AC_MEA[1]==FALSE GOTOF MESSTASTER2	; Vérification du succès de la mesure avec palpeur 1.
N120 R10=$AA_MM1[X]	; Mémorisation de la valeur de mesure correspondant au premier événement déclencheur programmé (front montant palpeur de mesure 1).
N130 R11=$AA_MM2[X]	; Mémorisation de la valeur de mesure correspondant au deuxième événement déclencheur programmé (front montant palpeur de mesure 1).
N140 DETECTEUR2 :	
N150 IF $AC_MEA[2]==FALSE GOTOF ENDE	; Vérification du succès de la mesure avec palpeur 2.
N160 R12=$AA_MM3[X]	; Mémorisation de la valeur de mesure correspondant au troisième événement déclencheur programmé (front montant palpeur de mesure 2).
N170 R13=$AA_MM4[X]	; Mémorisation de la valeur de mesure correspondant au quatrième événement déclencheur programmé (front montant palpeur de mesure 2).
N180 FIN :	

Exemple 3 : Mesure continue spécifique à l'axe en mode 1 (évaluation dans l'ordre chronologique)

a) Mesure de 100 valeurs ou moins

Code de programme	Commentaire
...	
N110 DEF REAL VAL_MESURE[100]	
N120 DEF INT boucle=0	
N130 **MEAC[X]=(1,1,-1)** G01 X1000 F100	; Mesure en mode 1 avec système de mesure actif, mémorisation des valeurs de mesure dans $AC_FIFO1, attente du signal de mesure avec front descendant du palpeur 1 sur le trajet vers X=1000.
N135 STOPRE	
N140 MEAC[X]=(0)	; Abandon de la mesure dès l'accostage de la position d'axe.
N150 R1=$AC_FIFO1[4]	; Mémorisation du nombre de valeurs de mesure accumulées dans le paramètre R1.
N160 FOR boucle=0 TO R1-1	

Code de programme	**Commentaire**
N170 VAL_MESURE[boucle] = $AC_FIFO1[0]	; Lecture et mémorisation des valeurs de mesure à partir de $AC_FIFO1.
N180 ENDFOR	

b) Mesure avec effacement de la distance restant à parcourir après 10 valeurs de mesure

Code de programme	**Commentaire**
...	
N10 WHEN $AC_FIFO1[4]>=10 DO MEAC[x]=(0) DELDTG(x)	; Effacement de la distance restant à parcourir.
N20 **MEAC[x]=(1,1,1,-1)** G01 X100 F500	
N30 MEAC [X]=(0)	
N40 R1=$AC_FIFO1[4]	; Nombre de valeurs de mesure.
...	

c) Mesure d'un flanc de dent montant/descendant à l'aide de 2 palpeurs de mesure

Code de programme	**Commentaire**
...	
N110 DEF REAL VAL_MESURE[16]	
N120 DEF INT boucle=0	
N130 **MEAC[X]=(1,1,-1,2)** G01 X100 F100	; Mesure en mode 1 avec système de mesure actif, mémorisation des valeurs de mesure dans $AC_FIFO1, attente du signal de mesure dans l'ordre front descendant du palpeur 1, front montant du palpeur 2, sur le trajet vers X=100.
N140 STOPRE	; Arrêt du prétraitement des blocs.
N150 MEAC[X]=(0)	; Abandon de la mesure dès l'accostage de la position d'axe.
N160 R1=$AC_FIFO1[4]	; Mémorisation du nombre de valeurs de mesure accumulées dans le paramètre R1.
N170 FOR boucle=0 TO R1-1	
N180 VAL_MESURE[boucle]=$AC_FIFO1[0]	; Lecture et mémorisation des valeurs de mesure à partir de $AC_FIFO1.
N190 ENDFOR	

Informations complémentaires

Tâche de mesure

La programmation d'une tâche de mesure peut avoir lieu dans le programme pièce ou à partir d'une action synchrone. On ne peut activer qu'une seule tâche de mesure par axe à un moment donné.

Remarque

L'avance est à adapter au problème de mesure posé.

Pour MEASA et MEAWA, des résultats corrects sont garantis uniquement avec des avances pour lesquelles il ne survient pas plus d'un même événement déclencheur et pas plus de quatre événements déclencheurs différents par cycle du régulateur de position.

Pour la mesure continue avec MEAC, le rapport entre le cycle d'interpolation et le cycle du régulateur de position ne doit pas être supérieur à 1:8.

Événement déclencheur

Un événement déclencheur se compose du numéro du palpeur et du critère de déclenchement du signal de mesure (front montant ou descendant).

Pour chaque mesure, le programme peut traiter respectivement jusqu'à 4 événements déclencheurs des palpeurs, autrement dit jusqu'à deux palpeurs avec deux fronts de mesure chacun. L'ordre de traitement des événements déclencheurs et leur nombre maximum dépendent du mode sélectionné.

Lorsque la mesure est effectuée avec deux systèmes de mesure, maximum deux événements déclencheurs sont programmables (front montant ou descendant). Pour chacun des deux événements déclencheurs, les valeurs de mesure des deux palpeurs sont saisies.

Lors de l'utilisation du télégramme PROFIBUS 391 (réglage par défaut pour la communication PROFIBUS), seule une valeur de mesure par événement déclencheur et cycle du régulateur de position est possible.

En cas d'utilisation du télégramme PROFIBUS 395, MEAC permet, en mode de fonctionnement 1, d'augmenter le nombre de valeurs de mesure à 8 valeurs de mesure au total par événement déclencheur pour un front montant et à 8 valeurs de mesure pour un front descendant par événement déclencheur et cycle du régulateur de position :

- Un palpeur : 8 valeurs de mesure pour front montant et 8 pour front descendant
- Deux palpeurs : 4 valeurs de mesure pour front montant et 4 pour front descendant par palpeur

Le télégramme PROFIBUS 395 permet ainsi de réaliser des avances ou des vitesses plus élevées.

Mode de fonctionnement

Le premier chiffre (dizaines) du mode de fonctionnement permet de sélectionner le système de mesure souhaité. S'il n'existe qu'un seul système de mesure, alors que le second a été programmé, c'est automatiquement le système de mesure existant qui sera pris en compte.

Le second chiffre (unités) permet de sélectionner le mode de mesure souhaité. La procédé de mesure s'adapte ainsi aux possibilités de la commande respective :

- **Mode 1**
  Les événements déclencheurs sont traités dans l'ordre chronologique de leur apparition.
- **Mode 2**
  Les événements déclencheurs sont traités dans l'ordre de leur programmation.
- **Mode 3**
  Les événements déclencheurs sont traités dans l'ordre de leur programmation, cependant aucune surveillance de l'événement déclencheur 1 au démarrage n'est réalisée.

Mesure avec et sans effacement de la distance restant à parcourir

Lors de la programmation de MEASA, l'effacement de la distance restant à parcourir est effectué seulement après la saisie de toutes les valeurs de mesure requises.

Pour des tâches de mesure spéciales où la position programmée doit impérativement être accostée, la fonction MEAWA est utilisée.

Remarque

MEASA n'est pas programmable dans des actions synchrones. Il est également possible de programmer MEAWA avec effacement de la distance restant à parcourir en tant qu'action synchrone.

Quand la tâche de mesure est lancée avec MEAWA à partir d'actions synchrones, les valeurs de mesure sont disponibles uniquement dans le système de coordonnées machine.

Axes géométriques / transformations

Si la mesure spécifique à l'axe doit être lancée pour un axe géométrique, il convient de programmer de façon explicite la même tâche de mesure pour tous les autres axes géométriques. Il en va de même pour les axes qui participent à une transformation.

Exemples :

```
N10 MEASA[Z]=(1,1) MEASA[Y]=(1,1) MEASA[X]=(1,1) G0 Z100
```

ou

```
N10 MEASA[Z]=(1,1) POS[Z]=100
```

Interrogation de l'état

S'il est nécessaire, dans le programme, de savoir si un palpeur a été actionné ou s'est déclenché, l'état peut être interrogé dans les variables système suivantes :

Variable système	Signification	Type de données	Valeur	
$A_PROBE[<n>]	État du palpeur	INT	0	Palpeur non actionné.
			1	Palpeur actionné.

Variable système	Signification	Type de données	Valeur	
$AC_MEA[<n>]	État de déclenchement du palpeur $AC_MEA[<n>] est automatiquement remis à zéro au début d'une mesure.	INT	0	Le palpeur ne s'est pas déclenché.
			1	Le palpeur s'est déclenché (tous les événements déclencheurs programmés dans le bloc ont eu lieu).

<n> = numéro du palpeur

Remarque

Si la mesure est lancée à partir d'actions synchrones, $AC_MEA n'est plus actualisé. Dans ce cas, il convient d'interroger le signal d'interface CN/AP DB31, ... DBX62.3 ou la variable équivalente $AA_MEAACT[<Axis>] :

$AA_MEAACT==1 : mesure activée

$AA_MEAACT==0 : mesure non activée

Limitation du palpeur

Avec les variables système $A_PROBE_LIMITED, il est possible de lire l'état de la limitation du palpeur dans le programme CN ou l'action synchrone lors de l'utilisation du télégramme PROFIBUS 395 :

$A_PROBE_LIMITED[<n>] == 0: Limitation du palpeur inactive/remise à zéro

$A_PROBE_LIMITED[<n>] == 1: Limitation du palpeur active

<n> = numéro de palpeur

Résultats de mesure pour MEASA / MEAWA

Les valeurs de mesure du palpeur peuvent être lues à l'aide des variables système suivantes dans le programme pièce et dans les actions synchrones :

Variable système	Signification
$AA_MM1[<Axis>]	Valeur de mesure du palpeur pour l'événement déclencheur 1 dans le SCM
...	...
$AA_MM4[<Axis>]	Valeur de mesure du palpeur pour l'événement déclencheur 4 dans le SCM
$AA_MW1[<Axis>]	Valeur de mesure du palpeur pour l'événement déclencheur 1 dans le SCP
...	...
$AA_MW4[<Axis>]	Valeur de mesure du palpeur pour l'événement déclencheur 4 dans le SCP

<Axis> = axe de mesure

Lorsque la tâche de mesure est exécutée avec deux systèmes de mesure, chacun des deux événements déclencheurs possibles est saisi par les deux systèmes de mesure. L'affectation des variables système s'effectue alors comme suit :

$AA_MM1[<Axis>]	ou	$AA_MW1[<Axis>]	Valeur de mesure du système de mesure 1 pour l'événement déclencheur 1
$AA_MM2[<Axis>]	ou	$AA_MW2[<Axis>]	Valeur de mesure du système de mesure 2 pour l'événement déclencheur 1
$AA_MM3[<Axis>]	ou	$AA_MW3[<Axis>]	Valeur de mesure du système de mesure 1 pour l'événement déclencheur 2
$AA_MM4[<Axis>]	ou	$AA_MW4[<Axis>]	Valeur de mesure du système de mesure 2 pour l'événement déclencheur 2

Mesure continue (MEAC)

Avec MEAC, les valeurs de mesure sont fournies dans le système de coordonnées machine et stockées dans la mémoire FIFO[<n>] indiquée (mémoire à file d'attente). Quand deux palpeurs ont été configurés pour la mesure, les valeurs de mesure du second palpeur sont stockées séparément dans la mémoire FIFO[<n>+1] configurée à cet effet (par le biais des PM).

La mémoire FIFO est une mémoire à file d'attente dans laquelle on enregistre les valeurs de mesure dans des variables $AC_FIFO suivant le principe "premier entré, premier sorti".

Remarque

Le contenu de la mémoire FIFO ne peut être lu qu'une seule fois. Pour pouvoir réutiliser les valeurs mesurées, il convient de les mémoriser temporairement dans les données utilisateur.

Dès que le nombre de valeurs de mesure destinées à la mémoire FIFO dépasse le nombre maximum prévu dans les paramètres machine, il est mis fin automatiquement à la mesure.

Pour réaliser une mesure sans fin, programmer la lecture cyclique des valeurs de mesure. La lecture doit être effectuée au moins à la même fréquence que l'entrée de nouvelles valeurs de mesure.

Informations complémentaires : Description fonctionnelle Actions synchrones

Protection contre les erreurs de programmation

Les erreurs de programmation suivantes sont détectées et signalées par un message d'erreur :

- MEASA/MEAWA programmés avec MEAS/MEAW dans un même bloc
  Exemple :
  ```
  N01 MEAS=1 MEASA[X]=(1,1) G01 F100 POS[X]=100
  ```
- MEASA/MEAWA avec un nombre de paramètres <2 ou >5
  Exemple :
  ```
  N01 MEAWA[X]=(1) G01 F100 POS[X]=100
  ```

- MEASA/MEAWA avec événement déclencheur différent de 1/ -1/ 2/ -2
  Exemple :
  `N01 MEASA[B]=(1,1,3) B100`
- MEASA/MEAWA avec mode incorrect
  Exemple :
  `N01 MEAWA[B]=(4,1) B100`
- MEASA/MEAWA avec événement déclencheur programmé deux fois
  Exemple :
  `N01 MEASA[B]=(1,1,-1,2,-1) B100`
- MEASA/MEAWA et axe géométrique manquant
  Exemple :
  `N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) G01 X50 Y50 Z50 F100 ;axe GEOX/Y/Z`
- Tâche de mesure hétérogène pour axes géométriques
  Exemple :
  `N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) MEASA[Z]=(1,1,2) G01 X50 Y50 Z50 F100`

Voir aussi

Actions synchrones (Page 983)

3.7.9 Fonctions spéciales pour l'utilisateur OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829)

Adresses OEM

L'utilisateur OEM définit la signification des adresses OEM. La fonctionnalité est introduite par le biais de cycles de compilation. Cinq adresses OEM sont réservées (`OMA1` ... `OMA5`). Les descripteurs d'adresse peuvent être spécifiés. Les adresses OEM sont admises dans chaque bloc.

Appels d'instruction G réservés

Les appels d'instruction G suivantes sont réservés pour l'utilisateur OEM :

- `OEMIPO1`, `OEMIPO2` (à partir du groupe G 1)
- `G810` ... `G819` (groupe G 31)
- `G820` ... `G829` (groupe G 32)

La fonctionnalité est introduite par le biais de cycles de compilation.

Fonctions et sous-programmes

En outre, l'utilisateur OEM peut créer des fonctions et des sous-programmes prédéfinis avec transfert de paramètres.

Remarque

Simulation de pièce

Jusqu'à la version 4.4. du logiciel aucun cycle de compilation (CC) n'est pris en charge par la simulation de pièce. A partir de la version 4.4, seule une sélection de cycles de compilation est prise en charge.

Les instructions de langage du programme pièce provenant de cycles de compilation non pris en charge (`OMA1` ... `OMA5`, `OEMIPO1/2`, `G810` ... `G829`, procédures et fonctions utilisateur) entraînent donc, sans traitement individuel, la signalisation d'une alarme et l'interruption de la simulation.

Solution : Traiter individuellement les éléments de langage spécifiques CC manquants du programme pièce (requête $P_SIM).
Exemple :

```
N1 G01 X200 F500
IF (1==$P_SIM)
N5 X300 ; inactif lors de la simulation CC
ELSE
N5 X300 OMA1=10
ENDIF
```

3.7.10 Réduction de l'avance avec décélération aux angles (FENDNORM, G62, G621)

Lors de la décélération automatique aux angles, l'avance est réduite juste avant l'angle concerné selon une courbe en forme de cloche. En outre, l'étendue du comportement d'outil pertinent pour l'usinage peut être paramétrée au moyen des données de réglage :

- Début et fin de la réduction de l'avance
- Correction de vitesse avec laquelle l'avance est réduite
- Identification de l'angle pertinent

En tant qu'angles pertinents, on prend en considération les angles dont l'angle interne est inférieur à l'angle paramétré par l'intermédiaire de la donnée de réglage .

La valeur par défaut FENDNORM désactive la fonction de correction automatique aux angles.

Syntaxe

```
FENDNORM
G62 G41
G621
```

Signification

FENDNORM :	Décélération automatique aux angles désactivée
G62 :	Décélération **aux angles intérieurs** avec correction du rayon d'outil activée
G621 :	Décélération **à tous les angles** avec correction du rayon d'outil activée

G62 n'agit qu'aux angles rentrants programmés avec une correction de rayon d'outil G41/G42 active et un contournage actif G64/G641.

L'angle correspondant est accosté avec l'avance réduite, qui se calcule comme suit :

F * (correction de vitesse pour la réduction de l'avance) * correction de vitesse de l'avance

La réduction maximale de l'avance peut être atteinte lorsque l'outil doit assurer, par rapport à la trajectoire du centre de l'outil, le changement de direction vers l'angle concerné.

G621 intervient comme G62 au niveau de chaque angle formé par les axes définis avec FGROUP.

3.7.11 Critère programmable de fin de déplacement (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)

De façon analogue aux critères de changement de bloc en interpolation de trajectoire (G601, G602 et G603), vous pouvez programmer un critère de fin de déplacement en interpolation d'axes individuels dans un programme pièce ou une action synchrone pour axes de commande/AP.

Selon le réglage du critère de fin de déplacement, les blocs de programme pièce ou de cycle technologique comportant des déplacements d'axes individuels se terminent plus ou moins rapidement. Cela vaut également pour l'AP via FC15/16/18.

Syntaxe

```
FINEA[<axe>]
COARSEA[<axe>]
IPOENDA[<axe>]
IPOBRKA(<axe>[,<instant>])
ADISPOSA(<axe>[,<mode>,<taille de fenêtre>])
```

Signification

FINEA :	Critère de fin de déplacement : "Arrêt précis fin"	
	Prise d'effet :	modale
COARSEA :	Critère de fin de déplacement : "Arrêt précis grossier"	
	Prise d'effet :	modale
IPOENDA :	Critère de fin de déplacement : "Arrêt interpolateur"	
	Prise d'effet :	modale
IPOBRKA :	Critère de changement de bloc : Rampe de freinage	
	Prise d'effet :	modale

ADISPOSA :	Fenêtre de tolérance pour le critère de fin de déplacement		
	Prise d'effet :	modale	
<axe> :	Nom d'axe de canal (X, Y....)		
<instant> :	Instant du changement de bloc en % de la rampe de freinage : • 100 % = début de la rampe de freinage • 0 % = fin de la rampe de freinage, même signification que IPOENDA		
	Type :	REAL	
<mode> :	Référence de la fenêtre de tolérance		
	Plage de valeurs :	0	Fenêtre de tolérance pas active
		1	Fenêtre de tolérance rapportée à la consigne de position
		2	Fenêtre de tolérance rapportée à la position réelle
	Type :	INT	
<taille de fenêtre> :	Taille de la fenêtre de tolérance		
	Type :	REAL	

Exemples

Exemple 1 : critère de fin de déplacement : "Arrêt interpolateur"

Code de programme

```
; déplacement axe de positionnement X sur 100, vitesse
200 m/min, accélération 90 %,
; critère de fin de déplacement : Arrêt interpolateur
N110 G01 POS[X]=100 FA[X]=200 ACC[X]=90 IPOENDA[X]

; Action synchrone :
; TOUJOURS LORSQUE : l'entrée 1 est à 1
; PUIS déplacement axe de positionnement X sur 50, vi-
tesse 200 m/min, accélération 140 %,
;      critère de fin de déplacement : Arrêt interpola-
teur
N120 EVERY $A_IN[1] DO POS[X]=50 FA[X]=200 ACC[X]=140
IPOENDA[X]
```

Exemple 2 : Critère de changement de bloc : "Rampe de freinage"

Code de programme	Commentaire
	; Prise d'effet des paramètres par défaut
N40 POS[X]=100	; Mouvement de positionnement de X à la position 100
	Critère de changement de bloc : Arrêt précis fin
N20 IPOBRKA(X,100)	; Critère de changement de bloc : "Rampe de freinage",
	100 % = début de la rampe de freinage

Code de programme	Commentaire
N30 POS[X]=200	: Le changement de bloc a lieu dès que l'axe X commence à freiner
N40 POS[X]=250	; L'axe X ne continue pas à freiner à la position 200 mais continue à se déplacer jusqu'à la position 250.
	Dès que l'axe commence à freiner, le changement de bloc intervient.
N50 POS[X]=0	; L'axe X freine et retourne à la position 0.
	Le changement de bloc intervient à la position 0 avec "Arrêt précis fin"
N60 X10 F100	; L'axe X se déplace, en tant qu'axe à interpolation, vers la position 10

Informations complémentaires

Variable système pour le critère de fin de déplacement

Le critère de fin de déplacement actif peut être lu via la variable système $AA_MOTEND.

Critère de changement de bloc : "Rampe de freinage" (IPOBRKA)

Lors de l'activation du critère de changement de bloc "Rampe de freinage", si une valeur a été programmée pour l'instant de changement de bloc optionnel, cette valeur prend effet pour le mouvement de positionnement suivant et elle est écrite dans la donnée de réglage, de façon synchrone à l'exécution. Si aucune valeur n'a été indiquée pour l'instant de changement de bloc, la valeur actuelle de la donnée de réglage prend effet.

SD43600 $SA_IPOBRAKE_BLOCK_EXCHANGE

Lors de la programmation suivante d'un critère de fin de déplacement axial (FINEA, COARSEA, IPOENDA), IPOBRKA est désactivé pour l'axe correspondant.

Critère de changement de bloc supplémentaire : "Fenêtre de tolérance" (ADISPOSA)

ADISPOSA permet de définir, en tant que critère de changement de bloc supplémentaire, une fenêtre de tolérance autour du point de fin de bloc (au choix, position réelle ou position de consigne). Les deux conditions doivent alors être remplies pour le changement de bloc :

- Critère de changement de bloc : "Rampe de freinage"
- Critère de changement de bloc : "Fenêtre de tolérance"

3.8 Transformations des coordonnées (frames)

3.8.1 Transformation de coordonnées par variables frames

Outre les instructions de frame (Page 313), comme ROT, AROT, SCALE, etc., le système de coordonnées pièce (SCP) peut également être transformé via les variables frame $P_...FR (frames de gestion des données) et $P_...FRAME (frames actifs).

La figure suivante donne un aperçu de la structuration des variables frame :

- Frames de gestion des données
- Frames actifs
- Frame global actif : concaténation de tous les frames actifs
- Frames à définition globale pour NCU
- Frames spécifiques à un canal

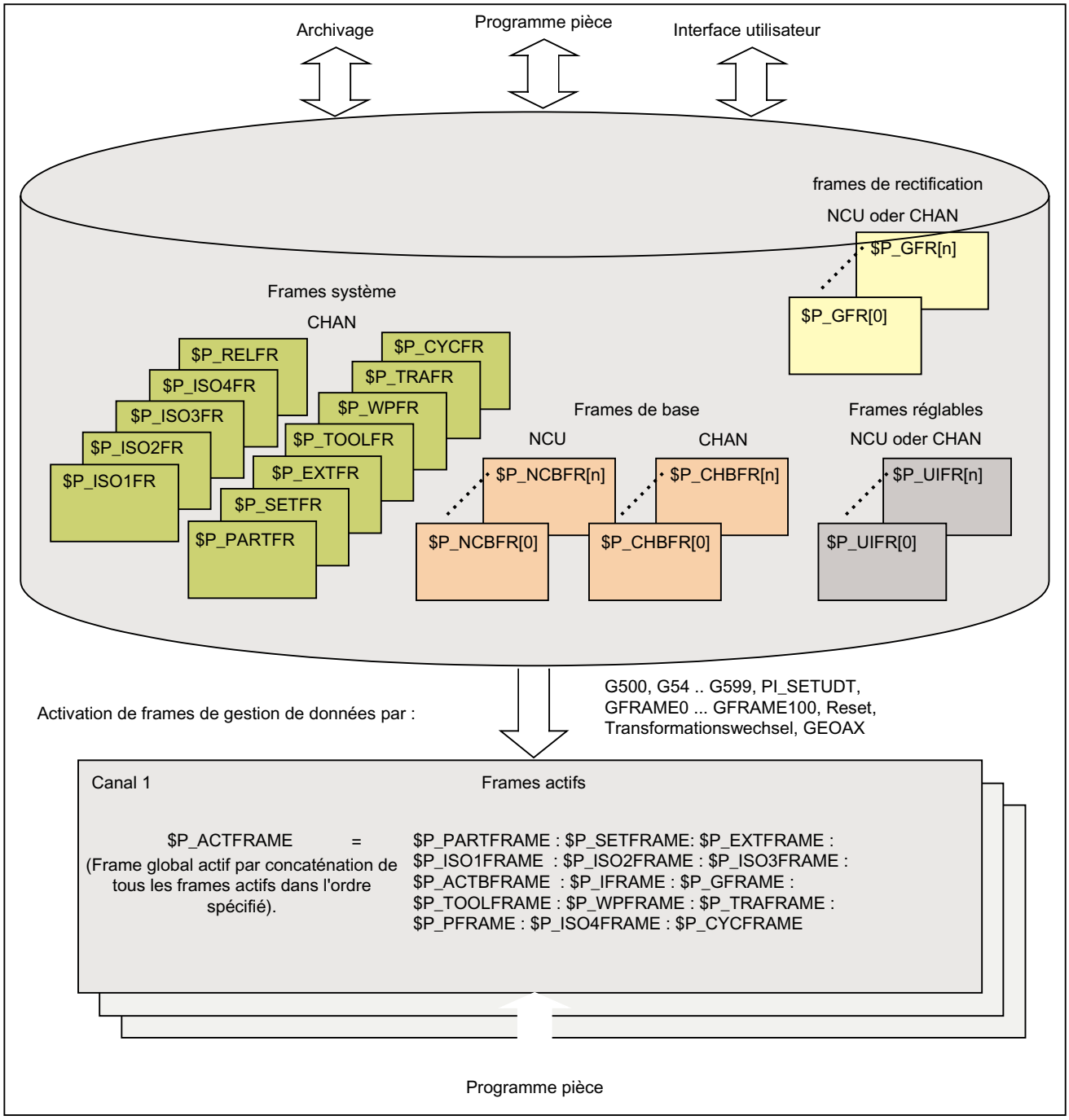


Figure 3-1 Vue d'ensemble des variables frame

3.8.1.1 Variable frame prédéfinie ($P_CHBFRAME, $P_IFRAME, $P_PFRAME, $P_ACTFRAME)

Coché : frames de base spécifiques à un canal $P_CHBFRAME[<n>] ($P_BFRAME)

Remarque

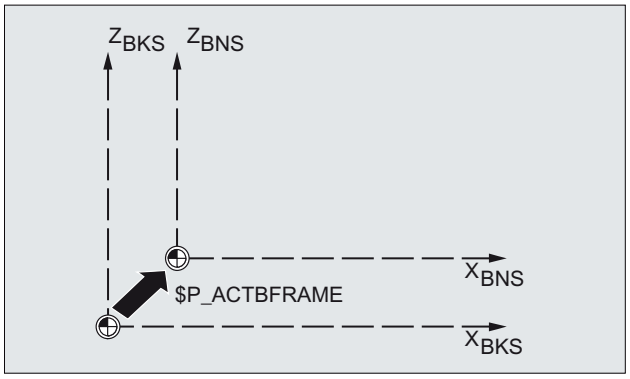
Le frame de base actuel `$P_BFRAME` et le frame de base de gestion des données `$P_UBFR` sont conservés pour des raisons de compatibilité.

- `$P_BFRAME` ≙ `$P_CHBFRAME[0]`
- `$P_UBFR` ≙ `$P_CHBFR[0]`.

Les variables frame $P_CHBFRAME[<n>] définissent la corrélation entre le système de coordonnées de base (SCB) et le système de coordonnées de base réglable (SBR).

Si le frame de base spécifique à un canal actuel $P_CHBFRAME[`<n>`] doit être immédiatement effectif dans le programme CN, les possibilités suivantes sont disponibles

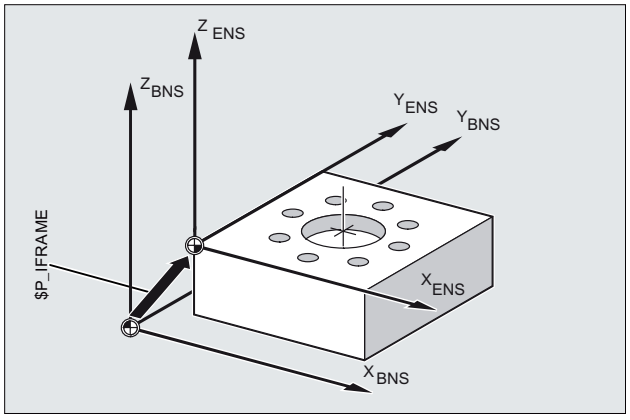
- Instructions :
  - `G500` (désactivation de tous les frames réglables, les frames de base restent actifs)
  - `G54` ... `G599` (décalages d'origine réglables)
- Affectation d'un frame de base spécifique à un canal de la gestion des données à un frame de base spécifique à un canal actuel :
  `$P_CHBFRAME[<n>] = $P_CHBFR[<m>]`



Coché : frame réglable spécifique à un canal $P_IFRAME

La variable frame $P_IFRAME définit la corrélation entre le système de coordonnées de base réglable (SBR) et le système de coordonnées pièce réglable (SPR).

- `$P_IFRAME` correspond à `$P_UIFR[$P_IFRNUM]`
- `$P_IFRAME` contient, après la programmation de `G54` par exemple, la translation, la rotation, l'homothétie et la fonction miroir définies par `G54`.

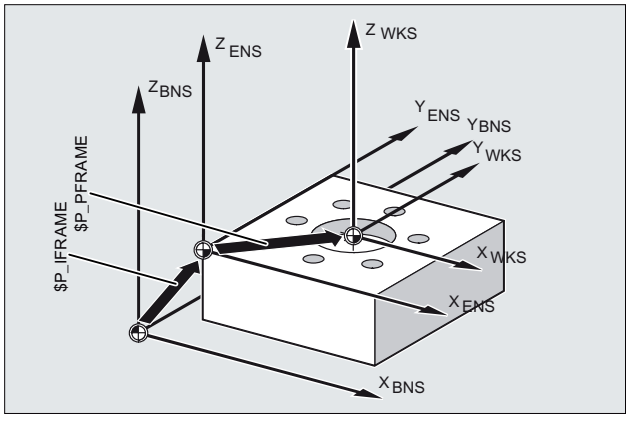


Coché : frame programmable spécifique à un canal $P_PFRAME

La variable frame $P_PFRAME définit la relation entre le système de coordonnés réglable (SCR) et le système de coordonnées-pièce (SCP).

`$P_PFRAME` contient le frame résultant, obtenu

- **à partir de la programmation** de `TRANS/ATRANS`, `ROT/AROT`, `SCALE/ASCALE`, `MIRROR/AMIRROR` ou
- **à partir de l'affectation** de `CTRANS`, `CROT`, `CMIRROR`, `CSCALE` au frame programmable.



Coché : frame global $P_ACTFRAME

Le frame global effectif dans un canal résulte de la concaténation de tous les frames actifs dans un canal.

```
$P_ACTFRAME =    $P_PARTFRAME : $P_SETFRAME  : $P_EXTFRAME :
                 $P_ISO1FRAME : $P_ISO2FRAME : $P_ISO3FRAME :
                 $P_ACTBFRAME : $P_IFRAME    : $P_GFRAME :
                 $P_TOOLFRAME : $P_WPFRAME   : $P_TRAFRAME :
                 $P_PFRAME    : $P_ISO4FRAME : $P_CYCFRAME
```

`$P_ACTFRAME` décrit l'origine du système de coordonnées pièce courant.

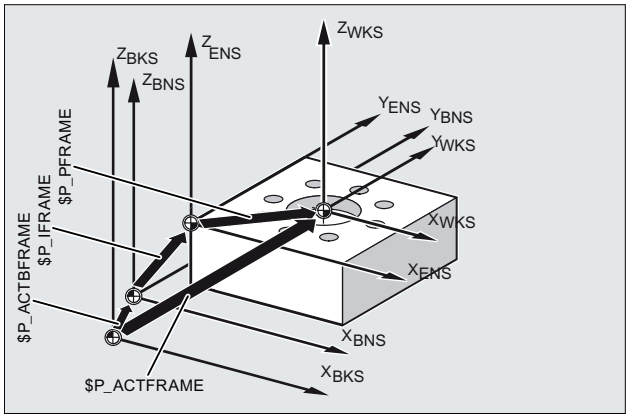
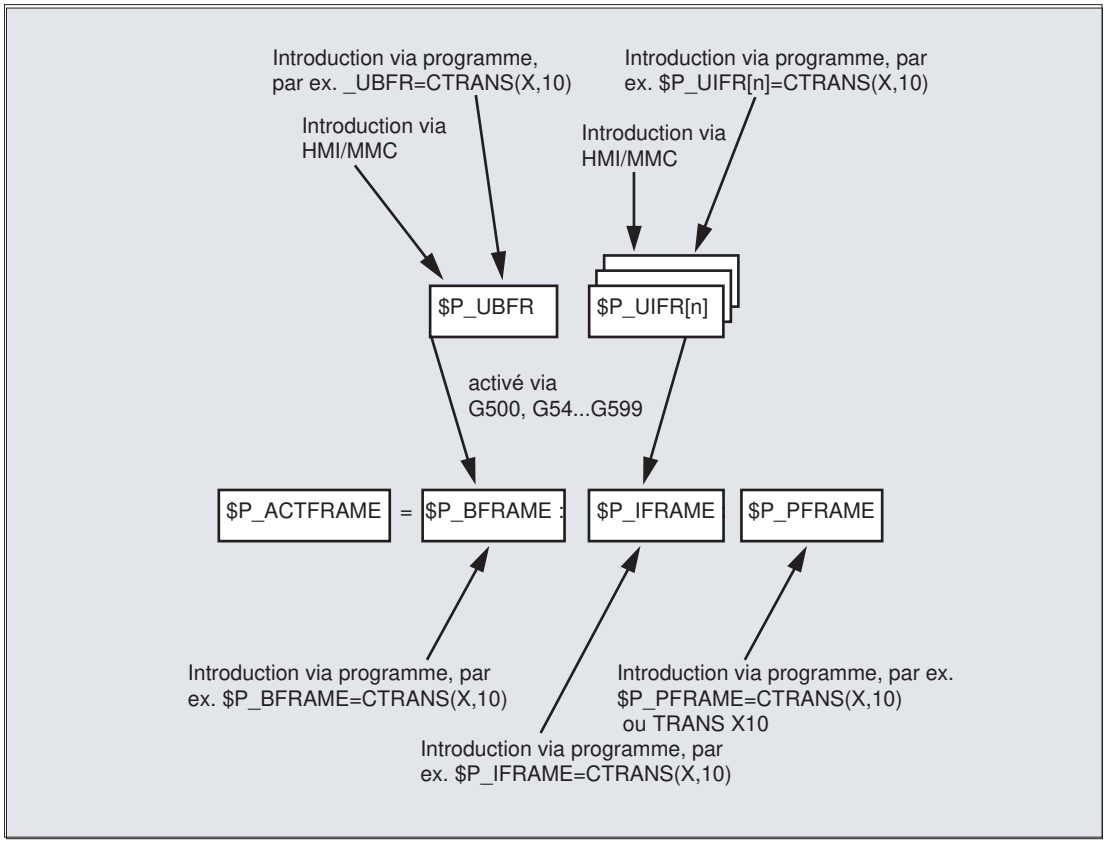


Figure 3-2 Variable frame $P_ACTFRAME

Si l'un des frames suivants `$P_BFRAME` / `$P_CHBFRAME[<n>]`, `$P_IFRAME` ou `$P_PFRAME` est modifié, le frame global courant `$P_ACTFRAME` est recalculé.



L'activation du frame de base ou du frame réglable après un reset dépend du réglage du PM 20110 RESET_MODE_MASK :

bit0=1, bit14=1 --> `$P_UBFR` (frame de base) activé

bit0=1, bit5=1 --> `$P_UIFR[$P_UIFRNUM]` (frame réglable) activé

Gestion des données : frames de base spécifiques à un canal $P_CHBFR[<n>]

Les frames de base de gestion des données sont lus/écrits via les variables frame $P_CHBFR[<n>]. La procédure d'écriture n'active pas immédiatement le frame de gestion des données dans le canal. L'activation du frame écrit intervient comme suit :

- Reset de canal et PM20110 $MC_RESET_MODE_MASK, Bit0 == 1 et Bit14 == 1
- Instruction `G500`, `G54` ... `G57`, `G505` ... `G599` (activation/désactivation des frames de base puis nouveau calcul du frame global courant)

Gestion des données : Frames réglables spécifiques à un canal $P_UIFR[<n>]

Les frames réglables de gestion des données sont lus/écrits via les variables frame $P_UIFR[<n>]. La procédure d'écriture n'active pas immédiatement le frame dans le canal. La prise en compte du frame écrit dans le canal intervient comme suit :

- Instruction `G500` (désactivation de tous les frames réglables ou décalages d'origine)
- Instruction `G54` ... `G57`, `G505` ... `G599` (activation d'un frame réglable ou d'un décalage d'origine)

Frame réglable actif	Frame de gestion des données	(correspond à l'instruction)
`$P_IFRAME` =	`$P_UIFR[0]`	`G500`
	`$P_UIFR[1]`	`G54`
	`$P_UIFR[2]`	`G55`
	`$P_UIFR[3]`	`G56`
	`$P_UIFR[4]`	`G57`
	`$P_UIFR[5]`	`G505`
	`$P_UIFR[6]`	`G506`
	...	...
	`$P_UIFR[99]`	`G599`

3.8.2 Affectations de valeurs aux frames

3.8.2.1 Affectation de valeurs directes (valeur d'axe, angle, échelle)

Vous pouvez affecter directement des valeurs à des frames ou à des variables frame dans le programme CN.

Syntaxe

Syntaxe

```
$P_PFRAME = CTRANS(X, <valeur du décalage>, Y, <valeur du décalage>,
Z, <valeur du décalage>, ...)
$P_PFRAME = ROT(X, <angle>, Y, <angle>, Z, <angle>, ...)
```

```
$P_UIFR[..] = CROT(X, <angle>, Y, <angle>, Z, <angle>,  ...
$P_PFRAME = CSCALE(X, <échelle>, Y, <échelle>, Z, <échelle>, ...)
$P_PFRAME = CMIRROR(X, Y, Z)
```

La syntaxe pour `$P_CHBFRAME[<n>]` est identique à `$P_PFRAME`.

Signification

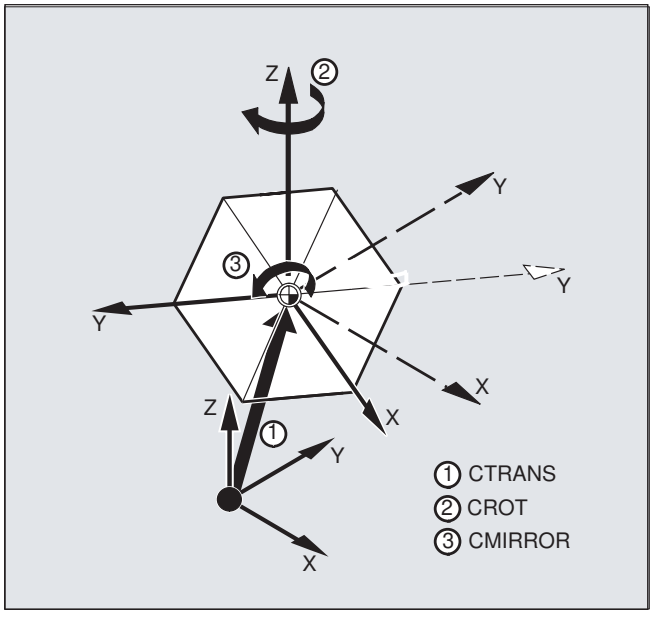
`CTRANS` :	Décalage (translation) dans les axes indiqués
`CROT` :	Rotation autour des axes indiqués
`CSCALE` :	Changement d'échelle dans les axes indiqués
`CMIRROR` :	Inversion du sens de l'axe indiqué
`X, Y, Z` :	Valeur du décalage en direction de l'axe géométrique indiqué
`<valeur du décalage>` :	Valeur du décalage
`<angle>` :	angle selon lequel s'effectue la rotation
`<échelle>` :	indication de l'échelle

Exemples

Affectations de valeurs à des composantes du frame programmable courant

Affectation de valeur aux composantes Translation, Rotation et Fonction miroir du frame programmable courant :

```
$P_PFRAME = CTRANS(X,10,Y,20,Z,5) : CROT(Z,45) : CMIRROR(Y)
```



Ecrire les composantes de rotation d'un frame

Affectation de valeurs aux trois axes de la composante Rotation du frame de gestion des données réglable `$P_UIFR` avec `CROT` :

```
$P_UIFR[5] = CROT(X, 0, Y, 0, Z, 0)
```

Affectation directe des valeurs individuelles à chaque axe de la composante Rotation du frame de gestion des données comme alternative :

```
$P_UIFR[5, Y, RT]=0
$P_UIFR[5, X, RT]=0
$P_UIFR[5, Z, RT]=0
```

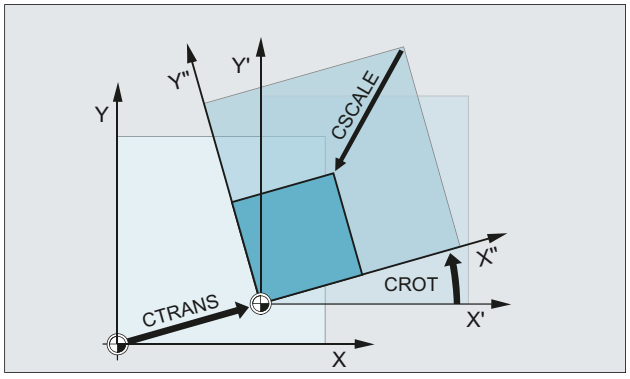
Description

Plusieurs opérations sur un frame peuvent être liées entre elles par l'opérateur de concaténation `:`. Les opérations sont alors exécutées l'une après l'autre de gauche à droite.

Exemple

Opérations concaténées sur $P_PFRAME avec décalage, rotation et homothétie :

```
$P_PFRAME = CTRANS(...) : CROT(...) : CSCALE...
```



3.8.2.2 Lecture et modification de composantes de frame (TR, FI, RT, SC, MI)

Vous avez la possibilité d'accéder à des données **individuelles** d'un frame, par exemple à une valeur de décalage ou à un angle de rotation bien défini. Vous pouvez modifier ces valeurs ou les affecter à une autre variable.

Syntaxe

`R10=$P_UIFR[$P_UIFNUM,X,RT]`	L'angle de rotation RT autour de l'axe X issu du décalage d'origine réglable actuellement valide $P_UIFRNUM doit être affecté à la variable R10.
`R12=$P_UIFR[25,Z,TR]`	La valeur de décalage TR dans Z, issue du bloc de données du frame n° 25 réglé, doit être affectée à la variable R12.

`R15=$P_PFRAME[Y,TR]`	La valeur de décalage TR en Y du frame programmable courant doit être affectée à la variable R15.
`$P_PFRAME[X,TR] = 25`	La valeur de décalage TR en X du frame programmable courant doit être modifiée. X25 s'applique dès à présent.

Signification

`$P_UIFRNUM:`	`Cette variable établit automatiquement le lien avec le décalage d'origine réglable actuellement valide.`
`P_UIFR[n,…,…]:`	`En indiquant le n° de frame n, vous accédez au frame réglable n° n`
	`Indication de la composante qui doit être lue ou modifiée :`
`TR:`	`TR Translation`
`FI:`	`FI Translation Fine`
`RT:`	`RT Rotation`
`SC:`	`SC Scale Changement d'échelle`
`MI:`	`MI Fonction miroir`
`X, Y, Z:`	`Est indiqué en outre (voir exemples) l'axe X, Y, Z correspondant.`

Plage de valeurs pour la rotation RT :

Rotation autour du 1er axe géométrique : -180° à +180°
Rotation autour du 2e axe géométrique : -90° à +90°
Rotation autour du 3e axe géométrique : -180° à +180°

Description

Appeler le frame

En indiquant la variable système $P_UIFRNUM, vous pouvez accéder directement au décalage d'origine courant, réglé avec $P_UIFR ou G54, G55, ...
($P_UIFRNUM contient le n° du frame courant réglé).

Vous appelez tous les autres frames réglables mémorisés $P_UIFR en indiquant le n° correspondant $P_UIFR[n].

Pour les variables frame prédéfinies et les frames que vous avez définis vous-même, indiquez le nom, par exemple $P_IFRAME.

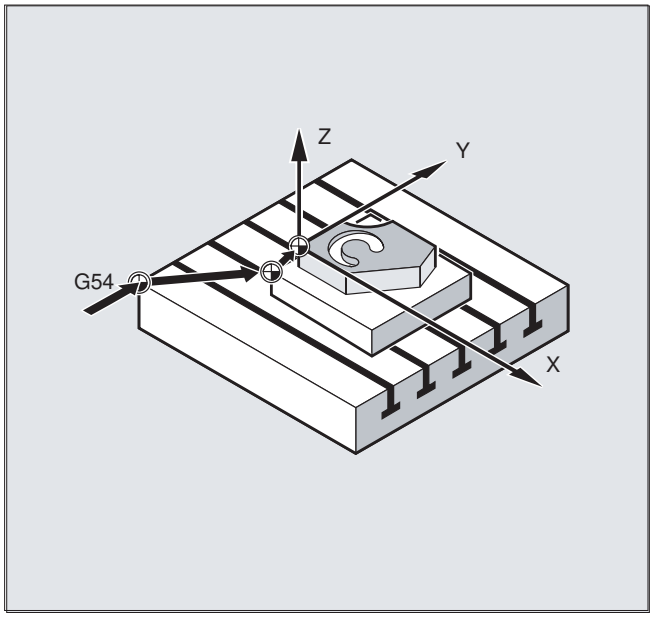
Appeler les données

Le nom de l'axe et la composante de frame à laquelle vous voulez accéder ou que vous voulez modifier figurent entre crochets droits, par ex. [X, RT] ou [Z, MI].

3.8.2.3 Calcul avec des frames

Dans le programme CN, un frame peut être affecté à un autre frame ou bien des frames peuvent être concaténés.

On utilise la concaténation de frames pour décrire, par exemple, plusieurs pièces disposées sur une palette et qui doivent être usinées en un seul cycle.



Pour la description des tâches d'usinage sur palette, les composantes de frame pourraient, par exemple, ne contenir que certaines valeurs partielles dont la concaténation pourraient générer différentes origines pièce.

Exemples

Affectations

Code de programme	Commentaire
`DEF FRAME REGLAGE_1`	`; Définition d'une variable frame locale`
`REGLAGE_1 = CTRANS(X,10)`	`; Affectation du résultat d'une fonction à la variable frame`
`$P_PFRAME = REGLAGE_1`	`; Affectation des variables frame au frame courant`
`DEF FRAME REGLAGE_4`	`; Définition d'une variable frame locale`
`REGLAGE_4 = $P_PFRAME`	`; Mise en mémoire tampon du frame courant dans les variables frame`
`...`	
`$P_PFRAME = REGLAGE_4`	`; Lecture du frame courant depuis les variables frame`

Concaténations

Les frames sont liés les uns aux autres dans l'ordre programmé par l'opérateur `:`. Les composantes de frame, comme les décalages, les rotations, etc., sont exécutées de manière additive les unes après les autres.

Code de programme	Commentaire

`$P_IFRAME = $P_UIFR[15] : $P_UIFR[16]`	`; Affectation du frame de résultat depuis la concaténation des`
	`; deux frames réglables de gestion de données au`
	`; frame global réglable actif.`
	`; Exemple d'application :`
	`; $P_UIFR[15] : Décalage`
	`; $P_UIFR[16] : Rotation`

`$P_UIFR[3] = $P_UIFR[4] : $P_UIFR[5]`	`; Affectation du frame de résultat depuis la concaténation des`
	`; deux frames réglables de gestion de données à un`
	`; autre frame réglable de gestion de données`

3.8.2.4 Définition des variables frame (DEF FRAME)

Outre les variables frame prédéfinies, il est également possible de définir des variables frame propres. Les variables frame propres sont des variables utilisateur de type FRAME. Le nom du frame peut être attribue librement dans le cadre des règles pour les variables utilisateur.

Avec les fonctions CTRANS, CROT, CSCALE, CMIRROR, les variables frame propres peuvent être affectées à des valeurs.

Syntaxe

```
DEF FRAME <nom>
```

Signification

`DEF FRAME` :	définition d'une variable utilisateur de type FRAME.
<Nom> :	nom de la variable frame

Exemple

Définition d'une variable frame "PALETTE" et affectation de valeurs de décalage et de rotation :

Code de programme	Commentaire
`DEF FRAME PALETTE`	`; Définir la variable frame PALETTE`
`PALETTE = CTRANS(...) : CROT(...)`	`; Affectation du frame de résultat de la concaténation du` `; décalage et de la rotation à la variable frame PALETTE`

3.8.3 Décalage grossier et fin (CTRANS, CFINE)

Décalage fin

Un décalage fin `CFINE(...)` peut être appliqué aux frames suivants :

- Frames réglables : $P_UIFR ou $P_IFRAME
- Cadres de base : $P_NCBFR[<n>], $P_CHBFR[<n>] / $P_CHBFRAMES[<n>] ou $P_ACTBFRAME
- Frame programmable : $P_PFRAME

Le décalage fin d'un frame se programme à l'aide de l'instruction `CFINE(...)`.

Décalage grossier

Le décalage grossier `CTRANS(...)` peut être appliqué à tous les frames.

Décalage total

Le décalage total découle de l'addition des décalages grossier et fin.

Paramètres machine

Déblocage du décalage fin

Le décalage fin est débloqué avec le paramètre machine :

PM18600 $MN_MM_FRAME_FINE_TRANS = 1

Syntaxe

Décalage fin

- Frame global
  - `<frame> = CFINE(<K_1>,<valeur>)`
  - `<frame> = CFINE(<K_1>,<valeur>, <K_2>, <valeur>)`
  - `<frame> = CFINE(<K_1>,<valeur>, <K_2>, <valeur>, <K_3>, <valeur>)`
- Composante de frame
  - `<frame>[<n>, <K_1>, FI] = <valeur>`

Décalage grossier

- Frame global
  - `<frame> = CTRANS(<K_1>,<valeur>)`
  - `<frame> = CTRANS(<K_1>,<valeur>, <K_2,<valeur>)`
  - `<frame> = CTRANS(<K_1>,<valeur>, <K_2,<valeur>, <K_3,<valeur>)`
- Composante de frame
  - `<frame>[<n>,<K_1>,TR] = <valeur>`

Spécialement pour le frame programmable $P_PFRAME :

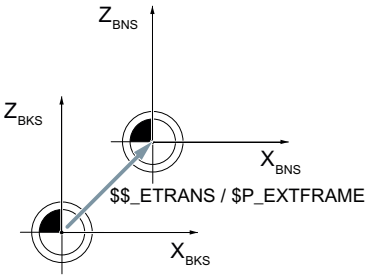
- TRANS <K_1> <valeur>
- TRANS <K_1> <valeur> <K_2> <valeur>
- TRANS <K_1> <valeur> <K_2> <valeur> <K_3> <valeur>

Signification

<frame> :	Frame, par exemple frame réglable de la gestion de données $P_UIFR[<n>]
CFINE :	Décalage fin, décalage additif.
CTRANS :	Décalage grossier, décalage absolu.
TRANS :	Uniquement frame programmable : Décalage grossier, décalage absolu.
<K_n> :	Axes de coordonnées X, Y, Z
<valeur> :	Valeur du décalage

3.8.4 Décalage d'origine externe ($AA_ETRANS)

Le décalage d'origine externe est un décalage linéaire entre le système de coordonnées de base (SCB) et le système de coordonnées de base (SBR).



Le décalage d'origine externe via $AA_ETRANS agit de deux manières, selon les paramètres machine :

1. La variable système $AA_ETRANS agit, après activation par le signal d'interface CN/AP, directement en tant que valeur de décalage
2. La valeur de la variable système $AA_ETRANS est reprise, après activation par le signal d'interface CN/AP, dans les frames système actifs $P:EXTFRAME et dans le frame de gestion de données $P_EXTFR. Enfin, le frame global actif $P_ACTFRAME est recalculé.

Paramètres machine

Concernant la variable système $AA_ETRANS, il convient de distinguer deux procédures sélectionnées via le paramètre machine suivant :

PM28082 $MC_MM_SYSTEM_FRAME_MASK,Bit1 = <valeur>

<Valeur>	Signification
0	Fonction : Ecriture directe de $AA_ETRANS[<axe>] par l'AP, l'IHM ou le programme CN. Déblocage de l'exécution du décalage d'origine de $AA_ETRANS[<axe>] dans le prochain bloc de déplacement possible : DB31, ... DBX3.0
1	Fonction : Activation du frame système actif $P:EXTFRAME et du frame de gestion de données $P_EXTFR Déblocage de l'exécution du décalage d'origine de $AA_ETRANS[<axe>] via : DB31, ... DBX3.0. Il se produit ensuite dans le canal : • arrêt de tous les déplacements dans le canal (hors axes d'ordre et d'AP) • Arrêt du prétraitement des blocs avec réorganisation (STOPRE) • Décalage grossier du frame actif $P_EXTFRAME[<axe>] = $AA_ETRANS[<axe>] • Décalage grossier du frame de gestion des données $P_EXTFR[<axe>] = $AA_ETRANS[<axe>] • Nouveau calcul du frame global actif $P_ACTFRAME • Exécution du décalage dans les axes programmés. • Poursuite du déplacement interrompu ou du programme CN

Programmation

- Syntaxe
  `$AA_ETRANS[<axe>] = <valeur>`
- Signification

`$AA_ETRANS` :	variable système pour mise en mémoire tampon du décalage d'origine externe
`<axe>` :	Axe de canal
`<valeur>` :	Valeur du décalage

Signal d'interface CN/AP

DB31, ... DBX3.0 = 0 → 1 ⇒ $P_EXTFRAME[<axe>] = $P_EXTFR[<axe>] = $AA_ETRANS[<axe>]

3.8.5 Forçage de valeurs réelles avec perte de l'état de référencement (PRESETON)

La procédure PRESETON() définit, pour un ou plusieurs axes, une nouvelle valeur réelle dans le système de coordonnées machine (SCM). Cela correspond à un décalage d'origine du SCM de l'axe. Ainsi, l'axe ne se déplace pas.

PRESETON déclenche un arrêt du prétraitement des blocs avec synchronisation. La position réelle est affectée à l'axe à l'arrêt seulement.

Si, pour PRESETON, l'axe n'est pas affecté au canal, la suite des opérations dépend de la configuration spécifique à l'axe de la procédure de permutation d'axe :

MD30552 $MA_AUTO_GET_TYPE

Etat de référencement

La définition d'une nouvelle valeur réelle dans le système de coordonnées machine remet à zéro l'état de référencement de l'axe machine :

DB31, ... DBX60.4 / .5 = 0 (référencé / synchronisé système de mesure 1 / 2)

Il est recommandé de n'utiliser PRESETON qu'avec des axes sans point de référence obligatoire.

Pour rétablir le système de coordonnées machine d'origine, le système de mesure de l'axe machine doit être de nouveau référencé, par exemple par accostage du point de référence depuis le programme pièce (G74).

⚠ PRUDENCE

Perte de l'état de référencement

La définition d'une nouvelle valeur réelle dans le système de coordonnées machine avec PRESETON remet l'état de référencement de l'axe machine sur "non référencé / synchronisé".

Programmation

Syntaxe

```
PRESETON(<axe_1>, <valeur_1> [, <axe_2>, <valeur_2>, ... <axe_8>,
<valeur_8>])
```

Signification

PRESETON :	Forçage de valeurs réelles avec perte de l'état de référencement	
	Arrêt du prétraitement des blocs :	oui
	Uniquement dans le bloc :	oui
<axe_x> :	Nom d'axe machine	
	Type :	AXIS
	Plage de valeurs :	noms d'axe machine définis dans le canal
<valeur_x> :	Nouvelle valeur réelle de l'axe machine dans le système de coordonnées machine (SCM) La saisie s'effectue dans le système de mesure actuellement en vigueur (inch/métrique) Une programmation active de diamètre (DIAMON) est prise en compte	
	Type :	REAL

Informations complémentaires

PRESETON dans les programmes CN

Une description détaillée de PRESETON dans les programmes CN figure dans la description fonctionnelle Fonctions de base.

PRESETON dans des actions synchrones

Une description détaillée de PRESETON dans les actions synchrones figure dans la description fonctionnelle des actions synchrones.

3.8.6 Forçage de valeurs réelles sans perte de l'état de référencement (PRESETONS)

La procédure PRESETONS() définit, pour un ou plusieurs axes, une nouvelle valeur réelle dans le système de coordonnées machine (SCM). Cela correspond à un décalage d'origine du SCM de l'axe. Ainsi, l'axe ne se déplace pas.

PRESETONS déclenche un arrêt du prétraitement des blocs avec synchronisation. La position réelle est affectée à l'axe à l'arrêt seulement.

Si, pour PRESETONS, l'axe n'est pas affecté au canal, la suite des opérations dépend de la configuration spécifique à l'axe de la procédure de permutation d'axe :

MD30552 $MA_AUTO_GET_TYPE

Etat de référencement

La définition d'une nouvelle valeur réelle dans le système de coordonnées machine (SCM) avec PRESETONS ne modifie **pas** l'état de référencement de l'axe machine.

Conditions

- **Type de capteur**
  PRESETONS n'est possible que pour les types de codeurs suivants du système de mesure actif :
  - MD30240 $MA_ENC_TYPE[<système de mesure>] = 0 (codeur simulé)
  - MD30240 $MA_ENC_TYPE[<système de mesure>] = 1 (codeur de signal brut)
- **Mode de référencement**
  PRESETONS n'est possible que pour les modes de référencement suivants du système de mesure actif :
  - MD34200 $MA_ENC_REFP_MODE[<système de mesure>] = 0 (aucun accostage du point de référence possible)
  - MD34200 $MA_ENC_REFP_MODE[<système de mesure>] = 1 (référencement de systèmes de mesure relatifs, rotatifs ou linéaires : impulsion top zéro sur la voie du codeur)

Programmation

Syntaxe

```
PRESETONS(<axe_1>, <valeur_1> [, <axe_2>, <valeur_2>, ... <axe_8>,
<valeur_8>])
```

Signification

PRESETONS :	Forçage de valeurs réelles sans perte de l'état de référencement	
	Arrêt du prétraitement des blocs :	oui
	Uniquement dans le bloc :	oui
<axe_x> :	Nom d'axe machine	
	Type :	AXIS
	Plage de valeurs :	noms d'axe machine définis dans le canal
<valeur_x> :	Nouvelle valeur réelle courante de l'axe machine dans le système de coordonnées machine (SCM) La saisie s'effectue dans le système de mesure actif (inch/métrique) Une programmation active de diamètre (DIAMON) est prise en compte	
	Type :	REAL

Informations complémentaires

PRESETONS dans les programmes CN

Une description détaillée de PRESETONS dans les programmes CN figure dans la description fonctionnelle Fonctions de base.

PRESETONS dans des actions synchrones

Une description détaillée de PRESETONS dans les actions synchrones figure dans la description fonctionnelle Actions synchrones.

3.8.7 Calcul du frame à partir de 3 points mesurés dans l'espace (MEAFRAME)

La fonction MEAFRAME s'utilise en assistance des cycles de mesure Elle calcule le frame à partir de trois points idéaux et des points mesurés correspondants.

Lorsqu'une pièce est positionnée pour l'usinage, sa position dans le système de coordonnées cartésiennes machine est en général décalée et pivotée par rapport à sa position idéale. Pour un usinage ou une mesure précise, un réglage physique onéreux ou une modification des déplacements dans le programme pièce est nécessaire.

Un frame peut être déterminé par palpage de trois points dans l'espace, dont les positions idéales sont connues. Le palpage s'effectue avec un capteur optique ou à contact qui accoste des trous ou des billes de mesure qui ont été spécialement positionnés avec une grande précision sur la plaque support.

Syntaxe

```
MEAFRAME(<points idéaux>,<points de mesure>,<qualité>)
```

Signification

MEAFRAME :	Appel de fonction		
<points idéaux> :	Tableau REAL à 2 dim. qui contient les 3 coordonnées des points idéaux		
<points de mesure> :	Tableau REAL à 2 dim. qui contient les 3 coordonnées des points mesurés		
<qualité> :	Variable permettant de retourner les informations de qualité du calcul de FRAME		
	Type :	VAR REAL	
	Valeur :	-1	Les points idéaux se trouvent approximativement sur une droite : le frame n'a pas pu être calculé. La variable FRAME en retour comprend un frame neutre.
		-2	Les points mesurés se trouvent approximativement sur une droite : le frame n'a pas pu être calculé. La variable FRAME en retour comprend un frame neutre.
		-4	Le calcul de la matrice de rotation est impossible pour une autre raison.
		≥ 0,0	Somme des distorsions (écarts entre les points) nécessaire pour transférer passer le triangle mesuré dans un triangle congruent au triangle idéal.

Remarque

Qualité de la mesure

Pour que les coordonnées mesurées puissent être mises en relation avec les coordonnées idéales par une rotation/translation combinées, il doit y avoir congruence entre le triangle formé par les points mesurés et le triangle idéal. Ceci est obtenu par un algorithme de compensation qui minimise la somme des carrés des écarts qui permettent de passer du triangle mesuré au triangle idéal.

La distorsion effectivement nécessaire des points mesurés étant le reflet de la qualité de la mesure, elle est fournie par MEAFRAME en tant que variable supplémentaire.

Remarque

Le frame créé par MEAFRAME peut être transformé en un autre frame de la chaîne de frames au moyen de la fonction ADDFRAME (voir l'exemple "Concaténation avec ADDFRAME").

Exemples

Exemple 1 :

Programme pièce 1 :

Code de programme

```
...
DEF FRAME CORR_FRAME
```

Définition de points de mesure :

Code de programme	**Commentaire**
DEF REAL IDEAL_POINT[3,3]= SET(10.0,0.0,0.0,0.0,10.0,0.0,0.0,0.0,10.0)	
DEF REAL MEAS_POINT[3,3]= SET(10.1,0.2,-0.2,-0.2,10.2,0.1,-0.2,0.2,9.8)	; Pour le test.
DEF REAL FIT_QUALITY=0	
DEF REAL ROT_FRAME_LIMIT=5	; Permet une rotation de la position de la pièce de 5 degrés au maximum.
DEF REAL FIT_QUALITY_LIMIT=3	; Permet au maximum un décalage de 3 mm entre le triangle idéal et le triangle mesuré.
DEF REAL SHOW_MCS_POS1[3]	
DEF REAL SHOW_MCS_POS2[3]	
DEF REAL SHOW_MCS_POS3[3]	

Code de programme	**Commentaire**
N100 G01 G90 F5000	
N110 X0 Y0 Z0	
N200 CORR_FRAME=MEAFRAME(IDEAL_POINT,MEAS_POINT,FIT_QUALITY)	
N230 IF FIT_QUALITY < 0	
SETAL(65000)	
GOTOF NO_FRAME	
ENDIF	
N240 IF FIT_QUALITY > FIT_QUALITY_LIMIT	
SETAL(65010)	
GOTOF NO_FRAME	
ENDIF	
N250 IF CORR_FRAME[X,RT] > ROT_FRAME_LIMIT	; Limitation du 1er angle RPY.
SETAL(65020)	
GOTOF NO_FRAME	
ENDIF	
N260 IF CORR_FRAME[Y,RT] > ROT_FRAME_LIMIT	; Limitation du 2e angle RPY.
SETAL(65021)	
GOTOF NO_FRAME	
ENDIF	
N270 IF CORR_FRAME[Z,RT] > ROT_FRAME_LIMIT	; Limitation du 3e angle RPY.
SETAL(65022)	
GOTOF NO_FRAME	
ENDIF	

Code de programme	**Commentaire**
N300 $P_IFRAME=CORR_FRAME	; Activation du frame de balayage par un frame réglable.
	; Vérification du frame en positionnant les axes géométriques sur les points idéaux.
N400 X=IDEAL_POINT[0,0] Y=IDEAL_POINT[0,1] Z=IDEAL_POINT[0,2]	
N410 SHOW_MCS_POS1[0]=$AA_IM[X]	
N420 SHOW_MCS_POS1[1]=$AA_IM[Y]	
N430 SHOW_MCS_POS1[2]=$AA_IM[Z]	
N500 X=IDEAL_POINT[1,0] Y=IDEAL_POINT[1,1] Z=IDEAL_POINT[1,2]	
N510 SHOW_MCS_POS2[0]=$AA_IM[X]	
N520 SHOW_MCS_POS2[1]=$AA_IM[Y]	
N530 SHOW_MCS_POS2[2]=$AA_IM[Z]	
N600 X=IDEAL_POINT[2,0] Y=IDEAL_POINT[2,1] Z=IDEAL_POINT[2,2]	
N610 SHOW_MCS_POS3[0]=$AA_IM[X]	
N620 SHOW_MCS_POS3[1]=$AA_IM[Y]	
N630 SHOW_MCS_POS3[2]=$AA_IM[Z]	
N700 G500	; Désactiver le frame réglable, car préréglage avec le frame nul (sans indication de valeur).
No_FRAME	; Désactivation du frame réglable, car préaffecté avec le frame nul (sans indication de valeur).
M0	
M30	

Exemple 2 : concaténation de frames

Concaténation de MEAFRAME à des fins de correction

La fonction MEAFRAME fournit un frame de correction. Si ce frame de correction est concaténé avec le frame réglable $P_UIFR[1] qui était actif lors de l'appel de la fonction (par ex. G54), on obtient un frame réglable pour d'autres calculs de déplacement ou d'usinage.

Concaténation avec ADDFRAME

Si ce frame de correction est actif à un autre endroit de la chaîne de frames ou si d'autres frames sont actifs avant le frame réglable, la fonction ADDFRAME peut être utilisée pour la concaténation à l'un des frames de base spécifiques à un canal ou à un frame système.

Ne doivent pas être actifs dans les frames :

- la fonction miroir avec MIRROR
- la mise à l'échelle avec SCALE

Les paramètres d'entrée pour les valeurs réelles et les valeurs de consigne sont les coordonnées de la pièce. Dans le système de base de la commande, ces coordonnées doivent toujours être indiquées en mm ou en inch (G71/G70) en tant que mesure se rapportant au rayon (DIAMOF).

Des informations complémentaires sur ADDFRAME figurent dans la description fonctionnelle Fonctions de base.

3.8.8 Frames à définition globale

Les frames à définition globale sont disponibles une seule fois par commande, pour tous les canaux. Les frames à définition globale peuvent être écrasés et lus par tous les canaux. L'activation des frames à définition globale a lieu dans le canal correspondant.

Les frames à définition globale permettent d'appliquer des changements d'échelle avec décalages et des fonctions miroir à des **axes de canal ou à des axes machine**.

Liens géométriques et concaténations de frames

Dans le cas de ces frames, il n'existe pas de lien géométrique entre les axes. C'est pourquoi les rotations et la programmation de descripteurs d'axes géométriques sont impossibles.

Les frames à définition globale ne peuvent contenir aucune composante de rotation. La programmation d'une rotation est refusée avec l'émission de l'alarme 18310 "Canal %1 Bloc %2 Frame : rotation non autorisée".

La concaténation de frames à définition globale et de frames spécifiques à un canal est possible. Le frame résultant contient toutes les composantes des frames, y compris les rotations pour tous les axes. L'affectation d'un frame comportant des composantes de rotation à un frame à définition globale est refusée avec émission de l'alarme "Frame : rotation non autorisée".

Frames à définition globale

Frames de base à définition globale $P_NCBFR[n]

Jusqu'à 8 frames de base à définition globale peuvent être configurés :

Des frames de base spécifiques à un canal peuvent être actifs simultanément.

Les frames de base à définition globale peuvent être écrasés et lus par tous les canaux d'une commande. Lors de l'écrasement d'un frame à définition globale, l'utilisateur doit assurer la coordination entre les canaux. Cela est par exemple réalisable avec des marques wait (WAITMC).

Remarque

Constructeur de machines

Le nombre de frames de base globaux est configuré par des paramètres machine.

Informations complémentaires : Description fonctionnelle Fonctions de base

Frames réglables $P_UIFR[n]

Tous les frames réglables G500, G54...G599 peuvent être configurés comme étant soit à définition globale, soit spécifiques à un canal.

Remarque

Constructeur de machines

Tous les frames paramétrables peuvent être reconfigurés en frames globaux à l'aide du MD18601 $MN_MM_NUM_GLOBAL_USER_FRAMES.

Comme descripteurs d'axes pour les instructions Frame, il est possible d'utiliser des descripteurs d'axes de canaux et des descripteurs d'axes machine. La programmation de descripteurs d'axes géométriques est refusée avec émission d'une alarme.

3.8.8.1 Frames spécifiques à un canal ($P_CHBFR, $P_UBFR)

Les frames réglables ou les frames de base peuvent être lus via le programme pièce et via OPI par la commande et par l'AP.

Le décalage fin est également possible pour les frames à définition globale. L'inhibition de frames à définition globale se déroule comme pour les frames spécifiques à un canal, avec G53, G153, SUPA et G500.

Constructeur de machine

Le paramètre machine PM28081 $MC_MM_NUM_BASE_FRAMES permet de définir le nombre de frames de base dans un canal. En configuration standard, il existe au moins un frame de base par canal. 8 frames de base sont possibles au maximum par canal. En plus de ces 8 frames de base, 8 frames de base à définition globale pour NCU peuvent également exister dans le canal.

Frames spécifiques à un canal

$P_CHBFR[n]

La variable système $P_CHBFR[n] permet de lire et d'écraser les frames de base. Lors de l'écrasement d'un frame de base, le frame de base concaténé résultant n'est pas activé immédiatement, mais uniquement lors de l'exécution d'une instruction G500, G54 ... G599. Cette variable sert principalement à enregistrer les opérations d'écriture dans le frame de base effectuées par l'IHM ou l'AP. Ces variables frame sont sauvegardées.

Premier frame de base du canal

Un écrasement de la variable prédéfinie $P_UBFR n'active pas simultanément le frame de base d'indice 0 ; l'activation n'a lieu que lors de l'exécution d'une instruction G500, G54 ... G599. Cette variable peut également être lue et écrasée dans le programme.

$P_UBFR

$P_UBFR est identique à $P_CHBFR[0]. En version standard, il y a toujours un frame de base dans le canal, de sorte que la variable système est compatible avec les versions de logiciel précédentes. S'il n'y a pas de frame de base spécifique au canal, l'alarme "Frame : instruction non autorisée" est émise lors de l'écriture ou de la lecture.

3.8.8.2 Frames actifs dans un canal

Les frames actifs dans un canal sont introduits par le programme pièce via les variables système respectives de ces frames. Les frames système en font également partie. Ces variables système permettent de lire et d'écraser le frame système courant dans le programme pièce.

Frames actifs courants dans un canal

Vue d'ensemble

Frames système courants	pour :
$P_PARTFRAME	TCARR et PAROT
$P_SETFRAME	Préréglage des mémoires de valeurs réelles et effleurement
$P_EXTFRAME	Décalage d'origine externe
$P_NCBFRAME[n]	Frames de base courants à définition globale
$P_CHBFRAME[n]	Frames de base courants spécifiques à un canal
$P_BFRAME	1er frame de base courant du canal
$P_ACTBFRAME	Frame de base résultant
$P_CHBFRMASK et $P_NCBFRMASK	Frame de base résultant
$P_IFRAME	Frame réglable courant
Frames système courants	pour :
$P_TOOLFRAME	TOROT et TOFRAME
$P_WPFRAME	Points de référence de la pièce
$P_TRAFRAME	Transformations
$P_PFRAME	Frame programmable courant
Frame système courant	pour :
$P_CYCFRAME	Cycles
P_ACTFRAME	Frame résultant courant
Concaténation de frames	Le frame courant contient le frame de base résultant.

$P_NCBFRAME [n] Frames de base courants à définition globale

La variable système $P_NCBFRAME[n] permet de lire et d'écraser les éléments de tableau des frames de base courants à définition globale. Le frame de base résultant est pris en considération dans le canal lors de l'opération d'écriture.

Le frame modifié n'est actif que dans le canal dans lequel le frame a été programmé. Si le frame doit être modifié pour tous les canaux d'une commande, $P_NCBFR[n] et $P_NCBFRAME[n] doivent être définis simultanément. Ensuite, le frame doit encore être activé dans les autres canaux à l'aide de G54, par ex. En cas d'écrasement d'un frame de base, le frame de base résultant est recalculé.

$P_CHBFRAME[n] Frames de base courants spécifiques à un canal

La variable système $P_CHBFRAME[n] permet de lire et d'écraser les éléments de tableau des frames de base courants spécifiques à un canal. Le frame de base résultant est pris en considération dans le canal lors de l'opération d'écriture. En cas d'écrasement d'un frame de base, le frame de base résultant est recalculé.

$P_BFRAME 1er frame de base courant du canal

La variable frame prédéfinie $P_BFRAME permet de lire et d'écraser, dans le programme pièce, le frame de base courant d'indice 0, qui est valide dans le canal. Ce frame est pris immédiatement en considération.

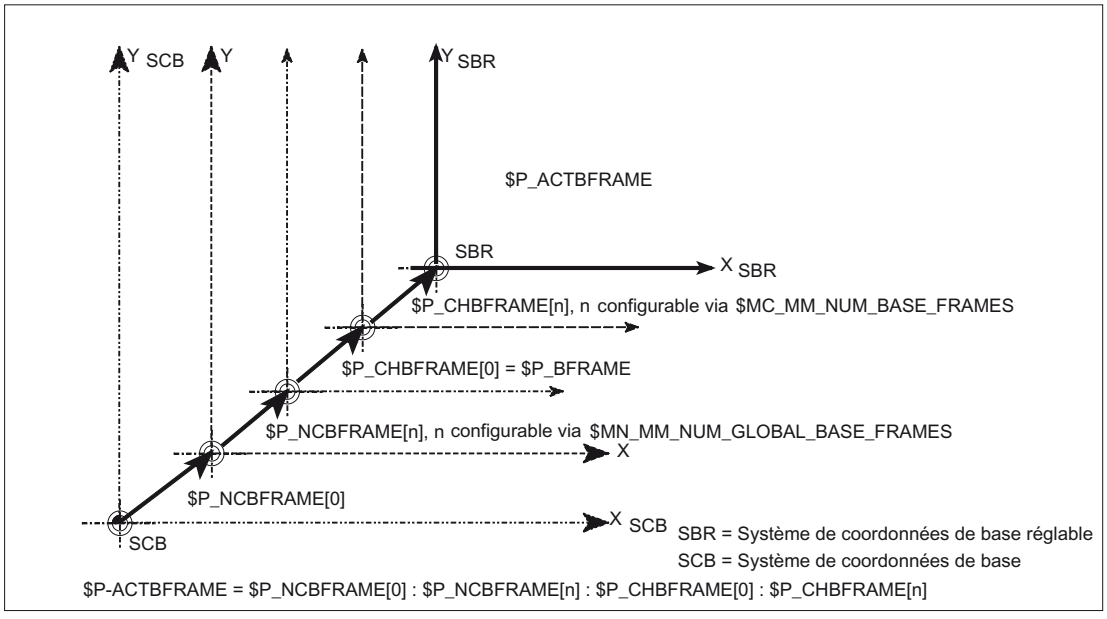
$P_BFRAME est identique à $P_CHBFRAME[0]. Par défaut, la variable système a toujours une valeur valide. S'il n'y a pas de frame de base spécifique au canal, l'alarme "Frame : instruction non autorisée" est émise lors de l'écriture ou de la lecture.

$P_ACTBFRAME Frame de base résultant

La variable $P_ACTFRAME détermine le frame de base résultant concaténé. Elle n'est accessible qu'en lecture.

$P_ACTFRAME correspond à :

$P_NCBFRAME[0] : ... : $P_NCBFRAME[n] : $P_CHBFRAME[0] : ... : $P_CHBFRAME[n].



$P_CHBFRMASK et $P_NCBFRMASK Frame de base résultant

Avec les variables système $P_CHBFRMASK et $P_NCBFRMASK, l'utilisateur peut sélectionner les frames de base pour le calcul du frame de base "résultant". Ces variables ne peuvent être programmées que dans le programme et lues via l'OPI. Les valeurs de ces variables sont interprétées en tant que masques binaires et indiquent les éléments de $P_ACTFRAME à prendre en considération.

$P_CHBFRMASK permet d'indiquer les frames de base spécifiques au canal à prendre en considération et $P_NCBFRMASK, les frames de base à définition globale à prendre en considération.

La programmation de ces variables entraîne un nouveau calcul du frame de base résultant et du frame résultant. Les valeurs de $P_CHBFRMASK et $P_NCBFRMASK après Reset et les préréglages sont les suivantes :

$P_CHBFRMASK = $MC_CHBFRAME_RESET_MASK

$P_NCBFRMASK = $MC_CHBFRAME_RESET_MASK

Exemple :

```
$P_NCBFRMASK = 'H81' ;$P_NCBFRAME[0] : $P_NCBFRAME[7]
$P_CHBFRMASK = 'H11' ;$P_CHBFRAME[0] : $P_CHBFRAME[4]
```

$P_IFRAME Frame réglable courant

La variable frame prédéfinie $P_IFRAME permet de lire et d'écraser, dans le programme pièce, le frame réglable courant qui est valide dans le canal. Le frame réglable écrasé est pris immédiatement en considération.

Dans le cas des frames réglables à définition globale, le frame modifié n'est actif que dans le canal dans lequel il a été programmé. Si le frame doit être modifié pour tous les canaux d'une commande, $P_UIFR[n] et $P_IFRAME doivent être définis simultanément. Ensuite, le frame correspondant doit encore être activé dans les autres canaux à l'aide de G54, par ex.

$P_PFRAME Frame programmable courant

$P_PFRAME est le frame programmable obtenu à partir de la programmation de TRANS/ ATRANS, G58/G59, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR ou de l'affectation de CTRANS, CROT, CMIRROR, CSCALE au FRAME programmable.

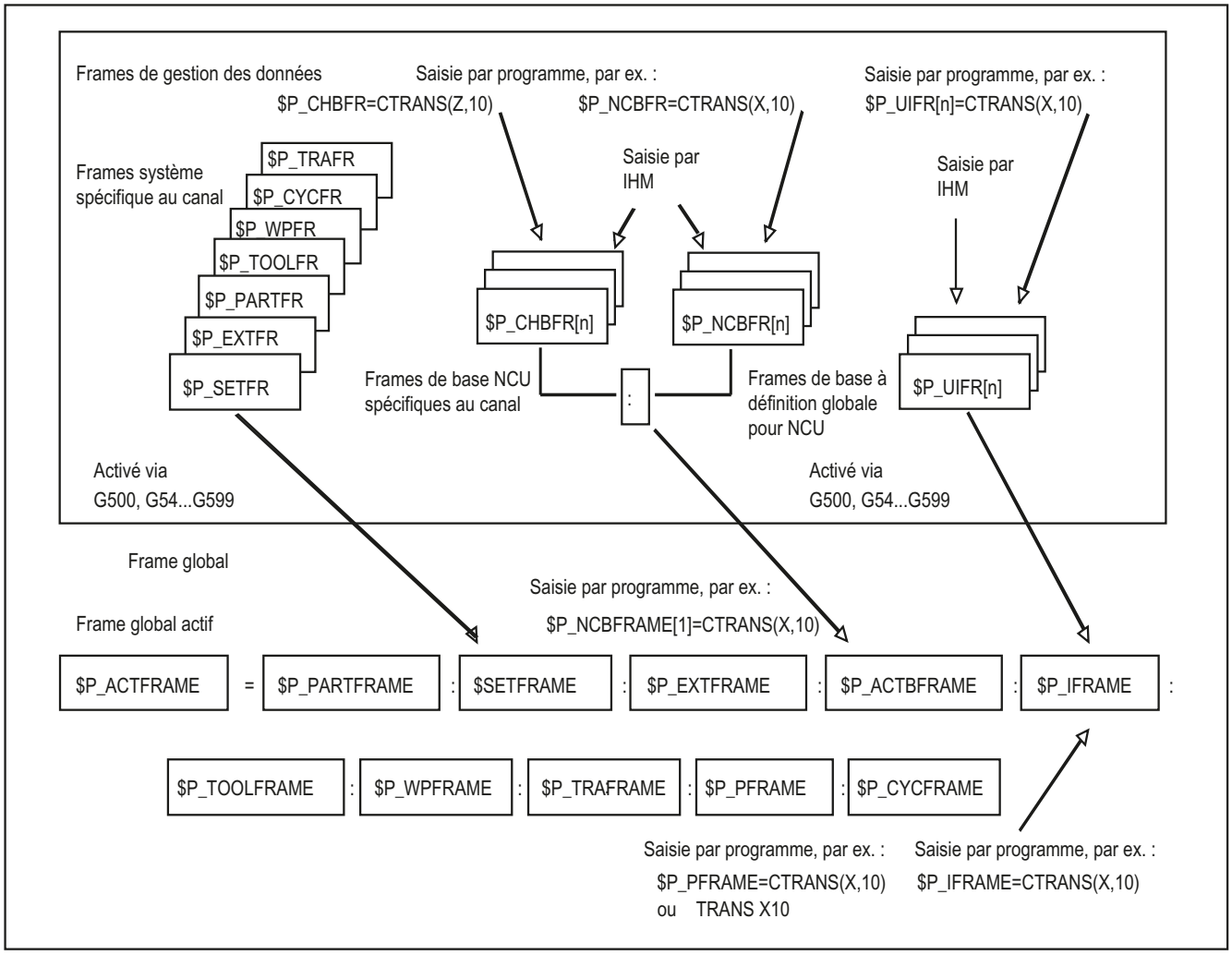
Variable de frame programmable courante qui établit la relation entre le système de coordonnés-pièce réglable (SPR) et le système de coordonnées-pièce (SCP).

P_ACTFRAME Frame résultant courant

Le frame résultant courant $P_ACTFRAME résulte de la concaténation de tous les frames de base, du frame réglable courant et du frame programmable. Le frame courant est toujours actualisé si une composante de frame subit une modification.

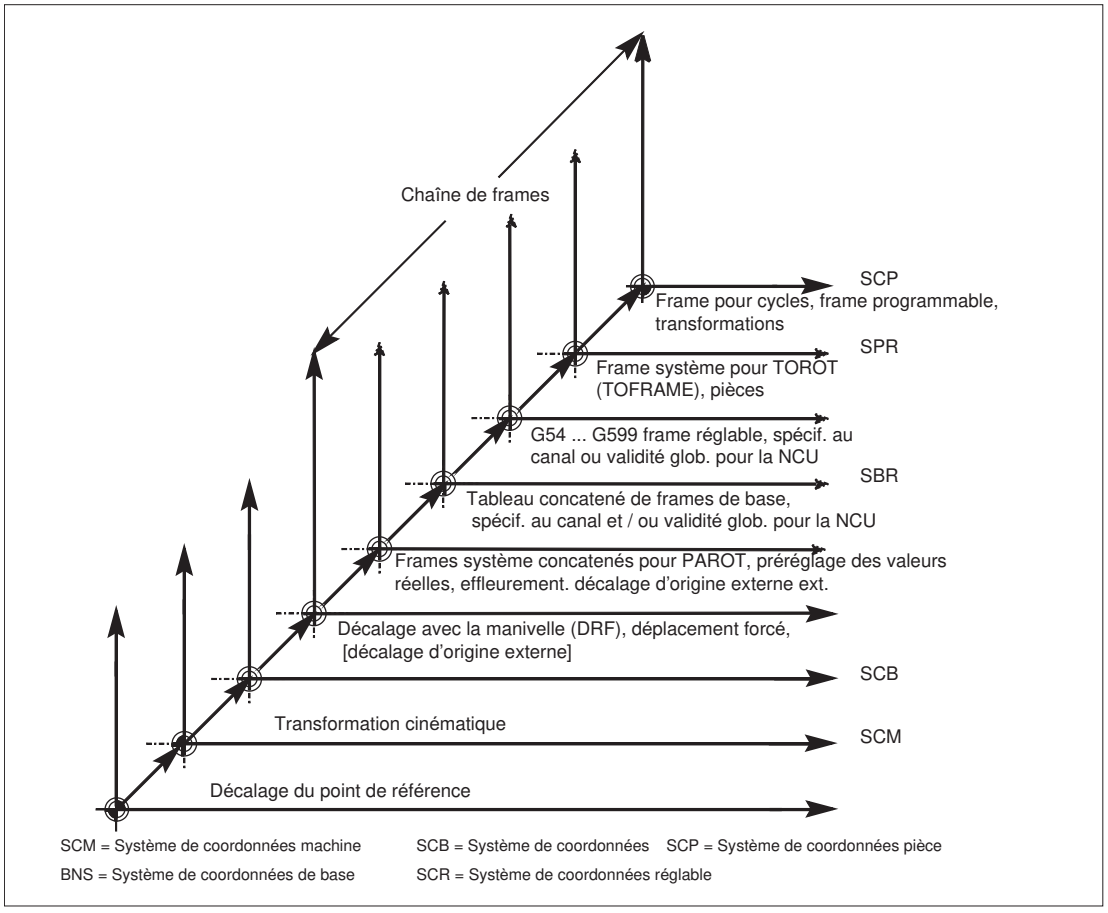
`$P_ACTFRAME` correspond à :

```
$P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME : $P_ACTBFRAME : $P_IFRAME :
$P_TOOLFRAME : $P_WPFRAME : $P_TRAFRAME : $P_PFRAME : $P_CYCFRAME
```


Frames de gestion des données
Saisie par programme, par ex. :
$P_CHBFR=CTRANS(Z,10)
$P_NCBFR=CTRANS(X,10)
Saisie par programme, par ex. :
$P_UIFR[n]=CTRANS(X,10)
Frames système spécifique au canal
$P_TRAFR
$P_CYCFR
$P_WPFR
$P_TOOLFR
$P_PARTFR
$P_EXTFR
$P_SETFR
Saisie par IHM
Saisie par IHM
$P_CHBFR[n]
$P_NCBFR[n]
$P_UIFR[n]
Frames de base NCU spécifiques au canal
:
Frames de base à définition globale pour NCU
Activé via G500, G54...G599
Activé via G500, G54...G599
Frame global
Saisie par programme, par ex. :
$P_NCBFRAME[1]=CTRANS(X,10)
Frame global actif
$P_ACTFRAME = $P_PARTFRAME : $SETFRAME : $P_EXTFRAME : $P_ACTBFRAME : $P_IFRAME :
$P_TOOLFRAME : $P_WPFRAME : $P_TRAFRAME : $P_PFRAME : $P_CYCFRAME
Saisie par programme, par ex. :
$P_PFRAME=CTRANS(X,10)
ou TRANS X10
Saisie par programme, par ex. :
$P_IFRAME=CTRANS(X,10)

Concaténation de frames

Le frame courant résulte de la concaténation du frame de base résultant, du frame réglable, du frame système et du frame programmable (voir frame résultant courant, ci-dessus).



3.9 Transformations

3.9.1 Programmation générale des types de transformation

Les transformations adaptent la commande à différentes cinématiques de machine. L'activation d'une transformation s'effectue par un appel de fonction dans le programme CN. Ses paramètres permettent de déterminer l'orientation de l'outil dans l'espace, ainsi que les mouvements d'orientation des axes rotatifs.

Avec les transformations d'orientation (transformations trois, quatre et cinq axes), les indications de position programmées se rapportent toujours à la pointe de l'outil qui est asservie perpendiculairement à la surface d'usinage dans l'espace. Les coordonnées cartésiennes sont converties du système de coordonnées de base au système de coordonnées machine et se rapportent aux axes géométriques. Ceux-ci décrivent le point de fonctionnement. Les axes rotatifs virtuels décrivent les orientations de l'outil dans l'espace.

Lors des transformations cinématiques (TRANSMIT, TRACYL, TRAANG), il est possible de programmer des positions dans le système de coordonnées cartésiennes. La commande transforme les déplacements programmés dans le système de coordonnées cartésiennes en déplacements des axes machine réels.

Programmation

Activation d'une transformation d'orientation

```
TRAORI(<numéro transfo>,<vecteur d'orientation>,<offset de l'axe rotatif>)
```

Activation d'une transformation cinématique

```
TRANSMIT(<numéro transfo>)
```

```
TRACYL(<diamètre de travail>,<numéro transfo>)
```

```
TRAANG(<angle de l'axe en oblique>,<numéro transfo>)
```

Désactivation de la transformation active

```
TRAFOOF
```

Transformation d'orientation

Transformations trois, quatre et cinq axes (TRAORI)

Pour usiner de manière optimale des surfaces à relief irrégulier dans la zone de travail de la machine, les machines-outils ont besoin d'axes supplémentaires en plus des trois axes linéaires X, Y et Z. Les axes supplémentaires décrivent l'orientation dans l'espace et sont nommés ci-après

"axes d'orientation". Ils sont mis à disposition sous forme d'axes de rotation sur quatre types de machines dotées d'une cinématique différente :

1. Tête orientable sur deux axes
   Exemple : tête d'outil à cardan avec un axe rotatif parallèle à un axe linéaire, le plateau porte-outil étant fixe.
2. Plateau tournant sur deux axes
   Exemple : tête orientable fixe avec plateau porte-outil pivotant sur deux axes
3. Tête orientable sur un axe et plateau tournant sur un axe
   Exemple : tête orientable pivotante avec outil orientable et plateau porte-outil pivotant sur un axe
4. Tête orientable sur deux axes et plateau tournant sur un axe
   Exemple : plateau porte-outil pivotant sur un axe et tête orientable pivotante avec outil pivotant autour de son propre axe

Les **transformations 3 et 4 axes** sont des formes particulières des transformations 5 axes et se programment de manière analogue aux transformations 5 axes.

Grâce à l'étendue de ses fonctions, la **transformation générique 3/4/5/6 axes** couvre les transformations pour les axes rotatifs disposés à angle droit, ainsi que pour la tête de fraisage à cardan. Comme toute autre transformation d'orientation, elle peut être activée avec TRAORI pour ces quatre types de machine. Dans le cas de la transformation générique 5/6 axes, l'orientation de l'outil dispose d'un troisième degré de liberté permettant, en plus d'une direction quelconque de l'outil dans l'espace, de le faire tourner autour de son propre axe.

Informations complémentaires : Description fonctionnelle Transformations

Position initiale d'orientation de l'outil indépendante de la cinématique

ORIRESET

Lorsque une transformation d'orientation est active avec TRAORI, alors ORIRESET peut indiquer les trois positions initiales de trois axes d'orientation maxi avec des paramètres A, B, C en option. L'affectation des paramètres programmés aux axes rotatifs s'effectue dans l'ordre des axes d'orientation défini par la transformation. Avec la programmation de ORIRESET(A, B, C), les axes d'orientation sont parcourus de façon linéaire et synchronisée de leur position momentanée à la position initiale indiquée.

Transformations cinématiques

TRANSMIT et TRACYL

Pour les fraisages à effectuer sur des tours, les types d'usinage suivants peuvent être activés pour la transformation convenue :

- usinage frontal sans démontage de la pièce avec TRANSMIT
- usinage de rainures quelconques sur des corps cylindriques avec TRACYL

TRAANG

Si l'axe de pénétration doit aussi pouvoir pénétrer en oblique (pour une utilisation en rectification par exemple), avec TRAANG, il est possible de programmer un angle paramétrable pour la transformation convenue.

Déplacement PTP cartésien

Ce que l'on appelle le "déplacement PTP cartésien", permettant de programmer jusqu'à 8 positions d'articulation (STAT=...) différentes, fait également partie de la transformation cinématique. Les positions sont programmées dans le système de coordonnées cartésiennes, le déplacement de la machine s'effectuant en coordonnées machine.

Informations complémentaires : Description fonctionnelle Transformations

Transformations concaténées

Il est possible d'effectuer deux transformations successives. Pour la deuxième transformation ainsi concaténée, les trajets des axes sont repris à partir de la première transformation.

La première transformation peut être :

- Transformation de l'orientation TRAORI
- Transformation polaire TRANSMIT
- Transformation de cylindre TRACYL
- Transformation d'axe oblique TRAANG

La deuxième transformation doit être de type "axe incliné TRAANG".

3.9.1.1 Mouvements d'orientation lors des transformations

Déplacements d'outils et déplacements d'orientation

Les déplacements des orientations programmables dépendent en premier lieu du type de machine. pour une transformation trois, quatre ou cinq axes avec TRAORI, les axes de rotation ou les axes linéaires orientables décrivent les déplacements d'orientation de l'outil.

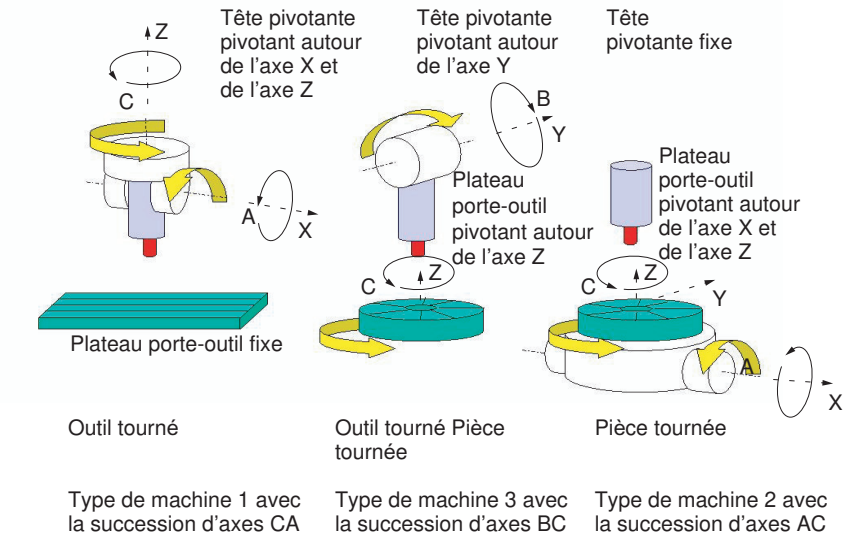
Toute modification de position des axes rotatifs participant à la transformation d'orientation entraîne des déplacements compensateurs des autres axes machine. Ce faisant, la position de la pointe de l'outil reste inchangée.

Les mouvements d'orientation de l'outil peuvent être programmés via les descripteurs d'axe rotatif A..., B..., C... des axes virtuels par l'indication d'angles d'Euler ou d'angles RPY, de vecteurs normaux à la direction ou à la surface, de vecteurs normés pour l'axe de rotation d'un cône ou pour l'orientation intermédiaire sur l'enveloppe d'un cône, selon l'application.

Lors de la transformation cinématique avec TRANSMIT, TRACYL et TRAANG, la commande transforme les déplacements programmés du système de coordonnées cartésiennes en déplacements des axes machine réels.

Cinématique de machine avec transformation trois, quatre et cinq axes TRAORI

Soit l'outil soit le plateau tournant est orientable avec deux axes rotatifs au plus. La combinaison d'une tête pivotante sur un axe et d'un plateau tournant lui aussi sur un axe est également possible.



Type de machine	Programmation de l'orientation
Transformation trois axes Types de machines 1 et 2	Programmation de l'orientation de l'outil uniquement dans le plan perpendiculaire à l'axe rotatif. Il existe **deux** axes de translation (axes linéaires) et **un** axe de rotation (axe rotatif).
Transformation quatre axes Types de machines 1 et 2	Programmation de l'orientation de l'outil uniquement dans le plan perpendiculaire à l'axe rotatif. Il existe **trois** axes de translation (axes linéaires) et **un** axe de rotation (axe rotatif).
Transformation cinq axes Types de machine 3 Tête pivotante sur un axe et plateau tournant sur un axe	Programmation de la transformation d'orientation. Cinématique avec **trois** axes linéaires et **deux** axes rotatifs orthogonaux. Les axes rotatifs sont parallèles à deux des trois axes linéaires. Le premier axe rotatif est déplacé par deux axes linéaires cartésiens. Il fait tourner le troisième axe linéaire avec l'outil. Le second axe rotatif fait pivoter la pièce.

Transformations 5/6 axes génériques

Type de machine	Programmation de la transformation d'orientation
Transformation cinq/six axes générique Types de machine 4 tête orientable à 2 axes avec outil indexable sur lui-même et plateau tournant à un axe	Programmation de la transformation d'orientation. Cinématique avec **trois** axes linéaires et **trois** axes rotatifs orthogonaux. Les axes rotatifs sont parallèles à deux des trois axes linéaires. Le premier axe rotatif est déplacé par deux axes linéaires cartésiens. Il fait tourner le troisième axe linéaire avec l'outil. Le second axe rotatif fait pivoter la pièce. L'orientation de base de l'outil peut être programmée par une rotation supplémentaire sur lui-même avec l'angle de rotation THETA.

A l'appel de la "transformation générique trois, quatre et cinq/six axes", l'orientation de base de l'outil peut de plus être transmise. Les restrictions concernant les directions des axes rotatifs ne s'appliquent plus. Lorsque les axes rotatifs ne sont pas exactement perpendiculaires entre eux ou que les axes rotatifs existants ne sont pas parfaitement parallèles aux axes linéaires, la "transformation générique cinq/six axes" peut fournir de meilleurs résultats pour l'orientation de l'outil.

Transformations cinématiques TRANSMIT, TRACYL et TRAANG

Pour les fraisages à effectuer sur des tours ou sur un axe à pénétration oblique en rectification, ce sont les dispositions d'axes suivantes qui sont valables en standard, en fonction de la transformation:

TRANSMIT	Activation de la transformation polaire
usinage frontal sans démontage de la pièce	un axe rotatif un axe de pénétration dans le plan perpendiculaire à l'axe rotatif un axe longitudinal parallèle à l'axe rotatif

TRACYL	Activation de la transformation de surface latérale de cylindre
Usinage de rainures de forme quelconque sur des corps cylindriques	un axe rotatif un axe de pénétration dans le plan perpendiculaire à l'axe rotatif un axe longitudinal parallèle à l'axe rotatif

TRAANG	Activation de la transformation axe oblique
Usinage avec axe de pénétration incliné	un axe rotatif un axe de pénétration avec angle paramétrable un axe longitudinal parallèle à l'axe rotatif

Déplacement PTP cartésien

Le déplacement de la machine s'effectue en coordonnées machine et se programme avec:

TRAORI	Activation de la transformation
Déplacement point à point PTP	Déplacement à une position dans le système de coordonnées cartésiennes (SCM)
CP	Déplacement avec interpolation des axes cartésiens dans (SCB)
STAT	La position des articulations dépend de la transformation
TU	Selon quel angle faut-il déplacer les axes au plus court?

Déplacement PTP avec une transformation générique cinq/six axes

Le déplacement de la machine s'effectue dans les coordonnées machine et l'orientation de l'outil peut aussi bien être programmé avec des positions d'axes rotatifs qu'avec des vecteurs Euler ou des angles RPY indépendants de la cinématique ou encore des vecteurs de direction.

Une interpolation d'axe rotatif, une interpolation vectorielle avec une interpolation circulaire de grand rayon ou une interpolation du vecteur d'orientation sur une enveloppe de cône sont possibles.

Exemple Transformation trois à cinq axes pour une tête de fraisage à Cardan

La machine-outil comporte au moins 5 axes, dont

- Trois par translation pour des mouvements rectilignes déplaçant le point de fonctionnement à n'importe quelle position dans l'espace de travail.
- Deux axes de pivotement rotatifs disposés selon un angle configurable (généralement 45 degrés) permettant à l'outil d'adopter des orientations dans l'espace qui se limitent à une demi-sphère pour une disposition à 45 degrés.

3.9.1.2 Aperçu de la transformation d'orientation TRAORI

Types de programmation possibles en relation avec TRAORI

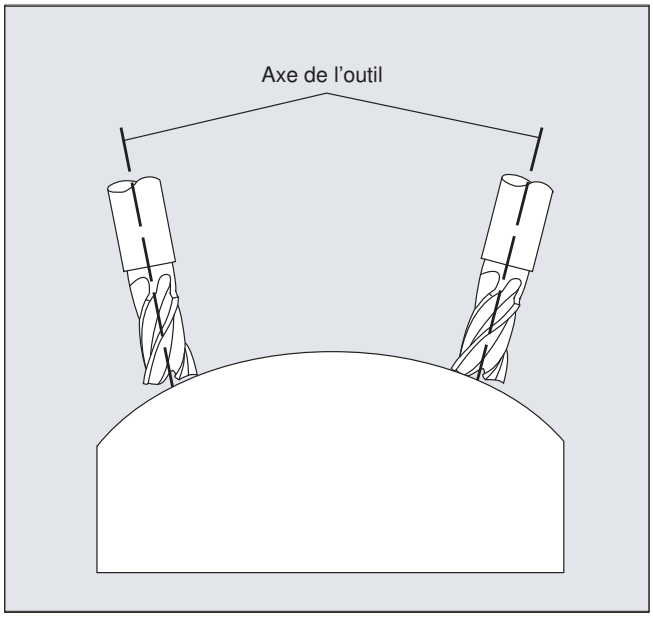
Type de machine	Programmation en cas de transformation TRAORI active
Types de machine 1, 2 ou 3 tête orientable à deux axes ou plateau tournant à deux axes ou une combinaison d'une tête orientable et d'un plateau tournant sur axe chacun.	La succession d'axes des axes d'orientation et la direction d'orientation de l'outil est configurable soit **par rapport à la machine** par le biais des paramètres machine en fonction de la cinématique de machine soit **par rapport à la pièce** avec orientation programmable indépendamment de la cinématique de machine. Les sens de rotation des axes d'orientation dans le système de référence se programment avec: - Système de référence ORIMKS = Système de coordonnées machine - Système de référence ORIWKS = Système de coordonnées pièce La configuration de base est ORIWKS Programmation des axes d'orientation avec: A, B, C positions d'axe machine directes A2, B2, C2 programmation d'angles d'axes virtuels avec - ORIEULER par angle d'Euler (par défaut) - ORIRPY par angle RPY - ORIVIRT1 par axes virtuels d'orientation 1e définition - ORIVIRT2 par axes virtuels d'orientation 2e définition Avec distinction du type d'interpolation : **Interpolation linéaire** - ORIAXES d'axes d'orientation ou d'axes machine **Interpolation circulaire de grand rayon** (interpolation du vecteur d'orientation) - ORIVECT d'axes d'orientation Programmation des axes d'orientation par indication A3, B3, C3 des composantes de vecteur (normale à la direction/surface) Programmation de l'orientation résultante de l'outil A4, B4, C4 du vecteur de la normale à la surface au début du bloc A5, B5, C5 du vecteur de la normale à la surface à la fin du bloc LEAD angle d'avance pour l'orientation de l'outil TILT angle latéral pour l'orientation de l'outil

Type de machine	Programmation en cas de transformation TRAORI active
	Interpolation du vecteur d'orientation sur une enveloppe de cône Changements d'orientation sur une enveloppe de cône placée **librement dans l'espace** Enveloppe de cône par interpolation : - ORIPLANE dans le plan (interpolation circulaire de grand rayon) - ORICONCW sur une enveloppe de cône dans le sens horaire - ORICONCCW sur une enveloppe de cône dans le sens antihoraire A6, B6, C6 vecteur de direction (axe de rotation du cône) -OICONIO interpolation sur une enveloppe de cône avec : A7, B7, C7 Vecteurs intermédiaires (orientation de départ et orientation finale) ou - ORICONTO sur une enveloppe de cône transition tangentielle Changements d'orientation par rapport à **une trajectoire** avec - ORICURVE Indication du déplacement de deux points de contact par le biais de PO[XH]=(xe, x2, x3, x4, x5) Polynômes d'orientation jusqu'au 5e degré PO[YH]=(ye, y2, y3, y4, y5) Polynômes d'orientation jusqu'au 5e degré PO[ZH]=(ze, z2, z3, z4, z5) Polynômes d'orientation jusqu'au 5e degré - ORIPATHS lissage du tracé d'orientation avec A8, B8, C8 La phase de réorientation de l'outil correspond: à la direction et à la longueur de parcours de l'outil lors du mouvement de retrait
Types de machine 1 et 3 D'autres types de machine avec rotation supplémentaire de l'outil sur lui-même exigent un troisième axe rotatif Transformation d'orientation, par exemple une transformation générique six axes. Rotations du vecteur d'orientation.	Programmation **des rotations** de l'orientation de l'outil avec LEAD Angle d'avance Angles par rapport au vecteur normal à la surface PO[PHI]=(.. , ..)] Programmation d'un polynôme jusqu'au 5e degré TILT Angle latéral Rotation par rapport à la tangente à la trajectoire (direction Z) PO[PSI] Programmation d'un polynôme jusqu'au 5e degré THETA Angle de rotation (rotation par rapport à la direction en Z de l'outil) THETA= Valeur atteinte en fin de bloc THETA=AC(...) Passage bloc par bloc à l'introduction des cotes en valeurs absolues THETA=IC(...) Passage bloc par bloc à l'introduction des cotes relatives THETA=Θ_e Interpoler l'angle programmé G90/G91 PO[THT]=(..) Programmation d'un polynôme jusqu'au 5e degré Programmation du vecteur de rotation - ORIROTA Rotation absolue - ORIROTR Vecteur de rotation relatif - ORIROTT Vecteur de rotation tangentiel
Orientation relative à la trajectoire pour des modifications d'orientation par rapport à la trajectoire ou rotation du vecteur de rotation tangentiellement à la trajectoire	Changements d'orientation **par rapport à la trajectoire** avec - ORIPATH Orientation de l'outil par rapport à la trajectoire - ORIPATHS additionnellement en cas de coude dans le tracé d'orientation Programmation du vecteur de rotation - ORIROTC Vecteur de rotation tangentiel, Rotation par rapport à la tangente à la trajectoire

3.9.2 Transformation trois, quatre et cinq axes (TRAORI)

3.9.2.1 Corrélations générales de tête de fraisage de type cardan

Pour réunir des conditions de coupe optimales lors de l'usinage de surfaces courbes, il est nécessaire de pouvoir modifier l'angle d'inclinaison de l'outil.

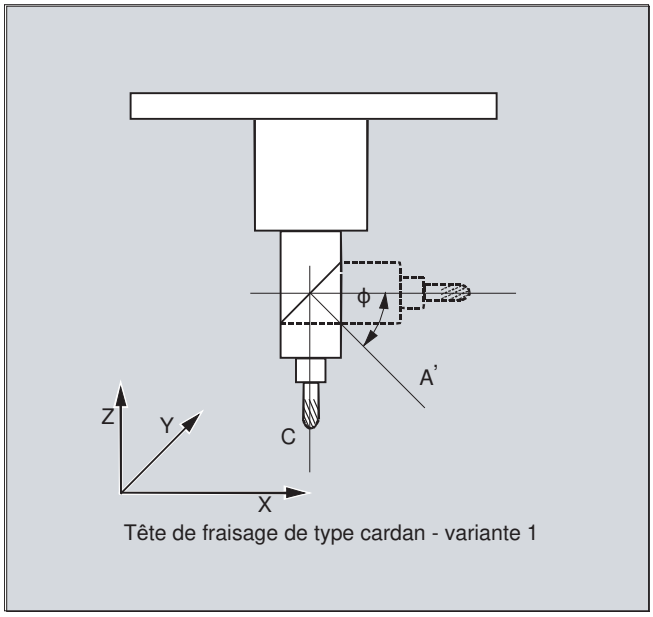


La configuration correspondante de la machine est rangée dans les paramètres machine relatifs aux axes.

Transformation 5 axes

Tête de fraisage de type cardan

Trois axes linéaires (X, Y, Z) et deux axes d'orientation (C, A) déterminent l'angle d'inclinaison et le point d'attaque de l'outil. L'un des deux axes d'orientation - dans l'exemple A' - est incliné, souvent à 45°.

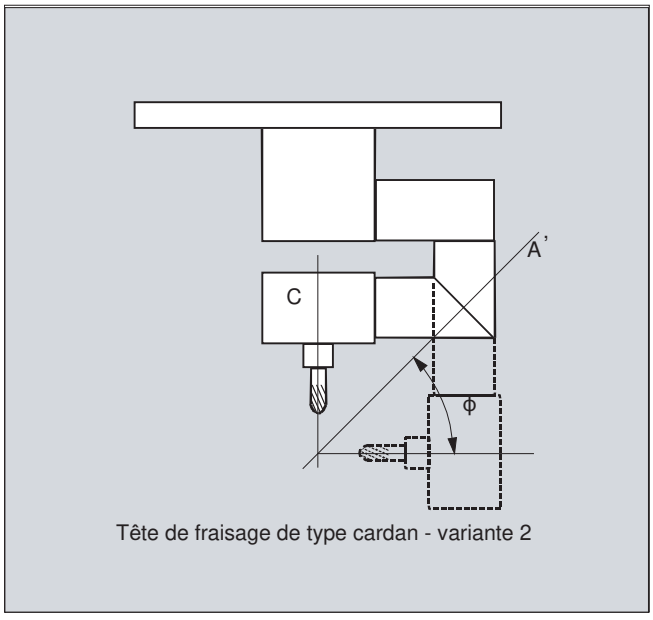


Tête de fraisage de type cardan - variante 1

Les exemples montrés ici illustrent les dispositions illustrées avec la tête d'outil à Cardan de la cinématique de machine CA!

Constructeur de machines

La succession d'axes des axes d'orientation et la direction d'orientation de l'outil peuvent se régler en fonction de la cinématique de machine par le biais de paramètres machine.



Tête de fraisage de type cardan - variante 2

Sur cet exemple, A' se trouve sous l'angle φ par rapport à l'axe X

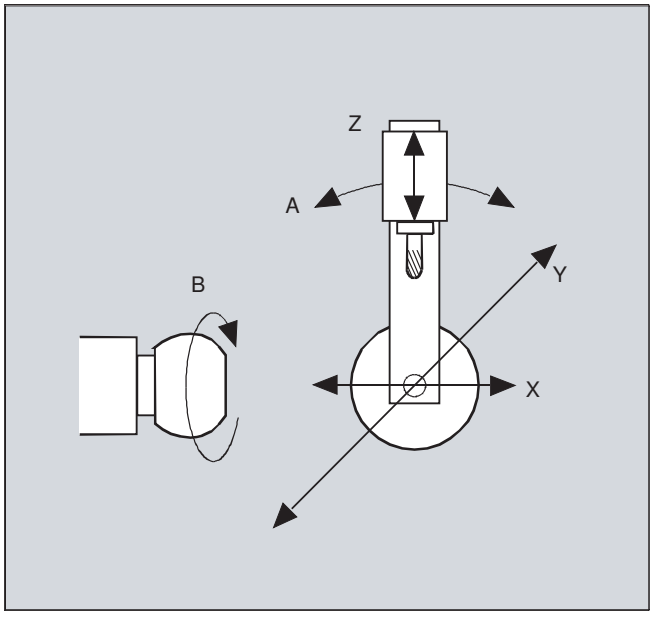
D'une manière générale, les corrélations possibles sont les suivantes:

A' se trouve sous l'angle φ par rapport à	l'axe X
B' se trouve sous l'angle φ par rapport à	l'axe Y
C' se trouve sous l'angle φ par rapport à	l'axe Z

L'angle φ peut être configuré dans la plage 0° à +89° par le biais des paramètres machine.

Avec axe linéaire pivotant

Il s'agit d'une configuration avec une pièce mobile et un outil mobile. La cinématique se compose de trois axes linéaires (X, Y, Z) et de deux axes rotatifs orthogonaux. Le premier axe rotatif se déplace sur un chariot à deux axes linéaires croisés. L'outil est parallèle au troisième axe linéaire. Le second axe rotatif fait pivoter la pièce. Le troisième axe linéaire (axe de pivotement) se situe dans le plan du chariot croisé.



La succession d'axes des axes rotatifs et la direction d'orientation de l'outil sont peuvent se régler en fonction de la cinématique de machine par le biais de paramètres machine.

Les corrélations possibles sont les suivantes :

Axes :	**Configuration axiale :**
1er axe rotatif	A A B B C C
2e axe rotatif	B C A C A B
Axe linéaire pivoté	Z Y Z X Y X

Des informations complémentaires sur les successions d'axes configurables pour la direction d'orientation de l'outil figurent dans la description fonctionnelle Transformations.

3.9.2.2 Transformation trois, quatre et cinq axes (TRAORI)

L'utilisateur peut configurer deux ou trois axes linéaires et deux axes rotatifs. Les transformations partent du principe que les axes rotatifs sont perpendiculaires au plan d'orientation.

L'orientation de l'outil est possible uniquement dans le plan perpendiculaire à l'axe rotatif. Les transformations sont plus particulièrement destinées aux machines avec outil et pièce mobiles.

La configuration et la programmation des transformations 3 axes et 4 axes sont analogues aux transformations 5 axes.

Syntaxe

```
TRAORI(<n>)
TRAORI(<n>,<X>,<Y>,<Z>,<A>,<B>)
TRAFOOF
```

Signification

`TRAORI` :	Activation de la première transformation d'orientation convenue	
`TRAORI(<n>)` :	Activation de la n-ième transformation d'orientation convenue	
`<n>` :	Numéro de la transformation	
	Valeur :	1 ou 2
	Exemple : TRAORI(1) active la transformation d'orientation 1	
`<X>,<Y>,<Z>` :	Composante du vecteur d'orientation dans laquelle est dirigé l'outil.	
`<A>,<B>` :	Offset programmable pour les axes rotatifs	
`TRAFOOF` :	Désactiver la transformation	

Orientation de l'outil

En fonction de l'orientation de l'outil sélectionnée, il convient de régler le plan de travail actif (G17, G18, G19) défini dans le programme CN, de façon que la correction de longueur d'outil agisse dans la direction de l'orientation de l'outil.

Remarque

Dès que la transformation est activée, les indications de position (X, Y, Z) se réfèrent toujours à la pointe de l'outil. Toute modification des positions des axes rotatifs impliqués dans la transformation entraîne des déplacements compensatoires des autres axes machine, ce qui permet que la position de la pointe de l'outil reste inchangée.

La transformation d'orientation est toujours dirigée de la pointe de l'outil vers le porte-outil.

Offset pour les axes d'orientation

L'activation de la transformation d'orientation permet de programmer directement un offset supplémentaire pour les axes d'orientation.

Les paramètres peuvent être omis si l'ordre correct est respecté lors de la programmation.

Exemple :

```
TRAORI(, , , ,A,B)  ; lorsqu'un seul offset doit être renseigné
```

Comme alternative à la programmation directe, l'offset supplémentaire pour les axes d'orientation peut aussi être repris automatiquement à partir du décalage d'origine présentement actif. La prise en charge est définie à l'aide des paramètres machine.

Exemples

```
TRAORI(1,0,0,1)          ; L'orientation de base de l'outil pointe dans la direction
                         Z
TRAORI(1,0,1,0)          ; L'orientation de base de l'outil pointe dans la direction
                         Z
TRAORI(1,0,1,1)          ; L'orientation de base de l'outil pointe dans le direction
                         Y/Z (correspond à la position -45°)
```

3.9.2.3 Variantes de programmation d'orientation et position initiale (ORIRESET)

Programmation d'orientation de l'outil avec TRAORI

En association avec une transformation d'orientation programmable TRAORI, des positions d'axe ou des axes virtuels avec angles ou composants vectoriels peuvent être également programmés par des descripteurs d'axes rotatifs A.., B..., C... en plus des axes linéaires X, Y, Z. Pour les axes d'orientation et les axes machine, différents types d'interpolations sont possibles. Indépendamment des polynômes d'orientation PO[angle] et des polynômes d'axe PO[axe] activés actuellement, on peut programmer plusieurs types de polynômes différents tels que G1, G2, G3, CIP ou POLY.

La modification d'orientation de l'outil peut se programmer également par le biais de vecteurs d'orientation. Ce faisant, l'orientation finale de chaque bloc peut s'effectuer soit par programmation directe du vecteur, soir par programmation des positions des axes rotatifs.

Variantes de programmation d'orientation pour transformations trois à cinq axes

Les variantes de programmation de l'orientation suivantes s'excluent mutuellement.

`A, B, C`	Indication directe de positions d'axe rotatif
`A2, B2, C2`	Programmation des angles des axes virtuels par angle d'Euler ou angle RPY
`A3, B3, C3`	Indication des composantes du vecteur
`LEAD, TILT`	Indication de l'angle d'avance et de l'angle latéral par rapport à la trajectoire et à la surface
`A4, B4, C4` `A5, B5, C5`	Vecteur normal à la surface en début et en fin de bloc
`A6, B6, C6` `A7, B7, C7`	Interpolation du vecteur d'orientation sur une enveloppe de cône
`A8, B8, C8`	Réorientation de l'outil, direction et longueur de parcours du mouvement de retrait

Accostage de la position de base de l'orientation de l'outil (ORIRESET)

`ORIRESET(...)` permet de déplacer les axes d'orientation de la cinématique de machine active, de manière linéaire et synchrone, à partir de la position actuelle vers la position initiale programmée. Si, pour un axe, aucune position initiale n'est programmée, alors la position du paramètre machine correspondant $MC_TRAFO5_ROT_AX_OFFSET_1/2 est utilisée.

Des frames des axes rotatifs actifs ne sont pas pris en compte.

Exemples de cinématique de machine CA (nom d'axe de canal C, A)

Instructions	Description
`ORIRESET(90, 45)`	`Axe C : 90°` `Axe A : 45°`
`ORIRESET(, 30)`	`Axe C : $MC_TRAFO5_ROT_AX_OFFSET_1/2[0],` `Axe A : 30°`
`ORIRESET( )`	`Axe C : $MC_TRAFO5_ROT_AX_OFFSET_1/2[0],` `Axe A : $MC_TRAFO5_ROT_AX_OFFSET_1/2[1]`

Exemples de cinématique de machine CAC (nom d'axe de canal C, A, B)

Instructions	Description

`ORIRESET(90, 45, 90)`	`Axe C : 90°`
	`Axe A : 45°`
	`Axe B : 90°`

`ORIRESET( )`	`Axe C : $MC_TRAFO5_ROT_AX_OFFSET_1/2[0],`
	`Axe A : $MC_TRAFO5_ROT_AX_OFFSET_1/2[1]`
	`Achse B : $MC_TRAFO5_ROT_AX_OFFSET_1/2[2]`

Remarque

La position initiale de l'orientation de l'outil ne doit être accostée avec `ORIRESET(...)` que si la transformation d'orientation`TRAORI(...)` est activée.

Programmation des rotations LEAD, TILT et THETA

Angle d'avance LEAD et angle latéral TILT

Dans le cas de la transformation trois à cinq axes, les rotations de l'orientation de l'outil se programment avec l'angle d'avance LEAD et l'angle latéral TILT.

Angle de rotation THETA

Lors d'une transformation **avec un troisième axe rotatif**, la rotation de l'outil sur lui-même peut être programmée avec l'angle de rotation THETA, aussi bien pour l'orientation avec des composantes de vecteur qu'avec la programmation des angles LEAD, TILT.

3.9.2.4 Programmation de l'orientation de l'outil (A..., B..., C..., LEAD, TILT)

On dispose des possibilités suivantes pour programmer l'orientation de l'outil:

1. Programmation directe du déplacement des axes rotatifs. Le changement d'orientation s'effectue toujours dans le système de coordonnées de base ou dans le système de coordonnées machine. Les axes d'orientation sont déplacés sous forme d'axes synchrones.
2. Programmation dans les angles Euler ou RPY selon la définition d'angle via `A2, B2, C2`.
3. Programmation du vecteur de direction par le biais de `A3, B3, C3`. Le vecteur de direction est dirigé de la pointe de l'outil vers le porte-outil.
4. Programmation du vecteur normal à la surface en début de bloc avec `A4, B4, C4` et en fin de bloc avec `A5, B5, C5` (fraisage avant).
5. Programmation par le biais de l'angle d'avance `LEAD` et de l'angle latéral `TILT`
6. Programmation de l'axe rotatif du cône comme vecteur normé par le biais de `A6, B6, C6` ou de l'orientation intermédiaire sur l'enveloppe du cône par le biais de `A7, B7, C7`, voir chapitre "Programmation de l'orientation le long d'une enveloppe de cône (ORIPLANE, ORICONxx)".
7. Programmation de réorientation, de direction et longueur de parcours de l'outil pendant le mouvement de relèvement via `A8, B8, C8`,
   voir chapitre "Lissage du tracé d'orientation (ORIPATHS A8=, B8=, C8=)"

Remarque

Dans tous les cas, la programmation de l'orientation est autorisée uniquement si une transformation d'orientation est activée.

Avantage : ces programmes sont transposables sur n'importe quelle cinématique de machine.

Définition de l'orientation de l'outil via une instruction G

Remarque

Constructeur de machine

Le paramètre machine permet de commuter entre angle Euler ou RPY. Avec des paramètres machine adaptés, une commutation aussi bien dépendante qu'indépendante de l'instruction G active du groupe 50 est possible. Les paramétrages suivants sont possibles :

1. Lorsque les deux paramètres machine pour la définition des axes d'orientation et la définition de l'angle d'orientation sont remis à zéro par l'instruction G :
   Les angles programmés avec `A2, B2, C2` sont interprétés **en fonction du paramètre machine** Définition de l'angle de la programmation d'orientation soit comme des angles Euler soit comme des angles RPY.
2. Lorsque le paramètre machine pour la définition des axes d'orientation est mis à un par l'instruction G, la commutation a lieu
   en fonction de l'instruction G active du groupe 50 :
   Les angles programmés avec `A2, B2, C2` sont interprétés selon l'une des instructions G actives `ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2, ORIAXPOS` et `ORIPY2`. Les valeurs programmées avec les axes d'orientation sont également interprétées comme angles d'orientation selon l'instruction G active du groupe 50.
3. Lorsque le paramètre machine pour la définition des angles d'orientation au moyen de l'instruction G est mis à un et que le paramètre machine pour la définition des axes d'orientation au moyen de l'instruction G est mis à zéro, la commutation s'effectue **indépendamment** de l'instruction G active dans le groupe 50 :
   les angles programmés avec `A2, B2, C2` sont interprétés en fonction de l'une des instructions G `ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2 ORIAXPOS` et `ORIPY2` actives. Les valeurs programmées avec les axes d'orientation sont toujours interprétées comme positions d'axe rotatif, indépendamment de l'instruction G active du groupe 50.

Syntaxe

Positions des axes rotatifs

```
G1 X<valeur> Y<valeur> Z<valeur> A<valeur> B<valeur> C<valeur>
```

Angles d'Euler

```
G1 X<valeur> Y<valeur> Z<valeur> A2<valeur> B2<valeur> C2<valeur>
```

Vecteur de direction

```
G1 X<valeur> Y<valeur> Z<valeur> A3<valeur> B3<valeur> C3<valeur>
```

Vecteur normal à la surface en début de bloc

```
G1 X<valeur> Y<valeur> Z<valeur> A4<valeur> B4<valeur> C4<valeur>
```

Vecteur normal à la surface en fin de bloc

```
G1 X<valeur> Y<valeur> Z<valeur> A5<valeur> B5<valeur> C5<valeur>
```

Angle d'avance

`LEAD=<valeur>`

Angle latéral

`TILT=<valeur>`

Signification

`G1` :	interpolation linéaire
`X, Y, Z` :	Positions des axes linéaires
`A, B, C` :	Positions des axes rotatifs
`A2=, B2=, C2=`:	Programmation d'angles (angles d'Euler ou RPY)
`A3=, B3=, C3=`:	Vecteur de direction en coordonnées X, Y et Z du SCP
`A4=, B4=, C4=`:	Vecteur normal à la surface en début de bloc, en coordonnées X, Y et Z du SCP
`A5=, B5=, C5=`:	Vecteur normal à la surface en fin de bloc, en coordonnées X, Y et Z du SCP
`LEAD=` :	Angle d'avance [1)]
`TILT=` :	Angle latéral [1)]
1) L'interprétation des indications d'angle dépend des réglages dans PM21094 $MC_ORIPATH_MODE	

Informations complémentaires

En règle générale, les programmes 5 axes sont générés par des systèmes de CAO/FAO et ne sont pas entrés au pied de la machine. C'est pourquoi les explications suivantes s'adressent principalement aux programmeurs de postprocesseurs.

Les instructions suivantes sont disponibles pour la programmation de l'orientation :

Instruction	Signification
`ORIEULER` :	Angle d'Euler avec ordre de rotation ZX'Z''
`ORIRPY` :	Angle RPY avec ordre de rotation XY'Z''
`ORIRPY2` :	Angle RPY avec ordre de rotation ZY'X''
`ORIVIRT1` :	Axes virtuels d'orientation avec positions ordre de rotation librement définissables par : PM21120 $MC_ORIAX_TURN_TAB_1
`ORIVIRT2` :	Axes virtuels d'orientation avec positions ordre de rotation librement définissables par : PM21130 $MC_ORIAX_TURN_TAB_2
`ORIAXPOS` :	Axes virtuels d'orientation avec positions d'axe rotatif

Remarque

Le constructeur de machine peut définir différentes variantes via des paramètres machine. Veuillez tenir compte des indications du constructeur de machine.

Programmation en angles d'Euler ORIEULER, ordre de rotation Z X' Z"

Les valeurs programmées avec A2, B2, C2 lors de la programmation de l'orientation ORIEULER sont interprétées comme angles d'Euler (en degrés).

Le nouveau vecteur d'orientation est obtenu à partir des trois rotations suivantes du vecteur d'orientation d'origine

1. avec l'axe rotatif A2 autour de l'axe de coordonnées Z
2. avec l'axe rotatif B2 autour du nouvel axe de coordonnées X'
3. avec l'axe rotatif C2 autour du nouvel axe de coordonnées Z"

Dans ce cas, la valeur de C2 (rotation autour du nouvel axe Z) est sans signification et n'est pas à programmer.

Programmation en angles RPY ORIRPY, ordre de rotation X Y' Z"

Les valeurs programmées avec A2, B2, C2 lors de la programmation de l'orientation ORIRPY sont interprétées comme angles RPY (en degrés) avec l'ordre de rotation X Y' Z".

Remarque

Contrairement à la programmation d'angles avec ORIEULER, les trois valeurs agissent avec ORIRPY sur le vecteur d'orientation.

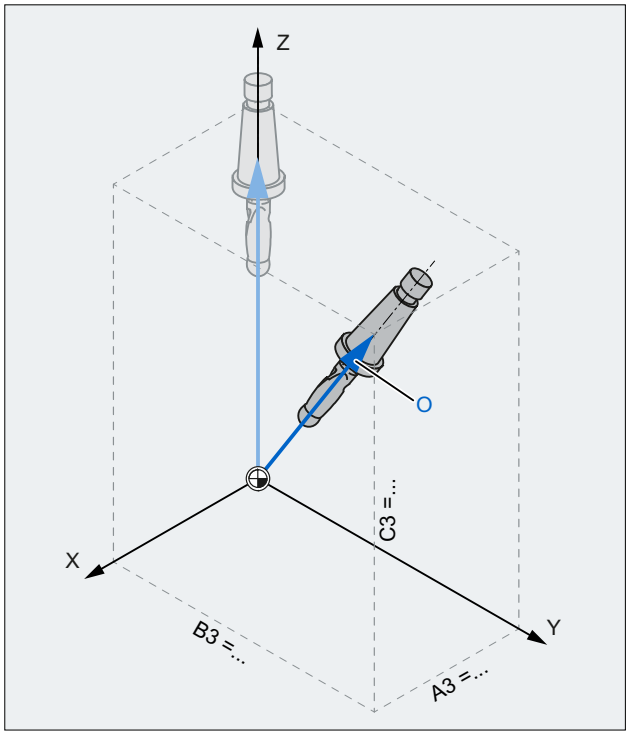
Le nouveau vecteur d'orientation est obtenu à partir des trois rotations suivantes du vecteur d'orientation d'origine

1. avec l'axe rotatif A2 autour de l'axe de coordonnées X
2. avec l'axe rotatif B2 autour du nouvel axe de coordonnées Y'
3. avec l'axe rotatif C2 autour du nouvel axe de coordonnées Z"

Programmation du vecteur de direction

Les composantes du vecteur de direction sont programmées avec A3, B3, C3 . Le vecteur est dirigé vers le porte-outil ; la longueur du vecteur est sans importance.

La commande attribue la valeur zéro aux composantes non programmées du vecteur.



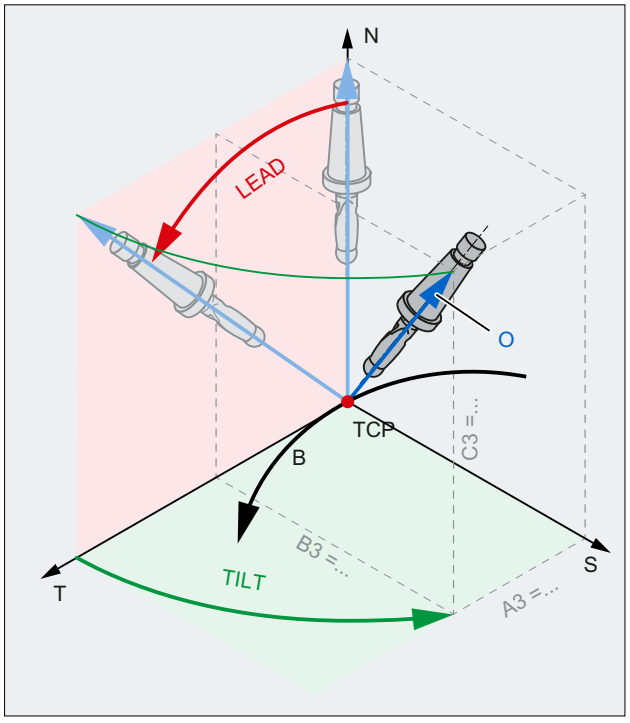
X, Y, Z	Axes de coordonnées du SCP
A3, B3, C3	Composantes du vecteur de direction
O	Vecteur d'orientation

Figure 3-3 Programmation du vecteur de direction

Programmation de l'orientation de l'outil avec LEAD et TILT

L'orientation résultante de l'outil est déterminée à partir de :

- de la tangente à la trajectoire
- Vecteur normal à la surface
  En début de bloc `A4, B4, C4` et en fin de bloc `A5, B5, C5`
- Angle d'avance `LEAD`
  Angle dans le plan défini par la tangente à la trajectoire et le vecteur normal à la surface
- Angle latéral `TILT` en fin de bloc
  Angle par rapport au vecteur normal à la surface, dans le plan perpendiculaire à la tangente à la trajectoire



T	Tangente à la trajectoire
S	Perpendiculaire à la tangente à la trajectoire
N	Normale à la surface
B	Trajectoire
TCP	Tool Center Point :
O	Vecteur d'orientation

Figure 3-4 Programmation de LEAD TILT

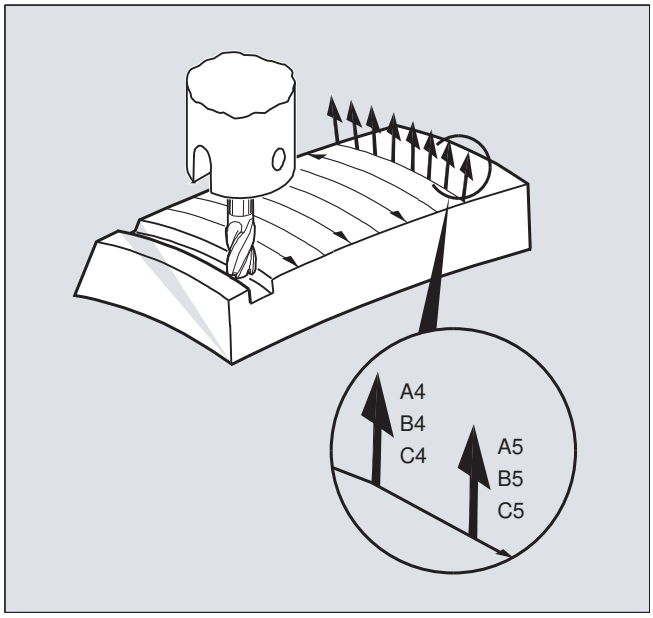
Remarque

Comportement aux angles rentrants pour CO 3D

L'orientation programmée de l'outil est toujours effectuée en fin de bloc, même si le bloc est tronqué au niveau d'un angle rentrant.

3.9.2.5 Fraisage en bout (A4, B4, C4, A5, B5, C5)

Le fraisage en bout sert à l'usinage des surfaces à courbure quelconque.



Pour ce type de fraisage 3D, vous avez besoin de la description des différentes trajectoires 3D qui balayent la surface de la pièce, ligne après ligne.

Les calculs sont généralement réalisés à l'aide d'un système de FAO, en tenant compte de la forme et des dimensions de l'outil. Les blocs CN générés sont ensuite lus dans la commande par l'intermédiaire de postprocesseurs.

Programmation de la courbure de la trajectoire

Description des surfaces

La description de la courbure de la trajectoire s'effectue par le biais de vecteurs normaux à la surface ayant les composantes suivantes :

`A4, B4, C4` : vecteur de début en début de bloc

`A5, B5, C5` : vecteur de fin en fin de bloc

Si un bloc ne contient que le vecteur de début, le vecteur normal à la surface restera constant pendant tout le bloc. Si un bloc ne contient que le vecteur de fin, une interpolation sur le grand cercle est effectuée de la valeur de fin du bloc précédent jusqu'à la valeur de fin programmée.

Si le vecteur de début et le vecteur de fin sont programmés, une interpolation sur le grand cercle est également effectuée entre les deux directions. Ceci permet d'obtenir des trajectoires lisses.

Dans la configuration de base, les vecteurs normaux à la surface sont orientés dans la direction Z, indépendamment du plan G17 à G19 activé.

La longueur des vecteurs est sans importance.

La valeur zéro est attribuée par défaut aux composantes de vecteur non programmées.

Lorsque ORIWKS est activé (voir "Référence des axes d'orientation (ORIWKS, ORIMKS) (Page 691)"), les vecteurs normaux à la surface se réfèrent au frame actif et subissent également une rotation en cas de rotation du frame.

Constructeur de machine

Le vecteur normal à la surface doit être perpendiculaire à la tangente à la trajectoire avec une zone de tolérance qui est réglable par le biais d'un paramètre machine, sinon une alarme est émise.

3.9.2.6 Référence des axes d'orientation (ORIWKS, ORIMKS)

Lors de la programmation de l'orientation dans le système de coordonnées pièces par le biais

- des angles d'Euler ou RPY ou
- du vecteur d'orientation,

ORIMKS/ORIWKS permettent de régler la trajectoire du mouvement de rotation.

Remarque

Constructeur de machine

Le mode d'interpolation pour l'orientation est défini par le paramètre machine :

PM21104 $MC_ORI_IPO_WITH_G_CODE

= FALSE : Les instructions G ORIWKS et ORIMKS servent de référence

= TRUE : Les instructions G du 51e groupe (ORIAXES, ORIVECT, ORIPLANE, ...) servent de référence

Syntaxe

```
ORIMKS=...
ORIWKS=...
```

Signification

ORIMKS :	Rotation dans le système de coordonnées machine
ORIWKS :	Rotation dans le système de coordonnées pièce

Remarque

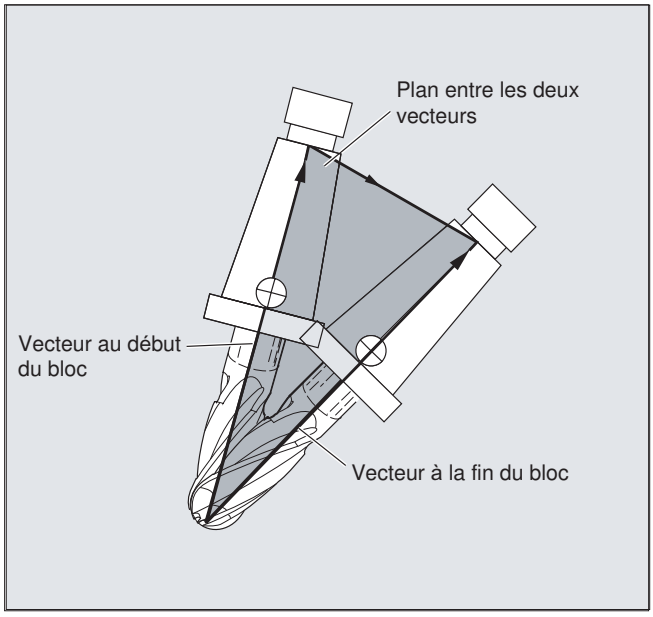
ORIWKS correspond au réglage de base. Si on ne connaît pas d'avance la machine sur laquelle doit fonctionner un programme à 5 axes, il faut par principe choisir ORIWKS. Les déplacements effectivement exécutés par la machine dépendent de sa cinématique.

ORIMKS permet de programmer les mouvements réels de la machine, p. ex. pour éviter des collisions avec des équipements, etc.

Informations complémentaires

Avec ORIMKS, le déplacement effectué par l'outil **dépend** de la cinématique de la machine. Dès qu'il y a un changement d'orientation avec pointe de l'outil fixe, il y a exécution d'une interpolation linéaire entre les positions des axes rotatifs.

Avec ORIWKS, le déplacement de l'outil est **indépendant** de la cinématique de la machine. Dès qu'il y a un changement d'orientation avec pointe de l'outil fixe, l'outil se déplace dans le plan défini par le vecteur de début et le vecteur de fin.



Positions singulières

Remarque

ORIWKS

Les mouvements d'orientation au voisinage de la position singulière d'une machine à 5 axes nécessitent des déplacements importants de la part des axes machine. (C'est le cas par exemple sur une tête pivotante et tournante, avec C comme axe de rotation et A comme axe de pivotement, toutes les positions avec A = 0 étant des positions singulières.)

Constructeur de machines

Pour ne pas surcharger les axes machine, le pilotage de la vitesse réduit fortement la vitesse tangentielle au voisinage des points singuliers.

Les paramètres machine

$MC_TRAFO5_NON_POLE_LIMIT

$MC_TRAFO5_POLE_LIMIT

permettent de paramétrer la transformation de façon que les déplacements d'orientation au voisinage du pôle passent effectivement par le pôle, ce qui permet un usinage plus rapide.

Les positions singulières sont traitées uniquement avec le MD $MC_TRAFO5_POLE_LIMIT .

Informations complémentaires : Description fonctionnelle Transformations

3.9.2.7 Programmation des axes d'orientation (ORIAXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2)

La fonction "Axes d'orientation" décrit l'orientation de l'outil dans l'espace et s'obtient en programmant les offsets pour les axes rotatifs. On peut obtenir un troisième degré de liberté par une rotation supplémentaire de l'outil autour de son propre axe. Cette orientation de l'outil s'effectue au choix dans l'espace par le biais d'un troisième axe rotatif et requiert la transformation six axes. La rotation de l'outil autour de son propre axe se fixe en fonction du type d'interpolation des vecteurs de rotation avec l'angle de rotation THETA (voir "Rotations de l'orientation de l'outil (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Page 702)".

Les axes d'orientation se programment par le biais des descripteurs d'axes A2, B2, C2.

Syntaxe

```
N... ORIAXES/ORIVECT            ; Interpolation linéaire ou circulaire de
                                  grand rayon
N... G1 X Y Z A B C

N... ORIPLANE                   ; Interpolation d'orientation du plan

N... ORIEULER/ORIRPY/ORIRPY2    ; Angle d'orientation d'Euler / RPY
N... G1 X Y Z A2= B2= C2=       ; Programmation des angles des axes virtuels

N... ORIVIRT1/ORIVIRT2          ; Axes virtuels d'orientation, déf. 1/2
N... G1 X Y Z A3= B3= C3=       ; Programmation du vecteur de direction
```

Remarque

Pour les modifications d'orientation le long d'une enveloppe de cône située dans l'espace, on peut programmer d'autres offsets d'axes rotatifs des axes d'orientation (voir "Programmation de l'orientation le long d'une enveloppe de cône (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Page 695)".

Signification

`ORIAXES` :	Interpolation linéaire des axes machine ou d'orientation
`ORIVECT` :	Interpolation circulaire de grand rayon (identique avec ORIPLANE)
`ORIMKS` : `ORIWKS` :	Rotation dans le système de coordonnées machine Rotation dans le système de coordonnées pièce Pour une description, voir "Référence des axes d'orientation (ORIWKS, ORIMKS) (Page 691)".
`A= B= C=` :	Programmation de la position des axes machine
`ORIEULER` :	Programmation d'orientation par le biais d'angles d'Euler

`ORIRPY` :	Programmation de l'orientation par le biais d'angles RPY L'ordre de rotation est XYZ, avec : • A2 est l'angle de rotation autour de X • B2 est l'angle de rotation autour de Y • C2 est l'angle de rotation autour de Z
`ORIRPY2` :	Programmation de l'orientation par le biais d'angles RPY L'ordre de rotation est ZYX, avec : • A2 est l'angle de rotation autour de Z • B2 est l'angle de rotation autour de Y • C2 est l'angle de rotation autour de X
`A2= B2= C2=` :	Programmation des angles des axes virtuels
`ORIVIRT1/ORIVIRT2` :	Programmation de l'orientation par le biais d'axes d'orientation virtuels Définition 1 : Selon MD21120 $MC_ORIAX_TURN_TAB_1 Définition 2 : Selon MD21130 $MC_ORIAX_TURN_TAB_2
`A3= B3= C3=` :	Programmation du vecteur de direction de l'axe d'orientation

Informations complémentaires

Constructeur de machines

Le paramètre machine MD21102 $MC_ORI_DEF_WITH_G_CODE permet de fixer le mode de définition des angles A2, B2, C2 programmés :

La définition s'effectue d'après le paramètre machine MD21100 $MC_ORIENTATION_IS_EULER (par défaut) ou d'après le groupe de codes G 50 (ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2).

Le paramètre machine MD21104 $MC_ORI_IPO_WITH_G_CODE permet de définir le type d'interpolation : ORIWKS/ORIMKS ou ORIAXES/ORIVECT.

Mode de fonctionnement Jog

Dans ce mode de fonctionnement, les angles d'orientation sont toujours interpolés de façon linéaire. Lors d'un positionnement relatif et continu à l'aide des touches de déplacement, un seul axe d'orientation peut être déplacé. Avec les manivelles, le déplacement simultané des axes d'orientation est possible.

En déplacement manuel d'axes d'orientation, le commutateur de correction de l'avance spécifique au canal est actif ou, le cas échéant, le commutateur de correction du rapide en déplacement en rapide.

Les paramètres machine suivants permettent de définir séparément les vitesses :

MD21160 $MC_JOG_VELO_RAPID_GEO

MD21165 $MC_JOG_VELO_GEO

MD21150 $MC_JOG_VELO_RAPID_ORI

MD21155 $MC_JOG_VELO_ORI

Remarque

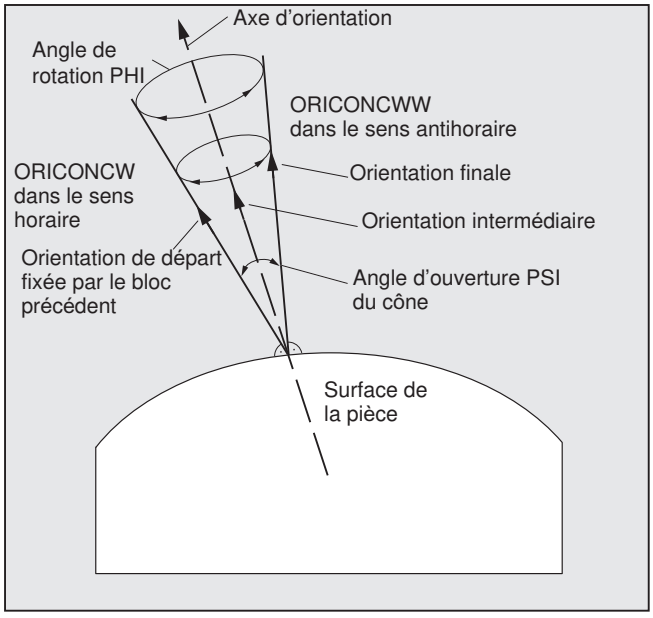
SINUMERIK 840D sl avec "Paquet de transformation pour manipulation"

En mode JOG, la fonction "Déplacement manuel cartésien" permet de régler la translation d'axes géométriques dans les systèmes de référence SCM, SCP et SCO, séparément les uns des autres.

Informations complémentaires : Description fonctionnelle Transformations

3.9.2.8 Programmation de l'orientation le long d'une enveloppe de cône (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO)

L'orientation étendue permet d'effectuer des changements d'orientation le long d'une enveloppe de cône située dans l'espace. L'interpolation du vecteur d'orientation sur une enveloppe de cône s'effectue avec les instructions modales ORICONxx. Pour l'interpolation dans un plan, on peut programmer l'orientation finale avec ORIPLANE. D'une manière générale, l'orientation de départ est déterminée par les blocs précédents.



Programmation

L'orientation finale se détermine soit par l'indication de la programmation d'angles en angles d'Euler ou RPY avec A2, B2, C2 ou par la programmation des positions des axes rotatifs avec a, B, C. Pour les axes d'orientation le long de l'enveloppe de cône, d'autres indications de programmation sont nécessaires:

- Axe rotatif du cône sous forme de vecteur avec `A6, B6, C6`
- Angle au centre `PSI` avec les descripteurs `NUT`
- Orientation intermédiaire dans l'enveloppe de cône avec `A7, B7, C7`

Remarque

Programmation du vecteur de direction A6, B6, C6 pour l'axe rotatif du cône

La programmation d'une orientation finale n'est pas absolument nécessaire. Si aucune orientation finale n'est indiquée, une enveloppe de cône complète est interpolée avec 360 degrés.

Programmation de l'angle d'ouverture du cône avec NUT=angle

L'indication d'une orientation finale est impérativement nécessaire.

De cette manière, on ne peut pas interpoler une enveloppe de cône complète avec 360 degrés.

Programmation de l'orientation intermédiaire A7, B7, C7 dans l'enveloppe de cône

L'indication d'une orientation finale est impérativement nécessaire. Le changement d'orientation et le sens de rotation sont déterminés de manière univoque par les trois vecteurs Orientation de départ, Orientation finale et Orientation intermédiaire. Ce faisant, les trois vecteurs doivent être différents. Si l'orientation intermédiaire programmée est parallèle à l'orientation de départ ou à l'orientation finale, une interpolation circulaire de grand rayon de l'orientation est effectuée dans le plan défini entre le vecteur de départ et le vecteur final.

Interpolation d'orientation étendue sur une enveloppe de cône

```
N... ORICONCW ou ORICONCCW
N... A6= B6= C6= A3= B3= C3=
ou
N... ORICONTO
N... G1 X Y Z A6= B6= C6=
ou
N... ORICONIO
N... G1 X Y Z A7= B7= C7=
N... PO[PHI]=(a2, a3, A4, A5)
N... PO[PSI]=(b2, b3, b4, b5)
```

Interpolation sur une enveloppe de cône avec

vecteur de direction dans le sens horaire/ antihoraire du cône et orientation finale
ou

transition tangentielle et
indication de l'orientation finale

ou

indication de l'orientation finale et d'une orientation intermédiaire dans l'enveloppe de cône avec

polynômes pour l'angle de rotation et polynômes pour l'angle d'ouverture

Paramètres

`ORIPLANE` :	Interpolation dans le plan (interpolation circulaire de grand rayon)
`ORICONCW` :	Interpolation sur une enveloppe de cône dans le sens horaire
`ORICONCCW` :	Interpolation sur une enveloppe de cône dans le sens anti-horaire
`ORICONTO` :	Interpolation sur une enveloppe de cône avec transition tangentielle
`A6= B6= C6=` :	Programmation de l'axe rotatif du cône (vecteur normé)
`NUT=angle` :	Angle au centre du cône en degrés
`NUT=+179` :	Angle de déplacement inférieur ou égal à 180 degrés
`NUT=-181` :	Angle de déplacement supérieur ou égal à 180 degrés
`ORICONIO` :	Interpolation sur une enveloppe de cône
`A7= B7= C7=` :	Orientation intermédiaire (programmation comme vecteur normé)
`PHI` :	Angle de rotation de l'orientation autour de l'axe d'orientation du cône
`PSI` :	Angle d'ouverture du cône
Polynômes possibles PO[PHI]=(a2, a3, A4, A5) PO[PSI]=(b2, b3, b4, b5)	En dehors des différents angles, on peut aussi programmer des polynômes au maximum du 5e degré

Exemple : Différents changements d'orientation

Code de programme	**Commentaire**
…	
`N10 G1 X0 Y0 F5000`	
`N20 TRAORI(1)`	`; Transformation d'orientation activée.`
`N30 ORIVECT`	`; Interpoler l'orientation de l'outil comme vecteur.`
…	`; Orientation de l'outil dans le plan.`
`N40 ORIPLANE`	`; Sélectionner l'interpolation circulaire de grand rayon.`
`N50 A3=0 B3=0 C3=1`	
`N60 A3=0 B3=1 C3=1`	`; Orientation dans le plan Y/Z tournée de 45 degrés, l'orientation (0,1/√2,1/√2) est atteinte en fin de bloc.`
…	
`N70 ORICONCW`	`; Programmation de l'orientation sur l'enveloppe de cône :`
`N80 A6=0 B6=0 C6=1 A3=0 B3=0 C3=1`	`; Le vecteur d'orientation est interpolé dans le sens horaire sur une enveloppe de cône avec la direction (0,0,1) jusqu'à l'orientation (1/√2,0,1/√2), l'angle de rotation étant de 270 degrés.`
`N90 A6=0 B6=0 C6=1`	`; L'orientation de l'outil effectue un tour complet sur la même enveloppe de cône.`

Informations complémentaires

Si l'on veut décrire des changements d'orientation sur une enveloppe de cône située à un endroit quelconque dans l'espace, il faut que le vecteur autour duquel la rotation de l'orientation de l'outil doit s'effectuer, soit connu. Il faut en outre que l'orientation de départ et l'orientation finale soient prédéfinies. L'orientation de départ découle du bloc précédent et l'orientation finale doit être soit programmée, soit déterminée par d'autres conditions.

La programmation dans le plan ORIPLANE correspond à ORIVECT

La programmation de l'interpolation circulaire de grand rayon conjugée aux polynômes trigonométriques correspond à l'interpolation linéaire et polynômiale de contours. L'orientation de l'outil est interpolée dans un plan défini par l'orientation de départ et l'orientation finale. Si l'on programme aussi des polynômes, il se peut que le vecteur d'orientation bascule également hors du plan.

Programmation de cercles dans un plan G2/G3, CIP et CT

L'orientation étendue correspond à l'interpolation de cercles dans un plan. En ce qui concerne les modes de programmation de cercles avec indication du centre ou du rayon comme G2/G3, cercle par point intermédiaire CIP et cercles tangentiels CT, voir le chapitre "Interpolation circulaire (Page 193)".

Programmation de l'orientation

Interpolation du vecteur d'orientation sur une enveloppe de cône ORICONxx

Pour l'interpolation d'orientations sur une enveloppe de cône, on peut sélectionner quatre types d'interpolations différents dans le groupe G 51 :

1. Interpolation sur une enveloppe de cône dans le sens horaire ORICONCW avec indication de l'orientation finale et de la direction du cône ou de son angle d'ouverture. Le vecteur de direction se programme avec les descripteurs A6, B6, C6 et l'angle d'ouverture du cône avec le descripteur NUT= plage de valeurs dans l'intervalle 0 à 180 degrés.
2. Interpolation sur une enveloppe de cône dans le sens antihoraire ORICONCWW avec indication de l'orientation finale et de la direction du cône ou de son angle d'ouverture. Le vecteur de direction se programme avec les descripteurs A6, B6, C6 et l'angle d'ouverture du cône avec le descripteur NUT= plage de valeurs dans l'intervalle 0 à 180 degrés.
3. Interpolation sur une enveloppe de cône ORICONIO avec indication de l'orientation finale et d'une orientation intermédiaire, programmée avec les descripteurs A7, B7, C7.
4. Interpolation sur une enveloppe de cône ORICONTO avec transition tangentielle et indication de l'orientation finale. Le vecteur de direction se programme avec les descripteurs A6, B6, C6.

3.9.2.9 Définition de l'orientation de deux points de contact (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=)

Programmation du changement d'orientation par la deuxième courbe spatiale ORICURVE

Une autre possibilité pour programmer les changements d'orientation est de programmer avec ORICURVE, outre la pointe de l'outil le long d'une courbe spatiale, également le mouvement d'un deuxième point de contact de l'outil. On peut ainsi, comme lors de la programmation du vecteur de l'outil lui-même, définir de manière univoque les changements d'orientation de l'outil.

Constructeur de machines

Toujours respecter les indications du constructeur de machines en ce qui concerne les descripteurs d'axes réglables par le biais de paramètres machine pour la programmation de la deuxième trajectoire d'orientation de l'outil.

Programmation

Dans le cas de ce type d'interpolation, on peut, pour les deux courbes spatiales, programmer des points avec G1 ou des polynômes avec POLY. Les cercles et les développantes ne sont pas admis. De plus, une interpolation de type spline peut être activée avec BSPLINE et avec la fonction "groupement de blocs spline courts".

Des informations complémentaires sur la fonction "Groupement de blocs spline courts" figurent dans la description fonctionnelle Fonctions de base.

Les autres types de spline ASPLINE et CSPLINE ainsi que l'activation d'un compresseur avec COMP... ne sont pas admis.

Le déplacement des deux points de contact de l'outil peut être prédéfini lors de la programmation des polynômes d'orientation pour des coordonnées du 5e degré maximum.

Interpolation d'orientation étendue avec courbe spatiale supplémentaire et polynômes pour coordonnées

```
N... ORICURVE                          ; Indication du déplacement du deuxième point de contact de l'outil et polynômes supplémentaires des coordonnées correspondantes
N... PO[XH]=(xe, x2, x3, x4, x5)
N... PO[YH]=(ye, y2, y3, y4, y5)
N... PO[ZH]=(ze, z2, z3, z4, z5)
```

Paramètre

`ORICURVE`	Interpolation de l'orientation avec indication du déplacement de deux points de contact de l'outil.
`XH YH ZH`	Descripteur des coordonnées du deuxième point de contact de l'outil du contour supplémentaire en tant que courbe spatiale

`PO[XH]=(xe, x2, x3, x4, x5)`
`PO[YH]=(ye, y2, y3, y4, y5)`
`PO[ZH]=(ze, z2, z3, z4, z5)`
	Polynômes possibles
	Les courbes spatiales peuvent se programmer avec les points de fin correspondants, mais en plus aussi avec des polynômes.

`xe, ye, ze`	Points finals de la courbe spatiale
`xi, yi, zi`	Coefficients des polynômes du 5e degré maximum

Remarque

Descripteurs XH YH ZH pour la programmation d'une 2e trajectoire d'orientation

On doit choisir les descripteurs de manière à ce qu'il ne se produise pas de conflits avec les autres descripteurs des axes linéaires

Axes X Y Z

et les axes rotatifs tels que

A2 B2 C2, les angles d'Euler ou les angles RPY

A3 B3 C3, les vecteurs de direction

A4 B4 C4 ou A5 B5 C5, les vecteurs normaux à la surface

A6 B6 C6, les vecteurs de rotation ou les coordonnées de points intermédiaires A7 B7 C7

ou d'autres paramètres d'interpolation.

3.9.3 Polynômes d'orientation (PO[angles], PO[coordonnée])

Indépendamment de l'interpolation polynomiale du groupe G 1 actuellement activée, on peut programmer deux types de polynômes d'orientation jusqu'au 5e degré maximum dans le cas d'une transformation trois à cinq axes.

1. Polynômes pour **angles:** Angle d'avance LEAD, angle latéral TILT
   par rapport au plan défini par l'orientation de départ et l'orientation finale.
2. Polynômes pour **coordonnées:** XH, YH, ZH de la deuxième courbe spatiale pour l'orientation de l'outil d'un point de référence sur l'outil.

Dans le cas d'une transformation six axes, on peut, en plus de l'orientation de l'outil, programmer la rotation du vecteur de rotation THT avec des polynômes du 5e degré maximum pour les rotations de l'outil lui-même.

Syntaxe

Polynômes d'orientation du type 1 pour **angles**

```
N... PO[PHI]=(a2, a3, A4, A5)
N... PO[PSI]=(b2, b3, b4, b5)
```

Transformation trois à cinq axes

Polynômes d'orientation du type 2 pour **coordonnées**

```
N... PO[XH]=(xe, x2, x3, x4, x5)
N... PO[YH]=(ye, y2, y3, y4, y5)
N... PO[ZH]=(ze, z2, z3, z4, z5)
```

Descripteur pour les coordonnées de la deuxième trajectoire d'orientation pour l'orientation de l'outil

On peut en outre, dans les deux cas, programmer un polynôme pour la **rotation** dans le cas d'une transformation six axes avec

`N... PO[THT]=(c2, c3, c4, c5)`	Interpolation par rapport à la trajectoire de la rotation
ou	
`N... PO[THT]=(d2, d3, d4, d5)`	Interpolation absolue, relative et tangentielle du vecteur d'orientation par rapport au changement d'orientation

. Ceci est possible si la transformation supporte un vecteur de rotation avec un décalage interpolable et programmable par le biais de l'angle de rotation THETA.

Signification

`PO[PHI]`	Angle dans le plan entre l'orientation de départ et l'orientation finale
`PO[PSI]`	Angle décrivant le basculement de l'orientation depuis le plan entre l'orientation de départ et l'orientation finale.
`PO[THT]`	Angle de rotation décrit par la rotation du vecteur de rotation d'une des instructions G du groupe 54 programmés avec THETA
`PHI`	Angle d'avance LEAD
`PSI`	Angle latéral TILT
`THETA`	Rotation par rapport à la direction de l'outil en Z
`PO[XH]`	Coordonnée X du point de référence sur l'outil
`PO[YH]`	Coordonnée Y du point de référence sur l'outil
`PO[ZH]`	Coordonnée Z du point de référence sur l'outil

Informations complémentaires

On ne peut pas programmer les polynômes d'orientation

- si les interpolations spline ASPLINE, BSPLINE, CSPLINE sont activées.
  Les polynômes du type 1 pour les angles d'orientation sont possibles pour chaque type d'interpolation sauf Spline c.-à-d. pour l'interpolation linéaire avec vitesse rapide G00 ou avec avance G01
  pour l'interpolation polynômiale avec POLY et
  pour l'interpolation circulaire ou à développante avec G02, G03, CIP, CT, INVCW et INCCCW

  Les polynômes du type 2 pour les coordonnées d'orientation ne sont en revanche possibles que si
  l'interpolation linéaire avec vitesse rapide G00 ou avec avance G01 ou
  l'interpolation polynômiale avec POLY est activée.
- si l'orientation est interpolée au moyen de l'interpolation d'axes ORIAXES. Dans ce cas, on peut programmer directement des polynômes avec PO[A] et PO[B] pour les axes d'orientation A et B.

Polynômes d'orientation du type 1 avec ORIVECT, ORIPLANE et ORICONxx

En cas d'interpolation circulaire de grand rayon et d'interpolation sur enveloppe de cône avec ORIVECT, ORIPLANE et ORICONxx, seuls les polynômes d'orientation du type 1 sont possibles.

Polynômes d'orientation du type 2 avec ORICURVE

Si l'interpolation est activée avec la courbe spatiale supplémentaire ORICURVE, les composantes cartésiennes du vecteur d'orientation sont interpolées et seuls les polynômes d'orientation du type 2 sont possibles.

3.9.4 Rotations de l'orientation de l'outil (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA)

Si l'orientation de l'outil doit pouvoir être modifiée sur les machines avec outil mobile, chaque bloc doit être programmé avec une orientation finale. En fonction de la cinématique de la machine, le sens d'orientation des axes d'orientation ou le sens de rotation du vecteur d'orientation THETA peuvent être programmés. Différents types d'interpolation sont programmables pour ces vecteurs de rotation :

- ORIROTA : Angle de rotation rapporté à un sens de rotation en indication absolue.
- ORIROTR : Angle de rotation relatif par rapport au plan entre orientation de départ et orientation finale.
- ORIROTT : Angle de rotation relatif par rapport à la modification du vecteur d'orientation.
- ORIROTC : Angle de rotation tangentiel à la tangente à la trajectoire.

Syntaxe

A la condition expresse que le type d'interpolation ORIROTA soit actif, l'angle de rotation ou le vecteur de rotation peut être programmé selon l'un des quatre modes suivants :

1. Directement dans les positions d'axes rotatifs `A, B, C`
2. En angles d'Euler (en degrés) par le biais de `A2, B2, C2`
3. En angles RPY (en degrés) par le biais de `A2, B2, C2`
4. En vecteur de direction par le biais de `A3, B3, C3` (angle de rotation à l'aide de `THETA=<valeur>`)

Si ORIOTR ou ORIOTT est actif, l'angle de rotation ne peut être programmé que directement à l'aide de `THETA`.

Une rotation peut également être programmée seule dans un bloc, sans qu'un changement d'orientation ne se produise. ORIROTR et ORIROTT n'ont alors aucune signification. Dans ce cas, l'angle de rotation est toujours interprété par rapport à la direction absolue (ORIROTA).

`N... ORIROTA`	Définition de l'interpolation du vecteur de rotation
`N... ORIROTR`	
`N... ORIROTT`	
`N... ORIROTC`	
`N... A3= B3= C3= THETA=<valeur>`	Définir la rotation du vecteur d'orientation
`N... PO[THT]=(`d_2`, `d_3`, `d_4`, `d_5`)`	Interpolation de l'angle de rotation à l'aide d'un polynôme du cinquième degré

Signification

`ORIROTA` :	Angle de rotation rapporté à un sens de rotation en indication absolue
`ORIROTR` :	Angle de rotation relatif par rapport au plan entre orientation de départ et orientation finale
`ORIROTT` :	Angle de rotation comme vecteur de rotation tangentiel au changement d'orientation
`ORIROTC` :	Angle de rotation comme vecteur de rotation tangentiel à la tangente à la trajectoire
`THETA` :	Rotation du vecteur d'orientation
`THETA=<valeur>` :	Angle de rotation en degrés atteint en fin de bloc
`THETA=`Θe :	Angle de rotation avec angle final Θ_e du vecteur de rotation
`THETA=AC(...)` :	Passage bloc par bloc à l'introduction des cotes en valeurs absolues
`THETA=AC(...)` :	Passage bloc par bloc à l'introduction des cotes relatives
Θe :	L'angle final du vecteur de rotation est actif aussi bien en valeurs absolues avec G90 qu'en valeurs relatives avec G91 (introduction de cotes relatives).
`PO[THT]=(....)` :	Polynôme pour l'angle de rotation

Exemple : Rotations des orientations

Code de programme	**Commentaire**
`N10 TRAORI`	`; Activer la transformation d'orientation`
`N20 G1 X0 Y0 Z0 F5000`	`; Orientation de l'outil`
`N30 A3=0 B3=0 C3=1 THETA=0`	`; en direction Z avec angle de rotation 0`
`N40 A3=1 B3=0 C3=0 THETA=90`	`; en direction X et rotation de 90 degrés`
`N50 A3=0 B3=1 C3=0 PO[THT]=(180,90)`	`; Orientation`
`N60 A3=0 B3=1 C3=0 THETA=IC(-90)`	`; en direction Y et rotation sur 180 degrés`
`N70 ORIROTT`	`; demeure constant et rotation sur 90 degrés`
`N80 A3=1 B3=0 C3=0 THETA=30`	`; Angle de rotation par rapport au changement d'orientation`
	`; Vecteur de rotation formant un angle de 30 degrés par rapport au plan X-Y`

Lors de l'interpolation du bloc N40, l'angle de rotation est interpolé de manière linéaire de la valeur de départ 0 degré à la valeur finale 90 degrés. Dans le bloc N50, l'angle de rotation passe de 90 à 180 degrés conformément à la parabole θ(u) = +90u2. En N60, une rotation peut également être effectuée sans qu'il se produise de changement d'orientation.

En N80, l'orientation de l'outil tourne, passant de la direction Y à la direction X. Ce faisant, le changement d'orientation s'opère dans le plan X-Y et le vecteur de rotation forme un angle de 30 degrés par rapport à ce plan.

Informations complémentaires

ORIROTA

L'angle de rotation THETA est interpolé dans l'espace par rapport à une direction déterminée de manière absolue. Le sens de rotation par défaut est fonction des paramètres machine.

ORIROTR

L'angle de rotation THETA est interprété de manière relative par rapport au plan défini par l'orientation initiale et l'orientation finale.

ORIROTT

L'angle de rotation THETA est interprété de manière relative par rapport au changement d'orientation. PourTHETA=0, le vecteur de rotation est interpolé de manière tangentielle par rapport au changement d'orientation et ne se distingue d'ORIROTR que si au moins un polynôme pour l'"angle de basculement PSI" a été programmé pour l'orientation. Il en résulte un changement d'orientation qui ne se produit pas dans le plan. La programmation d'un angle de rotation THETA supplémentaire permet d'interpoler par exemple le vecteur de rotation de manière à ce qu'il forme toujours une valeur définie par rapport au changement d'orientation.

ORIROTC

Le vecteur de rotation est interpolé par rapport à la tangente à la trajectoire avec un décalage programmable par le biais de l'angle THETA . Pour l'angle de décalage, on peut aussi programmer un polynôme PO[THT]=(c2, c3, c4, c5) du 5e degré maximum.

3.9.5 Orientations par rapport à la trajectoire

3.9.5.1 Types d'orientation par rapport à la trajectoire

Cette fonction étendue permet d'atteindre l'orientation relative non seulement en fin de bloc, mais aussi sur tout le tracé de la trajectoire. L'orientation atteinte au cours du bloc précédent est transférée au moyen de l'interpolation circulaire de grand rayon dans l'orientation finale programmée. Par principe, il existe deux possibilités pour programmer l'orientation souhaitée par rapport à la trajectoire:

1. L'orientation de l'outil et la rotation de l'outil sont interpolées avec ORIPATH, ORPATHTS par rapport à la trajectoire.
2. Le vecteur d'orientation est programmé et interpolé de la manière habituelle. La rotation du vecteur d'orientation par rapport à la tangente à la trajectoire s'opère avec ORIROTC.

Syntaxe

Le type d'interpolation de l'orientation et de la rotation de l'outil se programme avec:

`N... ORIPATH`	Orientation par rapport à la trajectoire
`N... ORIPATHS`	Orientation par rapport à la trajectoire avec lissage du tracé d'orientation
`N... ORIROTC`	Interpolation par rapport à la trajectoire du vecteur de rotation

ORIPATHS permet de lisser un coude de l'orientation provoqué par un angle dans le tracé de la trajectoire. La direction et la longueur de parcours du mouvement de retrait se programment par le biais du vecteur avec les composants A8 = X, B8 = Y et C8 = Z.

ORIPATH/ORIPATHS permettent de programmer différentes références par rapport à la tangente à la trajectoire par le biais des trois angles

- LEAD = indication de l'angle d'avance par rapport à la trajectoire et à la surface
- TILT = indication de l'angle latéral par rapport à la trajectoire et à la surface
- THETA = angle de rotation

pour l'ensemble du tracé de la trajectoire. En plus de l'angle de rotation THETA, il est aussi possible de programmer des polynômes du 5e degré maximum avec PO[THT] = (...).

Remarque

Constructeur de machines

Tenir compte des indications du constructeur de machines. Par le biais de paramètres machine et de données de réglage configurables, il est possible de procéder à d'autres réglages en plus du type d'orientation par rapport à la trajectoire.

Informations complémentaires : Description fonctionnelle Transformations

Signification

L'interpolation des angles LEAD et TILT peut se régler différemment par le biais de paramètres machine :

- La référence de l'orientation de l'outil programmée avec LEAD et TILT est respectée au-delà de toute la durée du bloc.
- Angle d'avance LEAD : Rotation par rapport à la direction perpendiculaire à la tangente et vecteur normal TILT : Rotation de l'orientation par rapport au vecteur normal.

- Angle d'avance LEAD : Rotation par rapport à la direction perpendiculaire à la tangente et vecteur normal Angle latéral TILT : Rotation de l'orientation par rapport à la direction de la tangente à la trajectoire.
- Angle de rotation THETA: Rotation de l'outil autour de son propre axe avec un troisième axe rotatif supplémentaire en tant qu'axe d'orientation dans le cas de transformation six axes.

Remarque

Orientation relative à la trajectoire non autorisée avec OSC, OSS, OSSE, OSD, OST

L'interpolation d'orientation par rapport à la trajectoire ORIPATH ou ORIPATHS et ORIOTC ne peut pas être programmée avec le lissage du tracé d'orientation avec l'une des instructions G du groupe 34. Pour ce faire, il faut qu'OSOF soit activé.

3.9.5.2 Rotation de l'orientation de l'outil relative à la trajectoire (ORIPATH, ORIPATHS, angle de rotation)

Dans le cas d'une transformation six axes, en plus de l'orientation de l'outil d'une manière quelconque dans l'espace, ce dernier peut également tourner autour de son propre axe avec un troisième axe supplémentaire. Dans le cas d'une rotation de l'orientation de l'outil par rapport à la trajectoire avec ORIPATH ou ORIPATHS, la rotation supplémentaire peut se programmer par le biais de l'angle de rotation THETA. En guise d'alternative, les angles LEAD et TILT peuvent être programmés par le biais d'un vecteur perpendiculaire à la direction de l'outil dans le plan.

Constructeur de machine

Veuillez tenir compte des indications du constructeur de machine. L'interpolation des angles LEAD et TILT peut se régler différemment par le biais de paramètres machine.

Syntaxe

Rotation de l'orientation de l'outil et de l'outil

On active le type d'orientation de l'outil par rapport à la trajectoire avec ORIPATH ou ORIPATHS.

`N... ORIPATH`	Activer le type d'orientation par rapport à la trajectoire
`N... ORIPATHS`	Activer le type d'orientation par rapport à la trajectoire avec lissage du tracé d'orientation

Activation des trois angles possibles avec effet de rotation:

`N... LEAD=`	Angle pour l'orientation programmée par rapport au vecteur normal à la surface
`N... TILT=`	Angle pour l'orientation programmée dans le plan perpendiculaire à la tangente à la trajectoire par rapport au vecteur normal à la surface
`N... THETA=`	; Angle de rotation par rapport au changement d'orientation par rapport à la direction d'outil du troisième axe rotatif

Les valeurs des angles en fin de bloc se programment avec `LEAD=valeur`, `TILT=valeur` ou `THETA=valeur`. En plus des angles constants, on peut programmer des polynômes du 5e degré maximum pour les trois angles.

```
N... PO[PHI]=(a2, a3, A4, A5)
N... PO[PSI]=(b2, b3, b4, b5)
N... PO[THT]=(d2, d3, d4, d5)
```

Polynôme de l'angle d'avance LEAD
Polynôme de l'angle latéral TILT
Polynôme de l'angle de rotation THETA

Pendant la programmation, on peut laisser de côté les coefficients de polynôme plus élevés, égaux à zéro. Exemple `PO[PHI]=a2` donne une parabole pour l'angle d'avance `LEAD`.

Signification

Orientation de l'outil par rapport à la trajectoire

`ORIPATH` :	Orientation de l'outil par rapport à la trajectoire
`ORIPATHS` :	Orientation de l'outil rapportée à la trajectoire, une cassure dans le tracé de l'orientation est lissée
`LEAD` :	Angle par rapport au vecteur normal à la surface, dans le plan défini par la tangente à la trajectoire et le vecteur normal à la surface
`TILT` :	Rotation de l'orientation par rapport à la direction Z ou rotation par rapport à la tangente à la trajectoire
`THETA` :	Rotation par rapport à la direction de l'outil vers Z
`PO[PHI]` :	Polynôme d'orientation pour l'angle d'avance LEAD
`PO[PSI]` :	Polynôme d'orientation pour l'angle latéral TILT
`PO[THT]` :	Polynôme d'orientation pour l'angle de rotation THETA

Remarque

Angle de rotation THETA

Pour la rotation de l'outil avec un troisième axe rotatif en tant qu'axe d'orientation par rapport à lui-même, une transformation six axes est nécessaire.

3.9.5.3 Interpolation relative à la trajectoire de la rotation de l'outil (ORIROTC, THETA)

Type d'interpolation avec vecteurs de rotation

En plus de la rotation de l'outil (programmée avec ORIROTC) par rapport à la tangente à la trajectoire, on peut aussi interpoler le vecteur de rotation avec un décalage programmable par le biais de l'angle de rotation THETA. À cette occasion, on peut, avec PO[THT], programmer un polynôme du 5e degré maximum pour l'angle de décalage.

Syntaxe

`N... ORIROTC`	Appliquer la rotation de l'outil par rapport à la tangente à la trajectoire
`N... A3= B3= C3= THETA=valeur`	Définir la rotation du vecteur d'orientation
`N... A3= B3= C3= PO[THT]=(c2, c3, c4, c5)`	Interpolation de l'angle de décalage avec polynôme du 5e degré au maximum

Une rotation peut également être programmée seule dans un bloc, sans qu'un changement d'orientation ne se produise.

Signification

Interpolation (par rapport à la trajectoire) de la rotation de l'outil dans le cas d'une transformation six axes

`ORIROTC` :	Appliquer le vecteur de rotation tangentiel à la tangente à la trajectoire
`THETA=valeur` :	Angle de rotation en degrés atteint en fin de bloc
`THETA=`θe :	Angle de rotation avec angle final Θ_e du vecteur de rotation
`THETA=AC(...)` :	Passage bloc par bloc à l'introduction des cotes en valeurs absolues
`THETA=IC(...)` :	Passage bloc par bloc à l'introduction des cotes relatives
`PO[THT]=(c2, c3, c4, c5)` :	Interpolation de l'angle de décalage à l'aide d'un polynôme du 5e degré

Remarque

Interpolation du vecteur de rotation ORIROTC

Si l'on veut appliquer également la rotation de l'outil par rapport à la tangente à la trajectoire dans le sens opposé à la direction d'orientation de l'outil, ce n'est possible que dans le cas d'une transformation six axes.

Dans le cas où ORIROTC est activé

Le vecteur de rotation ORIROTA ne peut pas être programmé. Dans le cas d'une programmation, le message d'alarme 14128 "Programmation absolue de la rotation de l'outil dans le cas où ORIROTC est activé" est émis.

Direction d'orientation de l'outil dans le cas d'une transformation trois à cinq axes

La direction d'orientation de l'outil peut se programmer comme on en a l'habitude avec une transformation trois à cinq axes par le biais d'angles d'Euler ou d'angles RPY ou de vecteurs de direction. Également possibles: changements d'orientation de l'outil dans l'espace par programmation de l'interpolation circulaire de grand rayon ORIVECT, de l'interpolation linéaire des axes d'orientation ORIAXES, de toutes les interpolations sur une enveloppe de cône

ORICONxx ainsi que de l'interpolation en plus de la courbe spatiale avec deux points de contact de l'outil ORICURVE.

`G....:`	Indication du type de déplacement des axes rotatifs
`X, Y, Z:`	Indication des axes linéaires
`ORIAXES:`	Interpolation linéaire des axes machine ou d'orientation
`ORIVECT:`	Interpolation circulaire de grand rayon (identique avec ORIPLANE)
`ORIMKS:` `ORIWKS:`	Rotation dans le système de coordonnées machine Rotation dans le système de coordonnées pièce Pour une description, voir le chapitre "Rotations de l'orientation de l'outil"
`A= B= C=:`	Programmation de la position des axes machine
`ORIEULER:`	Programmation d'orientation par le biais d'angles d'Euler
`ORIRPY:`	Programmation de l'orientation par le biais d'angles RPY
`A2= B2= C2=:`	Programmation des angles des axes virtuels
`ORIVIRT1:` `ORIVIRT2:`	Programmation de l'orientation par le biais d'axes d'orientation virtuels (définition 1), selon PM $MC_ORIAX_TURN_TAB_1 (définition 2), selon PM $MC_ORIAX_TURN_TAB_2
`A3= B3= C3=:`	Programmation du vecteur de direction de l'axe d'orientation
`ORIPLANE:`	Interpolation dans le plan (interpolation circulaire de grand rayon)
`ORICONCW:`	Interpolation sur une enveloppe de cône dans le sens horaire
`ORICONCCW:`	Interpolation sur une enveloppe de cône dans le sens anti-horaire
`ORICONTO:`	Interpolation sur une enveloppe de cône avec transition tangentielle
`A6= B6= C6=:`	Programmation de l'axe rotatif du cône (vecteur normé)
`NUT=angle`	Angle au centre du cône en degrés
`NUT=+179`	Angle de déplacement inférieur ou égal à 180 degrés
`NUT=-181`	Angle de déplacement supérieur ou égal à 180 degrés
`ORICONIO:`	Interpolation sur une enveloppe de cône
`A7= B7= C7=:`	Orientation intermédiaire (programmation comme vecteur normé)
`ORICURVE` `XH YH ZH par exemple avec polynôme PO[XH]=(xe, x2, x3, x4, x5)`	Interpolation de l'orientation avec indication du déplacement de deux points de contact de l'outil. Outre le(s) point(s) final(aux) en question, des courbes spatiales supplémentaires sont programmables sous forme de polynômes.

Remarque

Si l'orientation de l'outil est interpolée par le biais des axes d'orientation alors qu'ORIAXES est activé, l'application de l'angle de rotation par rapport à la trajectoire n'est réalisée qu'en fin de bloc.

3.9.5.4 Lissage du tracé d'orientation (ORIPATHS A8=, B8=, C8=)

Dans le cas de changements d'orientation à accélération continue sur le contour, toute interruption des déplacements avec interpolation, pouvant notamment survenir à un angle du contour, est inopportune. Le coude qui en résulte dans le tracé d'orientation peut être lissé en ajoutant un bloc intermédiaire qui lui est propre. Le changement d'orientation s'effectue alors à accélération continue même si ORIPATHS est actif pendant la réorientation. Pendant cette phase, un mouvement de relèvement de l'outil peut être exécuté.

Constructeur de machines

Toujours respecter les indications du constructeur de machines en ce qui concerne les paramètres machine et les données de réglage prédéfinis le cas échéant et avec lesquels cette fonction est activée.

Par le biais des paramètres machine, il est possible de définir comment interpréter le vecteur de relèvement.

1. Dans le système de coordonnées de l'outil, la coordonnée Z est définie par la direction de l'outil.
2. Dans le système de coordonnées pièce, la coordonnée Z est définie par le plan actif.

Des informations complémentaires sur la fonction "Orientation par rapport à la trajectoire" figurent dans la description fonctionnelle Transformations.

Syntaxe

Pour des orientations d'outil continues par rapport à l'ensemble de la trajectoire, d'autres indications de programmation sont nécessaires à un angle du contour. La direction et la longueur de parcours de ce mouvement se programment par le biais du vecteur avec les composants A8 = X, B8 = Y et C8 = Z.

```
N... ORIPATHS A8=X B8=Y C8=Z
```

Signification

`ORIPATHS` :	Orientation de l'outil par rapport à la trajectoire, lissage de coude dans le tracé d'orientation.
`A8= B8= C8=` :	Composants de vecteur pour la direction et la longueur de parcours
`X, Y, Z` :	Mouvement de relèvement dans la direction de l'outil

Remarque

Programmation du vecteur de direction A8, B8, C8

Si la longueur de ce vecteur est égale à zéro, il ne se produit pas de mouvement de relèvement.

ORIPATHS

L'orientation de l'outil par rapport à la trajectoire est activée avec ORIPATHS. Dans le cas contraire, l'orientation est transférée par le biais d'une interpolation circulaire de grand rayon de l'orientation de départ vers l'orientation finale.

3.9.6 Compactage de l'orientation (COMPON, COMPCURV, COMPCAD, COMPSURF)

Les programmes CN dans lesquels une transformation d'orientation (TRAORI) est active et des orientations d'outil sont programmées (de quelque type que ce soit), peuvent être compactés sous réserve de respecter les tolérances spécifiées.

Programmation

Orientation de l'outil

Si une transformation d'orientation (TRAORI) est active, l'orientation de l'outil peut être programmée de la manière suivante (indépendante de la cinématique) sur les machines à 5 axes :

- Programmation du **vecteur** de direction avec :
  `A3=<...> B3=<...> C3=<...>`
- Programmation des **angles** d'Euler ou des **angles** RPY avec :
  `A2=<...> B2=<...> C2=<...>`

Rotation de l'outil

Sur les machines à **6 axes**, on peut programmer la rotation de l'outil en plus de son orientation.

La programmation de l'angle de rotation s'effectue avec :

`THETA=<...>`

Voir " Rotations de l'orientation de l'outil (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Page 702) ".

Remarque

Les blocs CN dans lesquels une rotation est programmée en plus ne sont compactables que si l'angle de rotation varie **linéairement**. Cela signifie qu'aucun polynôme ne doit être programmé avec PO[THT] = (...) pour l'angle de rotation.

Forme générale d'un bloc CN compactable

La forme générale d'un bloc CN compactable peut donc être la suivante :

```
N... X=<...> Y=<...> Z=<...> A3=<...> B3=<...> C3=<...> THETA=<...> F=<...>
```

ou

```
N... X=<...> Y=<...> Z=<...> A2=<...> B2=<...> C2=<...> THETA=<...> F=<...>
```

Remarque

Les valeurs de position peuvent être indiquées directement (par exemple X90) ou indirectement par le biais d'affectations de paramètres (par exemple X=R1*(R2+R3)).

Programmation de l'orientation de l'outil par des positions d'axes rotatifs

L'orientation de l'outil peut également être indiquée par des positions d'axes rotatifs, par exemple sous la forme suivante :

```
N... X=<...> Y=<...> Z=<...> A=<...> B=<...> C=<...> THETA=<...> F=<...>
```

Dans ce cas, le compactage est effectué de deux manières différentes selon qu'une interpolation circulaire de grand rayon est exécutée ou non. Si une interpolation circulaire de grand rayon a lieu, le changement d'orientation compacté est représenté de manière habituelle par des polynômes axiaux pour les axes rotatifs.

Précision de contour

Selon le mode de compactage réglé (MD20482 $MC_COMPRESSOR_MODE), les tolérances qui prennent effet pour les axes géométriques et les axes d'orientation lors du compactage sont soit les tolérances spécifiques aux axes (configurées dans le paramètre machine MD33100 $MA_COMPRESS_POS_TOL) ou les tolérances spécifiques aux canaux (réglables dans les données de réglage) suivantes :

SD42475 $SC_COMPRESS_CONTUR_TOL (écart maximal par rapport au contour)

SD42476 $SC_COMPRESS_ORI_TOL (écart maximal par rapport à l'angle d'orientation de l'outil)

SD42477 $SC_COMPRESS_ORI_ROT_TOL (écart maximal par rapport à l'angle de rotation de l'outil) (disponible uniquement pour les machines à 6 axes)

Activation/désactivation

Les instructions G modales COMPON, COMPCURV, COMPCAD et COMPSURF activent les fonctions compacteur.

La fonction compacteur se termine par COMPOF.

Voir " Activer/désactiver le compactage de bloc CN (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF) (Page 618) ".

Remarque

Le déplacement d'orientation est uniquement compacté lorsque l'interpolation circulaire de grand diamètre est activée (c.-à-d. que l'orientation de l'outil se modifie dans le plan défini par l'orientation initiale et l'orientation finale).

Une interpolation circulaire de grand diamètre est exécutée dans les conditions suivantes :

- MD21104 $MC_ORI_IPO_WITH_G_CODE = 0,
  ORIWKS est actif et
  l'orientation est programmée au moyen de vecteurs (avec A3, B3, C3 et A2, B2, C2).
- MD21104 $MC_ORI_IPO_WITH_G_CODE = 1 et
  ORIVECT ou ORIPLANE est actif.
  L'orientation de l'outil peut être programmée sous forme de vecteur de direction ou de positions d'axes rotatifs. Si l'une des instructions G ORICONxx et ORICURVE est active ou si des polynômes sont programmés pour les angles d'orientation (PO[PHI] et PO[PSI]), il n'y a pas d'interpolation circulaire de grand rayon.

Exemple

Dans cet exemple de programmation, un cercle, qui est obtenu par une approximation polygonale, est compacté. L'orientation de l'outil se déplace de façon synchrone sur une surface latérale de cône. Bien que les changements d'orientation successivement programmés soient discontinus, la fonction compacteur génère un tracé lisse de l'orientation.

Programmation	**Commentaire**
DEF INT NOMBRE=60	
DEF REAL RAYON=20	
DEF INT COMPTEUR	
DEF REAL ANGLE	
N10 G1 X0 Y0 F5000 G64	
$SC_COMPRESS_CONTUR_TOL=0.05	; Écart maximal par rapport au contour = 0,05 mm
$SC_COMPRESS_ORI_TOL=5	; Écart maximal par rapport à l'orientation = 5 degrés
TRAORI	
COMPCURV	; Le déplacement a lieu sur un cercle obtenu par une approximation polygonale. L'orientation se déplace sur un cône autour de l'axe Z avec un angle au centre de 45 degrés.
N100 X0 Y0 A3=0 B3=-1 C3=1	
N110 FOR COUNTER=0 TO NOMBRE	
N120 ANGLE=360*COUNTER/NOMBRE	
N130 X=RAYON*cos(ANGLE) Y=RAYON*sin(ANGLE)	
A3=sin(ANGLE) B3=-cos(ANGLE) C3=1	
N140 ENDFOR	

3.9.7 Activation/désactivation du lissage du tracé d'orientation (ORISON, ORISOF)

Le "Lissage du tracé d'orientation" est activé/désactivé dans le programme pièce par les instructions du groupe G 61. Les instructions sont des fonctions à effet modal.

Conditions requises

- Système avec transformation 5/6 axes.
- La fonction de compresseur COMPCAD est active.

Syntaxe

```
ORISON
...
ORISOF
```

Signification

ORISON :	Activation du lissage du tracé d'orientation
ORISOF :	Désactivation du lissage du tracé d'orientation

Exemple

Code de programme	**Commentaire**
...	
TRAORI()	; Activation de la transformation d'orientation.
COMPCAD	; Activation de la fonction de compresseur COMPCAD.
ORISON	; Activation du lissage d'orientation.
$SC_ORISON_TOL=1.0	; Écart maximum par rapport à l'angle d'orientation de l'outil = 1,0 degré.
G91	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
X10 A3=1 B3=0 C3=1	
X10 A3=-1 B3=0 C3=1	
...	
ORISOF	; Désactivation du lissage d'orientation.
...	

Un pivotement de l'orientation de 90 degrés est réalisé dans le plan XZ, de -45 à +45 degrés. En raison du lissage de son tracé, l'orientation n'atteint plus les valeurs angulaires maximales de -45 ou +45 degrés.

3.9.8 Transformation cinématique

3.9.8.1 Activation de la transformation de face frontale (TRANSMIT)

La transformation de face frontale (TRANSMIT) est activée dans le programme pièce ou dans une action synchrone par l'instruction TRANSMIT.

Syntaxe

```
TRANSMIT
TRANSMIT(<n>)
```

Signification

`TRANSMIT` :	Activer TRANSMIT avec le premier bloc de données TRANSMIT
`TRANSMIT(n)` :	Activation de TRANSMIT avec le <n>-ième bloc de données TRANSMIT

Remarque

Une transformation TRANSMIT active dans le canal est désactivée par :

- Désactivation de la transformation : `TRAFOOF`
- Activation d'une autre transformation : p. ex. `TRACYL`, `TRAANG`, `TRAORI`

3.9.8.2 Activer la transformation de surfaces cylindriques (TRACYL)

La transformation de surface cylindrique (TRACYL) est activée dans le programme pièce ou dans une action synchrone par l'instruction `TRACYL`.

Syntaxe

```
TRACYL(<d>)
TRACYL(<d>,<n>)
TRACYL(<d>,<n>,<k>)
```

Signification

`TRACYL(<d>)` :	Activation de TRACYL avec le premier bloc de données TRACYL et le diamètre d'usinage <d>	
`TRACYL (<d>,<n>)` :	Activation de TRACYL avec le <n>-ième bloc de données TRACYL et le diamètre d'usinage <d>	
<d> :	Diamètre de référence ou d'usinage La valeur doit être supérieure à 1.	
<n> :	Numéro de bloc de données TRACYL (facultatif)	
	Plage de valeurs :	1, 2
<k> :	Le paramètre <k> est seulement pertinent pour le type de transformation 514	
	k = 0 :	sans correction des flancs de rainure
	k = 1 :	avec correction des flancs de rainure
	Si le paramètre n'est pas spécifié, la position de base paramétrée s'applique : $MC_TRACYL_DEFAULT_MODE_<n> avec <n> = numéro de bloc de données TRACYL	

Remarque

Une transformation TRACYL active dans le canal est désactivée par :

- Désactivation de la transformation : `TRAFOOF`
- Activation d'une autre transformation : p. ex. `TRAANG`, `TRANSMIT`, `TRAORI`

Exemple

Code de programme	Commentaire
...	
N40 TRACYL(40.)	; Activation de TRACYL avec le premier bloc de données TRACYL et le diamètre d'usinage 40 mm.
...	

Informations complémentaires

Structure du programme

Un programme pièce pour le fraisage d'une rainure avec la transformation TRACYL 513 (TRACYL avec correction du flanc de rainure) comprend généralement les étapes suivantes :

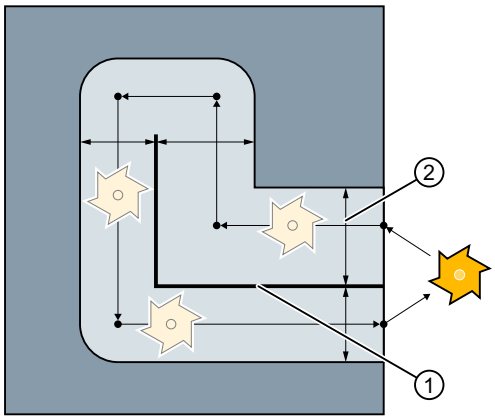
1. Sélection de l'outil.
2. Sélection de TRACYL.
3. Sélection du décalage du système de coordonnées approprié (FRAME).
4. Positionnement.
5. Programmation d'OFFN.
6. Sélection de la CRO (correction du rayon d'outil).
7. Bloc d'accostage (accostage de la CRO et accostage du flanc de la rainure).
8. Contour correspondant à l'axe de symétrie de la rainure.
9. Désactivation de la CRO.
10. Bloc de retrait (retrait de la CRO et éloignement du flanc de rainure).
11. Positionnement.
12. TRAFOOF.
13. Reprise du décalage du système de coordonnées initial (FRAME).

Offset sur contour (OFFN)

Pour fraiser des rainures avec la transformation TRACYL 513, programmer dans le programme pièce l'axe de symétrie de la rainure et **la demi-largeur** à l'aide de l'adresse OFFN.

Afin d'éviter toute détérioration du flanc de rainure, OFFN n'est activé que lorsque la correction du rayon d'outil a été sélectionnée.

Il est possible de modifier OFFN à l'intérieur du programme pièce. Ainsi, vous pouvez décentrer la ligne médiane de la rainure :



① OFFN
② Trajectoire programmée

Remarque

OFFN doit être au moins égal au rayon d'outil, afin d'empêcher toute détérioration du flanc de rainure opposé.

Remarque

OFFN avec TRACYL a un comportement différent de OFFN sans TRACYL. Il est recommandé de remettre à zéro OFFN après TRAFOOF, du fait que OFFN est également pris en compte sans TRACYL lorsque la correction de rayon d'outil CRO est activée.

IMPORTANT
L'effet d'OFFN dépend du type de transformation Avec la transformation TRACYL 513 (TRACYL avec correction du flanc de rainure), programmer la demi-largeur de la rainure pour OFFN. Par contre, avec la transformation TRACYL 512 (TRACYL sans correction du flanc de rainure), la valeur d'OFFN a pour effet de produire une surépaisseur qui s'ajoute à la correction de rayon d'outil (CRO).

Correction du rayon d'outil (CRO)

Avec la transformation TRACYL 513, la correction du rayon d'outil n'est pas calculée par rapport au flanc de rainure mais par rapport à l'axe de symétrie programmé de la rainure. Pour que l'outil se déplace à gauche du flanc de rainure, il convient de programmer l'instruction G42 à la place de G41 ou d'indiquer la valeur d'OFFN avec un signe négatif.

Diamètre de l'outil

Avec TRACYL et un outil dont le diamètre est inférieur à la largeur de la rainure, la géométrie de flanc de rainure générée n'est pas la même qu'avec un outil dont le diamètre est égal à la largeur

de la rainure. Pour améliorer la précision, il est conseillé de choisir un diamètre d'outil à peine inférieur à la largeur de la rainure.

Utilisation des axes

Remarque

Les axes suivants ne peuvent pas être utilisés comme axes de positionnement ou d'oscillation :

- l'axe géométrique dans le sens de la circonférence de l'enveloppe du cylindre (axe Y)
- l'axe linéaire supplémentaire en cas de correction des flancs de rainure (axe Z)

3.9.8.3 Activation de la transformation à angle oblique avec angle programmable (TRAANG)

La transformation à angle oblique avec angle programmable est activée dans le programme pièce ou dans une action synchrone par l'instruction `TRAANG`.

Limitation pour les chaînes cinématiques

Lors d'un paramétrage avec les paramètres machine, l'angle α de l'axe incliné est défini par rapport à l'axe de coordonnées correspondant dans un système de coordonnées cartésiennes à l'aide des paramètres machine. L'angle α peut être modifié lors de l'appel via TRAANG(<α>). Une modification de l'angle entraîne une modification du modèle cinématique, de sorte que d'autres applications peuvent également être concernées. Par conséquent, une indication de l'angle est autorisée uniquement lorsque l'angle est identique à celui résultant de la chaîne cinématique.

Remarque

Définition de l'angle

L'angle est défini uniquement lorsque l'axe incliné résulte de la rotation autour d'un axe de coordonnées. Il y a donc seulement un axe géométrique qui n'est pas parallèle à l'un des axes de coordonnées et se trouve dans le plan principal.

Syntaxe

```
TRAANG
TRAANG()
TRAANG(, <n>)
TRAANG(<α>)
TRAANG(<α>,<n>)
```

Signification

`TRAANG` : `TRAANG()` :	Activation de TRAANG avec le premier bloc de données TRAANG et le dernier angle <α> valide
`TRAANG(,<n>)` :	Activation de TRAANG avec le <n>-ième bloc de données TRAANG et le dernier angle <α> valide

TRAANG(<α>) :	Activation de TRAANG avec le premier bloc de données TRAANG et l'angle <α>	
TRAANG(<α>,<n>) :	Activation de TRAANG avec le <n>-ième bloc de données TRAANG et l'angle <α>	
<α> :	Angle de l'axe incliné (facultatif)	
	Plage de valeurs :	-90° < α < + 90°
	Si aucun angle n'est spécifié, la position de base paramétrée dans le paramètre machine s'applique : PM2xxxx $MC_TRAANG_ANGLE_<n>	
<n> :	Numéro de bloc de données TRAANG (facultatif)	
	Plage de valeurs :	1, 2

Remarque

Une transformation à angle oblique TRAANG active dans le canal est désactivée par :

- Désactivation de la transformation : TRAFOOF
- Activation d'une autre transformation : p. ex. TRACYL, TRANSMIT, TRAORI

Exemple

Code de programme	Commentaire
N20 TRAANG(45)	; Activation de TRAANG avec le premier bloc de données TRAANG et l'angle 45°

3.9.8.4 Plongée oblique sur les rectifieuses (G5, G7)

Les instructions G G7 et G5 permettent de simplifier la programmation pour la plongée oblique sur les rectifieuses avec transformation "axe incliné (TRAANG)", de manière que seul l'axe incliné se déplace lors de la plongée.

Pour la programmation, seule la position finale souhaitée du mouvement de plongée se trouve dans X et Z. La position de départ associée est calculée et accostée par la CN pour G7, en partant de la position actuelle de l'axe X, de la position finale programmée et de l'angle α de l'axe incliné.

La position de départ résulte du point d'intersection des deux droites :

- Droite parallèle à l'axe Z à la distance de la position actuelle de l'axe X
- Droite parallèle à l'axe incliné passant par la position finale programmée

L'axe incliné se déplacera à la position finale programmée avec la fonction G5 suivante.

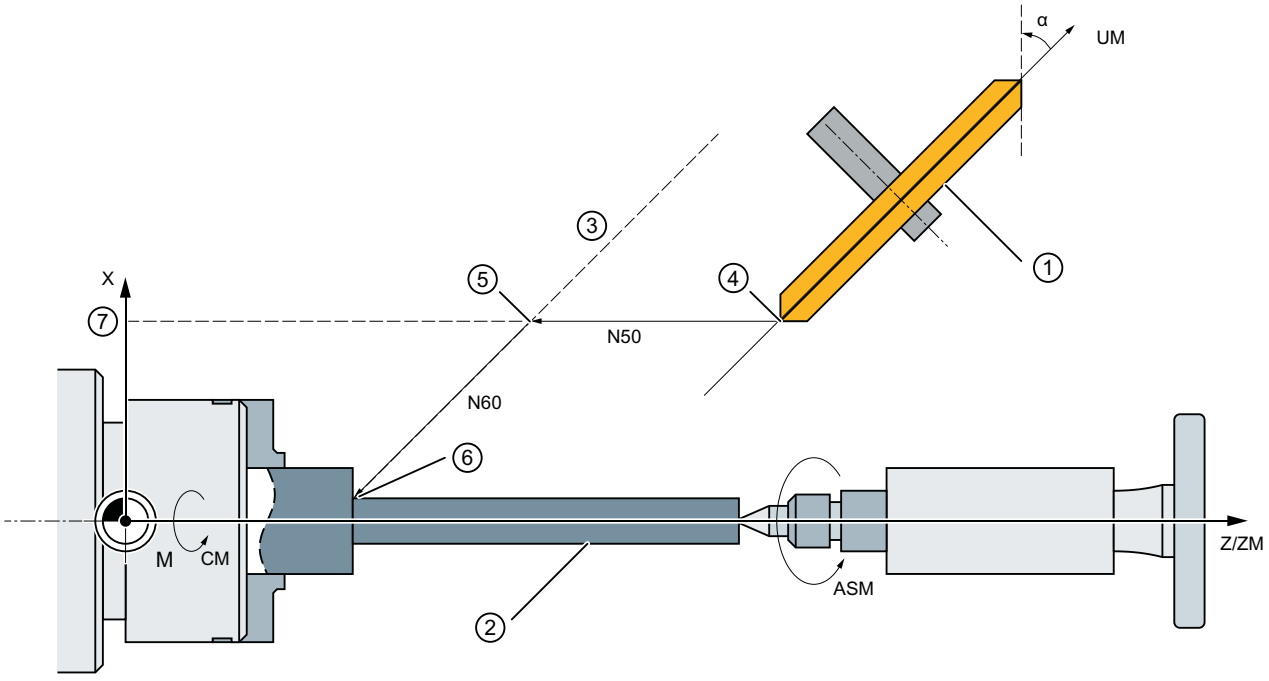
Syntaxe

```
G7 <PosFinale_X> <PosFinale_Z>
G5 <PosFinale_X>
```

Signification

`G7` :	Calcul et accostage de la position de départ pour la plongée oblique
`G5` :	Déplacement de l'axe incliné à la position finale programmée
`<PosFinale_X>` :	Position finale de l'axe X
`<PosFinale_Z>` :	Position finale de l'axe Z

Exemple



① Meule
② Pièce
③ Parallèle à l'axe incliné passant par la position finale programmée
④ Position de départ
⑤ Plongée : position de départ
⑥ Plongée : position finale
⑦ Parallèle à l'axe Z à la distance de la position actuelle de l'axe X
X Axe géométrique
Z Axe géométrique
ZM Axe machine
UM Axe machine

Figure 3-5 Programmer un axe incliné

Code de programme	**Commentaire**
`N... G18`	`; Sélection du plan XZ`
`N40 TRAANG(45.0)`	`; Activation de la transformation TRAANG, angle = 45°.`
`N50 G7 X40 Z70 F4000`	`; Calcul et accostage de la position de départ.`

Code de programme	**Commentaire**
N60 G5 X40 F100	; Déplacement de l'axe incliné à la position finale.
N70 ...	

3.9.9 Activer la transformation concaténée (TRACON)

Une transformation concaténée configurée est activée dans le programme pièce ou dans une action synchrone par l'instruction TRAFOON (Page 747) ou TRACON.

Afin de pouvoir activer une transformation paramétrée avec chaîne cinématique via TRACON, celle-ci doit être paramétrée dans la variable système $NT_TRACON_CHAIN.

Syntaxe

```
TRACON(<Trafo_Nr>,<Par_1>,...,<Par_n>,<Par_n+1>)
...
TRAFOOF
```

Signification

TRACON :	Activation de la transformation concaténée TRACON() désactive implicitement toute autre transformation active auparavant.		
<N° de transf> :	Numéro de la transformation concaténée		
	Type :	INT	
	Plage de valeurs :	0 ... 2	
	Valeur :	0, 1	Première/unique transformation concaténée
		2	Deuxième transformation concaténée
		Pas d'indication	Même signification que 0 ou 1
		Remarque : Les valeurs différentes de 0, 1, 2 génèrent une alarme.	

<Par_1>, ..., <Par_n>, <Par_n+1> :	Paramètres pour les transformations concaténées	
	<Par_1>, ..., <Par_n>	Paramètres de la première transformation de la chaîne Le nombre réel dépend du type de transformation : • TRAORI : 2 paramètres • TRACYL : 1 à 2 paramètres
	<Par_n+1>	Paramètres de la deuxième transformation de la chaîne (TRAANG) ≙ Angle de l'axe incliné
	Si des paramètres ne sont pas programmés, les préréglages ou les valeurs utilisées en derniers lieu sont actifs. Les virgules sont nécessaires pour que les paramètres spécifiés soient traités dans l'ordre dans lequel ils sont escomptés, si des préréglages doivent être pris en considération pour certains paramètres antéposés. En particulier, si un seul paramètre est indiqué, il convient de placer une virgule devant celui-ci, même si l'indication du <N° de transf> est inutile, soit par exemple TRACON(, 3.7).	
TRAFOOF :	Désactivation de la dernière transformation (concaténée) activée	

Exemple

```
Code de programme          Commentaire
...
N230 TRACON(1,45.)         ; Activation de la première transformation concaténée.
                           ; La transformation active précédemment est automatique-
                           ment désélectionnée.
                           ; L'angle pour l'axe incliné est 45°.
...
N330 TRACON(2,40.)         ; Activation de la deuxième transformation concaténée.
                           ; L'angle pour l'axe incliné est 40°.
...
N380 TRAFOOF               ; Désactivation de la deuxième transformation concaténée.
...
```

3.9.10 Déplacement PTP cartésien

3.9.10.1 Activer/désactiver le déplacement PTP cartésien (PTP, PTPG0, PTPWOC, CP)

Le déplacement point à point ou PTP cartésien est activé/désactivé dans le programme CN avec les instructions du groupe G 49.

Les instructions sont des fonctions à effet modal. Le réglage par défaut est le déplacement cartésien avec interpolation (CP).

Contrairement à la méthode CP, lorsque le déplacement PTP est actif, seule la transformation du point de destination cartésien est exécutée et les axes machine se déplacent de manière synchrone.

Pour que le point de destination cartésien puisse être converti de façon univoque en valeurs d'axes machine, d'autres informations caractérisant les positions des axes sont nécessaires en plus des indications de position et d'angle. Ces indications s'effectuent via les adresses réglables STAT (Page 724) et TU (Page 728).

Condition

La transformation TRAORI_DYN, TRAORI_STAT, TRANSMIT_K, RCTRA ou ROBX est active (TRAORI et TRANSMIT conventionnels).

RCTRA et ROBX ne sont pas disponibles pour les chaînes cinématiques.

Syntaxe

```
PTP / PTPG0 / PTPWOC
...
CP
```

Signification

`PTP` :	Activer le déplacement point à point PTP La position cartésienne programmée dans les blocs G0 et G1 est accostée avec un déplacement d'axe synchrone.
`PTPG0` :	Activer le déplacement point à point PTPG0 La position cartésienne programmée est accostée avec un déplacement d'axe synchrone uniquement dans les blocs G0. Dans les blocs G1, le déplacement avec interpolation CP est utilisé.
`PTPWOC` :	Activation du déplacement point à point PTPWOC (possible uniquement lorsque la transformation d'orientation est active) Comme PTP, mais sans les mouvements compensatoires provoqués par les déplacements d'axes rotatifs et d'axes d'orientation.
`CP` :	Désactiver le déplacement point à point et activer le déplacement avec interpolation CP Avec CP, un déplacement cartésien avec interpolation est exécuté.

Remarque

PTPWOC

En association avec une transformation RCTRA ou ROBX, l'utilisation de PTPWOC n'a pas de sens ! RCTRA et ROBX ne sont pas disponibles pour les chaînes cinématiques.

Exemples

Voir :

- Exemple 1 : Déplacement PTP d'un robot à 6 axes avec la transformation ROBX (Page 730)
- Exemple 2 : Déplacement PTP avec une transformation générique 5 axes (Page 732)
- Exemple 3 : PTPG0 et TRANSMIT (Page 732)

3.9.10.2 Indiquer la position des articulations (STAT)

L'indication de position avec des coordonnées cartésiennes et l'indication de l'orientation de l'outil ne suffisent pas pour déterminer de manière univoque la position de la machine, étant donné que plusieurs positions d'articulation sont possibles avec la même orientation d'outil. Selon le type de cinématique, 8 positions d'articulation différentes sont possibles. Ces différentes positions d'articulation sont spécifiques à la transformation.

Pour éviter les ambiguïtés, la position des articulations est indiquée sous l'adresse STAT.

Remarque

La commande prend en compte les valeurs STAT programmées uniquement lors des déplacements PTP. Lors des déplacements CP, elles sont ignorées car un changement de position n'est normalement pas possible lors du déplacement avec la transformation active. Lors du déplacement avec CP actif, la position du point de destination est reprise du point de départ.

Syntaxe

```
STAT=<valeur>
```

Signification

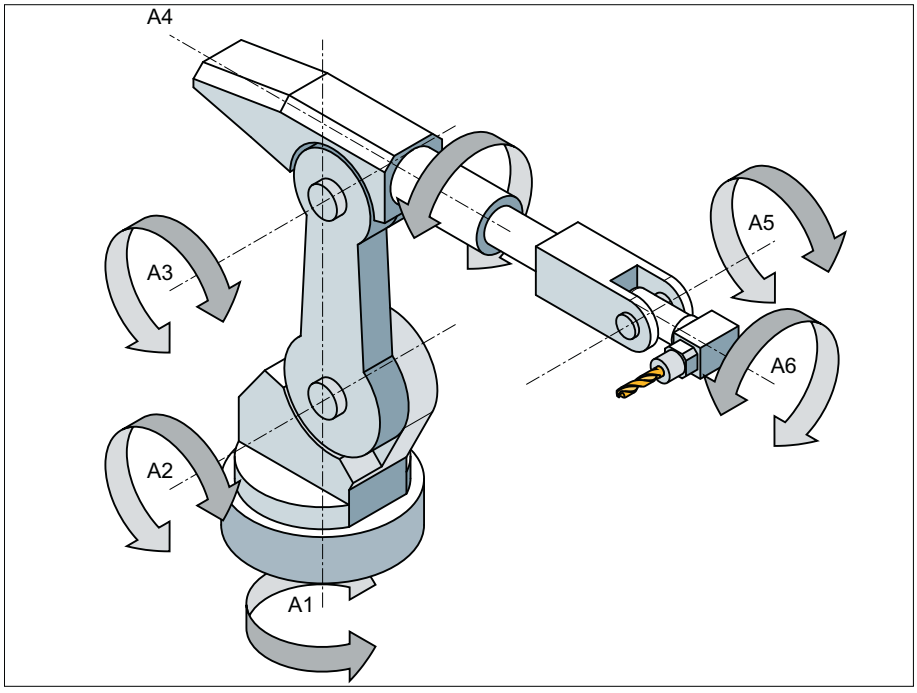
`STAT` :	Adresse réglable pour l'indication des positions d'articulation
`<valeur>` :	Valeur binaire ou décimale Contient un bit pour chacune des positions possibles. La signification des bits est définie par la transformation en question.

L'utilisation de STAT doit être illustrée en prenant l'exemple d'un robot anthropomorphe à 6 axes avec broche porte-fraise. La transformation cinématique doit s'effectuer via la transformation robotique ROBX (condition : le cycle de compilation "RMCC/ROBX Transformation robotique étendue" est chargé et actif).

Remarque

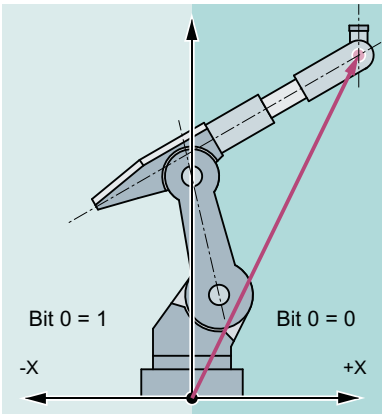
Exemple avec ROBX uniquement avec un paramétrage conventionnel via les paramètres machine

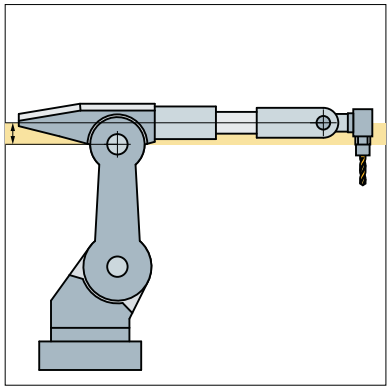
L'exemple est uniquement possible avec un paramétrage via les paramètres machine, pas avec des chaînes cinématiques.



Les axes A1, A2 et A3 sont les axes principaux du robot anthropomorphe. Les axes A4, A5 et A6, qui sont appelés axes de tête ou de poignet, sont positionnés dans la zone de travail avec les axes principaux. Grâce aux possibilités de déplacement supplémentaires des axes de poignet, la broche porte-fraise peut être orientée dans la zone de la manière requise pour la tâche d'usinage. Ainsi, avec la même orientation de l'outil, différentes positions d'articulation sont possibles.

La sélection des positions d'articulation à utiliser pour l'usinage s'effectue en programmant le bit 0 ... 2 de l'adresse STAT réglable :

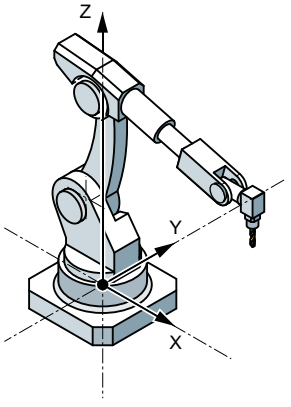
Bit 0	Position du point d'intersection des axes de poignet (A4, A5, A6)		
	= 0	Zone de base (**Shoulder Right**) Le robot se trouve dans la zone de base lorsque la valeur X du point d'intersection des axes de poignet, rapportée au système de coordonnées A1, est positive.	 Exemple : Le point d'intersection des axes de poignet se trouve dans la zone de base
	= 1	Zone de plafond (**Shoulder Left**) Le robot se trouve dans la zone de plafond lorsque la valeur X du point d'intersection des axes de poignet, rapportée au système de coordonnées A1, est négative.	

Bit 1	Position de l'axe 3 L'angle auquel la valeur du bit 1 change dépend du type de robot. Pour les robots dont les axes 3 et 4 se coupent, on a :	
	= 0	A3 < 0° (**Elbow Down**)
	= 1	A3 ≥ 0° (**Elbow Up**)
	Remarque : Pour les robots avec un décalage entre les axes 3 et 4, l'angle auquel la valeur du bit 1 change dépend de la valeur de ce décalage.	 Offset entre A3 et A4
Bit 2	Position de l'axe 5	
	= 0	A5 ≥ 0° (**No Handflip**)
	= 1	A5 < 0° (**Handflip**)

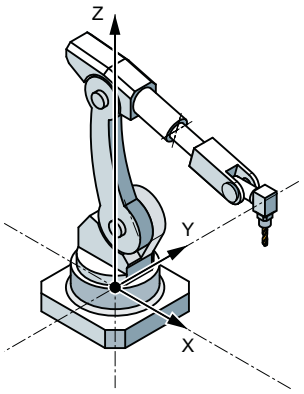
Exemple de programme :

Code de programme	**Commentaire**
...	
N14 T="T8MILLD20" D1	; $TC_DP3[1,1]=132.95
N16 ORIMKS	
N17 G1 PTP X1665.67 Y0 Z1377.405 A=0 B=0 C=0 **STAT=...** F2000	; La valeur STAT détermine les positions d'articulation (voir ci-dessous).
...	

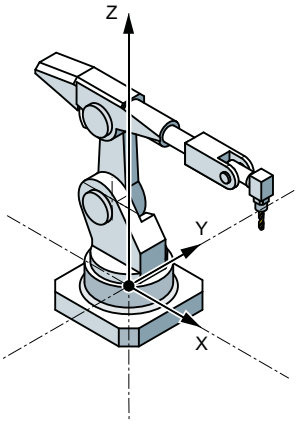
STAT=1 ('B001') → Shoulder Left
→ Elbow Down
→ No Handflip



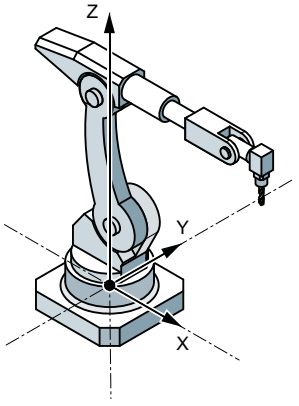
`STAT=2 ('B010')`
→ Shoulder Right
→ Elbow Up
→ No Handflip



`STAT=5 ('B101')`
→ Shoulder Left
→ Elbow Down
→ Handflip



`STAT=6 ('B110')`
→ Shoulder Right
→ Elbow Up
→ Handflip



TRANSMIT_K (TRANSMIT conventionnel)

Avec TRANSMIT_K, l'adresse STAT est utilisée pour éviter les ambiguïtés en ce qui concerne le pôle.

Lorsque l'axe rotatif doit tourner de 180° ou lorsque le contour passerait par le pôle avec CP, on a :

Bit 0	Pertinent uniquement quand $NT_POLE_SIDE_FIX[n] = 1 ou 2 Lors du paramétrage via les paramètres machine, $MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 1 ou 2 :	
	= 0	L'axe rotatif se déplace de +180° ou tourne dans le sens des aiguilles d'une montre.
	= 1	L'axe rotatif se déplace de -180° ou tourne dans le sens inverse des aiguilles d'une montre.
Bit 1	Pertinent uniquement quand $NT_POLE_SIDE_FIX[n] = 0 Lors du paramétrage via les paramètres machine, $MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 0 :	
	= 0	Il y a un passage par le pôle. L'axe rotatif ne tourne pas.
	= 1	Il y a une rotation autour du pôle. Le bit 0 de STAT est alors pertinent.

3.9.10.3 Indiquer les signes des angles d'axe (TU)

Avec les axes rotatifs, pour pouvoir accoster également les angles d'axe supérieurs à +180° ou inférieurs à -180° sans stratégie de déplacement particulière (par ex. points intermédiaires), les signes des angles d'axe doivent être indiqués sous l'adresse réglable TU.

Remarque

La commande prend en compte les valeurs TU programmées uniquement lors des déplacements PTP. Lors des déplacements CP, elles sont ignorées.

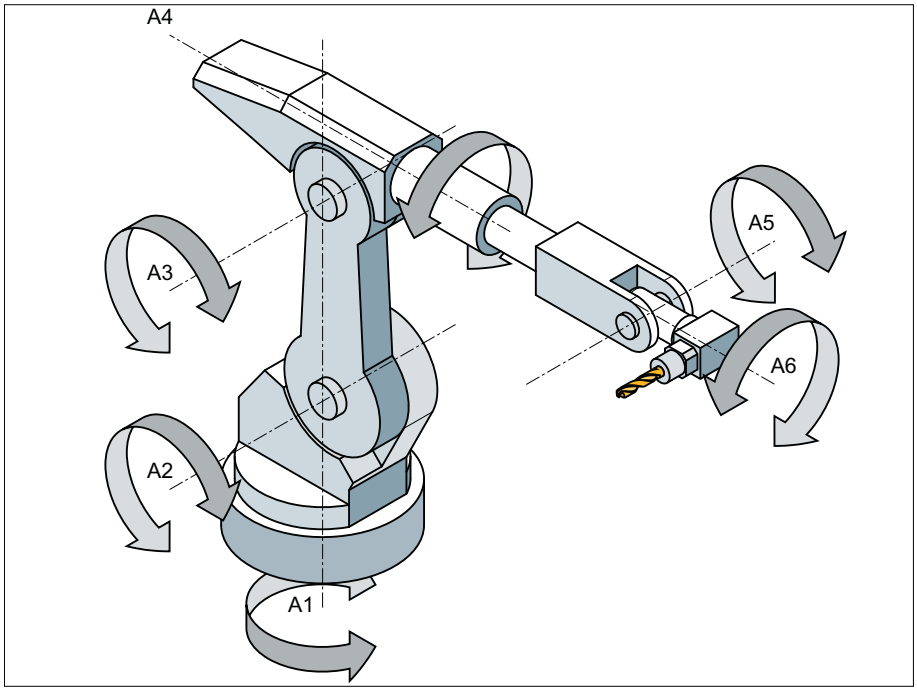
Syntaxe

```
TU=<valeur>
```

Signification

`TU` :	Adresse réglable pour l'indication des signes des angles d'axe			
`<valeur>` :	Valeur binaire ou décimale Pour chaque axe qui entre dans la transformation, un bit indique le signe de l'angle d'axe (θ) et par conséquent le sens de déplacement.			
	Bit	= 0	Signe de l'angle d'axe : +	Plage de l'angle d'axe : 0° ≤ θ < 360°
		= 1	Signe de l'angle d'axe : -	Plage de l'angle d'axe : - 360° < θ < 0°

Exemple : Robot anthropomorphe à 6 axes



Bit	Signification	Valeur	Signe de l'angle d'axe	Angle d'axe
Bit 0 [1]	Signe pour l'angle d'axe d'A1	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 1 [1]	Signe pour l'angle d'axe d'A2	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 2 [1]	Signe pour l'angle d'axe d'A3	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 3 [1]	Signe pour l'angle d'axe d'A4	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 4 [1]	Signe pour l'angle d'axe d'A5	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 5 [1]	Signe pour l'angle d'axe d'A6	= 0	+	≥ 0°
		= 1	-	< 0°

[1] Les numéros de bit TU réels sont obtenus à partir des numéros d'axe de canal des axes du robot ! Dans l'exemple, les axes du robot (A1 à A6) sont les six premiers axes du canal, par conséquent les bits TU 0 ... 5 sont utilisés. En cas d'affectation différente des axes de canal aux axes du robot, les numéros de bit TU des axes du robot changeraient en conséquence (par ex. : les axes du robot sont les axes 3 à 8 du canal, c'est-à-dire que les bits TU 2 ... 7 sont utilisés pour les axes du robot).

TU=19 (correspond à TU='B010011) signifierait également :

Bit	Valeur		Angle d'axe
0	= 1	⇒	$\theta_{A1} < 0°$
1	= 1	⇒	$\theta_{A2} < 0°$

2	= 0	⇒	$\theta_{A3} \geq 0°$
3	= 0	⇒	$\theta_{A4} \geq 0°$
4	= 1	⇒	$\theta_{A5} < 0°$
5	= 0	⇒	$\theta_{A6} \geq 0°$

Remarque

Pour les axes avec une plage de déplacement > ±360°, le déplacement se fait toujours par le trajet le plus court, car la position des axes ne peut pas être déterminée de manière univoque au moyen des informations TU.

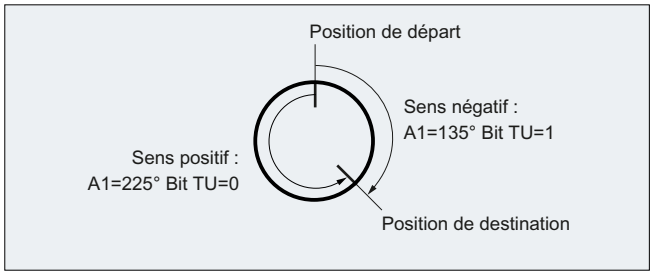
Si TU n'est pas programmé pour une position, le déplacement se fait par le trajet le plus court ou le plus long en fonction du MD30455 $MA_MISC_FUNCTION_MASK.

TRANSMIT_K (TRANSMIT conventionnel)

Lors du déplacement PTP avec TRANSMIT actif, l'adresse TU est sans importance.

Exemple

La position d'axes rotatifs indiquée dans l'illustration suivante peut être accostée en sens négatif ou positif. La position d'angle est programmée sous l'adresse A1. Le sens de déplacement ne devient univoque qu'avec l'indication TU.



3.9.10.4 Exemple 1 : Déplacement PTP d'un robot à 6 axes avec la transformation ROBX

Dans l'exemple d'application suivant, le déplacement PTP cartésien et les instructions CN s'y rapportant sont illustrés de manière exemplaire.

Remarque

Uniquement avec un paramétrage conventionnel via les paramètres machine

L'exemple ne fonctionne pas avec des chaînes cinématiques.

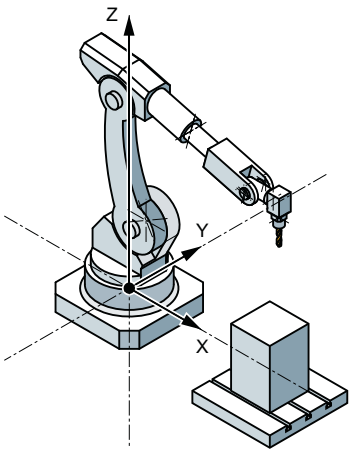


Figure 3-6 Robot anthropomorphe à 6 axes avec broche porte-fraise

```
N1 G90
N2 T="T8MILLD20" D1 M6
N3 TRAORI
;$P_UIFR[1]=CTRANS(X,1500,Y,0,Z,400):CROT(X,0,Y,0,Z,-90)
N4 G54
N5 M3 S20000
N6 ORIWKS
N7 ORIVIRT1
N8 CYCLE832(0.01,_FINISH,1)
;HOME
N9 TRAFOOF
N10 G0 RA1=0.0000 RA2=-90.0000 RA3=90.0000 A=0.0000 B=90.0000 C=0.0000
N11 TRAORI
N12 G54
N13 G0 PTP X1369.2426 Y956.7528 Z502.5517 A=135.5761 B=-33.2223
C=161.1435 STAT='B010' TU='B001011'
N14 G0 X1355.1242 Y1014.9394 Z424.9695 A=135.8491 B=-33.1439
C=160.9941 STAT='B010' TU='B001011'
N15 G1 CP X1354.8361 Y1016.1269 Z423.3862 A=136.0635 B=-33.0819 C=160.8770
F1000
N16 G1 X1336.4283 Y1016.1269 Z426.6311 A=136.0484 B=-32.2151 C=160.9643
F2000
N17 G1 X1317.9831 Y1016.1269 Z429.6730 A=136.0175 B=-31.3394 C=161.0655
;HOME
N18 TRAFOOF
N19 G0 RA1=0.0000 RA2=-90.0000 RA3=90.0000 A=0.0000 B=90.0000 C=0.0000
N20 M30
```

3.9.10.5 Exemple 2 : Déplacement PTP avec une transformation générique 5 axes

Hypothèse : Une cinématique orthogonale CA sert de base.

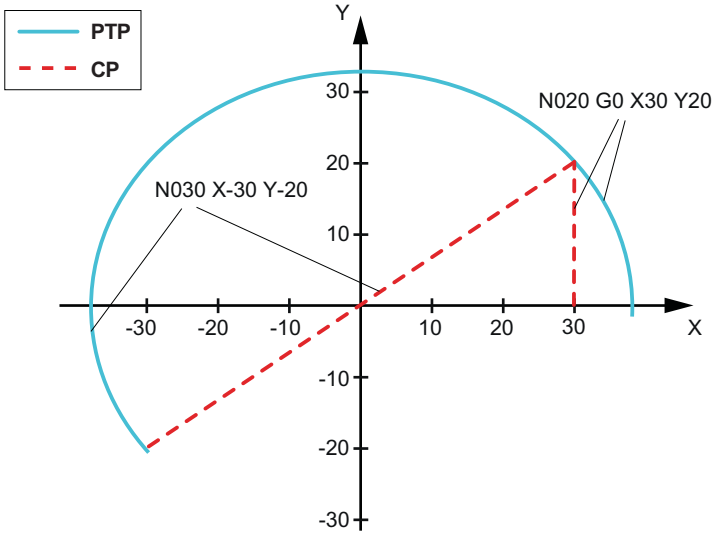
Code de programme	Commentaire
TRAORI	; Transformation de cinématique CA activée
PTP	; Activer le déplacement PTP
N10 A3=0 B3=0 C3=1	; Positions des axes rotatifs C=0 A=0
N20 A3=1 B3=0 C3=1	; Positions des axes rotatifs C=90 A=45
N30 A3=1 B3=0 C3=0	; Positions des axes rotatifs C=90 A=90
N40 A3=1 B3=0 C3=1 **STAT=1**	; Positions des axes rotatifs C=270 A=-45

Sélectionner la position d'accostage univoque de l'axe rotatif :

Dans le bloc N40, les axes rotatifs parcourent, du fait de la programmation **STAT=1**, le trajet le plus long de leur point de départ (C=90, A=90) à leur point final (C=270, A=–45). Au lieu de cela, avec **STAT=0**, les axes rotatifs parcourraient le trajet le plus court jusqu'au point final (C=90, A=45).

3.9.10.6 Exemple 3 : PTPG0 et TRANSMIT

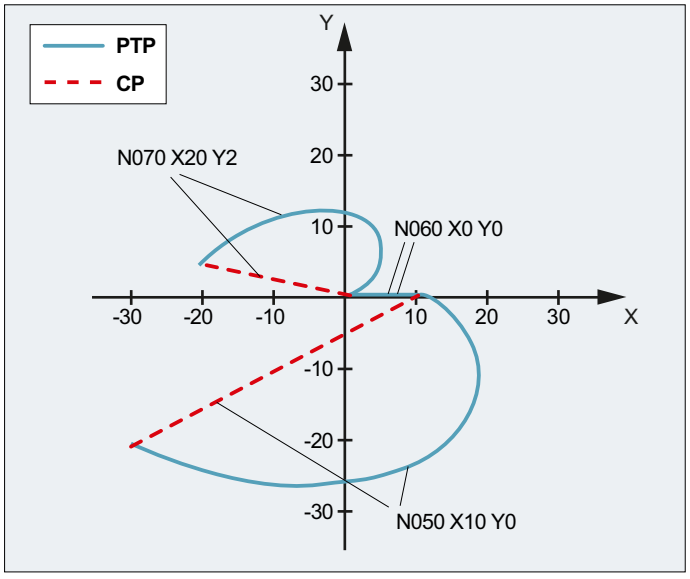
Contournement du pôle avec PTPG0 et TRANSMIT



Code de programme	Commentaire
N001 G0 X30 Z0 F10000 T1 D1 G90	; Position de départ Cote absolue
N002 SPOS=0	
N003 **TRANSMIT**	; Transformation TRANSMIT
N010 **PTPG0**	; PTP automatiquement pour chaque bloc G0 puis à nouveau CP
N020 G0 X30 Y20	
N030 X-30 Y-20	

Code de programme	Commentaire
N120 G1 X30 Y20	
N110 X30 Y0	
M30	

Dégagement du pôle avec PTPG0 et TRANSMIT



Programmation	Commentaire
N001 G0 X90 Z0 F10000 T1 D1 G90	; Position de départ
N002 SPOS=0	
N003 **TRANSMIT**	; Transformation TRANSMIT
N010 **PTPG0**	; PTP automatiquement pour chaque bloc G0 puis à nouveau CP
N020 G0 X90 Y60	
N030 X-90 Y-60	
N040 X-30 Y-20	
N050 X10 Y0	
N060 X0 Y0	
N070 X-20 Y2	
N170 G1 X0 Y0	
N160 X10 Y0	
N150 X-30 Y-20	
M30	

3.9.11 Conditions marginales pour l'activation d'une transformation

Fonction

La sélection de transformations est possible par le biais du programme pièce ou en mode MDA. Attention :

- Aucun bloc intermédiaire de déplacement n'est inséré (chanfreins/rayons).
- Une séquence de blocs de type spline doit être terminée, sinon un message s'affiche.
- La correction fine d'outil doit être désactivée (FTOCOF), sinon un message s'affiche.
- La correction de rayon d'outil doit être désactivée (G40), sinon un message s'affiche.
- Une correction de longueur d'outil activée est reprise dans la transformation par la commande.
- Le frame courant activé avant la transformation est désactivé par la commande.
- Une limitation active de la zone de travail est désactivée par la commande pour les axes impliqués dans la transformation (correspond à WALIMOF).
- La surveillance de la zone de protection est désactivée.
- Le contournage et l'arrondissement sont interrompus.
- Tous les axes indiqués dans les paramètres machine doivent être synchronisés au niveau des blocs.
- Les axes permutés sont rétablis, sinon un message apparaît.
- Un message apparaît en présence d'axes dépendants.

Changement d'outil

Un changement d'outil n'est autorisé que si la correction de rayon d'outil est désactivée.

Un changement de la correction de longueur d'outil et une activation/ désactivation de la correction de rayon d'outil ne doivent pas être programmés dans le même bloc.

Changement de frame

Toutes les instructions qui se rapportent uniquement au système de coordonnées de base sont autorisées (FRAME, correction de rayon d'outil). Un changement de frame avec G91 (cotes relatives) ne sera cependant pas traité à part, contrairement à ce qui se passe lorsque la transformation est inactive. L'incrément à parcourir est traité dans le système de coordonnées pièce du nouveau frame, quel que soit le frame qui était actif dans le bloc précédent.

Exclusions

Les axes impliqués dans une transformation ne peuvent pas être utilisés :

- comme axes Preset (alarme),
- pour l'accostage de point fixe (alarme),
- pour la prise de référence (alarme).

3.9.12 Désactivation de la transformation (TRAFOOF)

La procédure prédéfinie TRAFOOF permet de désactiver tous les frames et transformations actifs.

Remarque

Les conditions marginales (Page 734) pour l'activation d'une transformation s'appliquent également à la désactivation d'une transformation.

Les frames nécessaires doivent ensuite être activés par une nouvelle programmation.

Syntaxe

```
...
TRAFOOF
```

Signification

`TRAFOOF` :	Désactiver l'ensemble des transformations/frames actifs

3.10 Chaînes cinématiques

3.10.1 Suppression de composants (DELOBJ)

La fonction `DELOBJ()` "supprime" des composants en réinitialisant les variables système affectées sur leur valeur par défaut :

- Eléments de chaînes cinématiques
- Zones de protection, éléments de zone de protection et paires de collision
- Données de transformation

Syntaxe

```
[<RetVal>=] DELOBJ(<CompType>[,,,<NoAlarm>)])
[<RetVal>=] DELOBJ(<CompType>,<Index1>[,,<NoAlarm>])
[<RetVal>=] DELOBJ(<CompType>[,<Index1>][,<Index2>][,<NoAlarm>])
```

Signification

`DELOBJ` :	Suppression d'éléments de chaînes cinématiques, de zones de protection, d'éléments de zones de protection, de paires de collision et de données de transformation

<CompType> :	Type du composant à supprimer	
	Type de données :	STRING
	Valeur : "KIN_CHAIN_ELEM" **Signification** : Variables système de tous les éléments cinématiques : $NK_...	
	Valeur : "KIN_CHAIN_SWITCH" **Signification** : Variable système $NK_SWITCH[<i>]	
	Valeur : "KIN_CHAIN_ALL" **Signification** : Tous les éléments et commutateurs cinématiques. Même signification que l'appel successif de DELOBJ avec "KIN_CHAIN_ELEM" et "KIN_CHAIN_SWITCH"	
	Valeur : "PROT_AREA" **Signification** : Variables système des zones de protection : • $NP_PROT_NAME • $NP_CHAIN_NAME • $NP_CHAIN_ELEM • $NP_1ST_PROT	
	Valeur : "PROT_AREA_ELEM" **Signification** : Variables système des éléments de zone de protection des zones de protection pour machine et/ou des zones de protection automatiques pour outils : • $NP_NAME • $NP_NEXT • $NP_NEXTP • $NP_COLOR • $NP_D_LEVEL • $NP_USAGE • $NP_TYPE • $NP_FILENAME • $NP_PARA • $NP_OFF • $NP_DIR • $NP_ANG	
	Valeur : "PROT_AREA_COLL_PAIRS" **Signification** : Variables système des paires de collision : • $NP_COLL_PAIR • $NP_SAFETY_DIST	
	Valeur : "PROT_AREA_ALL" **Signification** : L'ensemble des zones de protection, des éléments de zone de protection et des paires de collision (variables système $NP_...) Même signification que l'appel successif de DELOBJ avec "PROT_AREA", "PROT_AREA_ELEM" et "PROT_AREA_COLL_PAIRS"	
	Valeur : "TRAFO_DATA" **Signification** : Variables système de toutes les transformations $NT_...	

<Index1> :	Indice du premier composant à supprimer (**facultatif**)	
	Type de données :	INT
	Valeur par défaut :	-1
	Plage de valeurs :	-1 ≤ x ≤ (nombre maximal de composants configurés -1)
	Valeur	**Signification**
	0, 1, 2, ...	Indice du composant à supprimer
	-1	Tous les composants du type indiqué sont supprimés. <Index2> n'est pas exploité.
<Index2> :	Indice des derniers composants à supprimer (**facultatif**) Si <Index2> n'est pas programmé, seules les variables système du composant référencé dans <Index1> sont supprimées.	
	Type de données :	INT
	Valeur par défaut :	Seules les variables système du composant référencé dans <Index1> sont supprimées.
	Plage de valeurs :	<Index1> < x ≤ (nombre max. de composants configurés -1)
<NoAlarm> :	Inhibition des alarmes (**facultatif**)	
	Type de données :	BOOL
	Valeur par défaut :	FALSE
	Valeur	**Signification**
	FALSE	En cas d'erreur (<RetVal> < 0), l'exécution du programme est arrêtée et une alarme est affichée.
	TRUE	En cas d'erreur, l'exécution du programme n'est pas arrêtée et aucune alarme n'est affichée. Cas d'application : réaction personnalisée en fonction de la valeur de retour

<RetVal> :	Valeur de retour de la fonction	
	Type de données :	INT
	Plage de valeurs :	0, -1, -2, ... -7
	Valeur	**Signification**
	0	Aucune erreur n'est survenue
	-1	Appel de la fonction sans paramètre. Au moins le paramètre <CompType> doit être spécifié.
	-2	<CompType> désigne un composant inconnu
	-3	<Index1> est inférieur à -1
	-4	<Index1> est supérieur au nombre de composants configuré
	-5	<Index1> a une valeur différente de -1 lors de la suppression d'un **groupe** de composants
	-6	<Index2> est inférieur à <Index1>
	-7	<Index2> est supérieur au nombre de composants configuré
	-8	Autorisation d'écriture insuffisante

3.10.2 Détermination de l'indice par le nom (NAMETOINT)

Des noms personnalisés sont saisis dans des tableaux de variables système du type STRING. Au moyen du descripteur des variables système et du nom, la fonction `NAMETOINT()` détermine la valeur de l'indice correspondant au nom, sous lequel il est enregistré dans le tableau de variables système.

Syntaxe

```
<RetVal> = NAMETOINT(<SysVar>,<Name>[,<NoAlarm>])
```

Signification

NAMETOINT :	Détermination de l'indice de variable système	
<SysVar> :	Nom du tableau de variables système du type STRING	
	Type de données :	STRING
	Plage de valeurs :	Noms de tous les tableaux de variables système de la CN du type STRING
<Name> :	Chaîne de caractères ou nom pour lequel/laquelle l'indice de variable système doit être déterminé.	
	Type de données :	STRING

<NoAlarm> :	Inhibition des alarmes (**facultatif**)	
	Type de données :	BOOL
	Valeur par défaut :	FALSE
	Valeur	**Signification**
	TRUE	En cas d'erreur, l'exécution du programme n'est pas arrêtée et aucune alarme n'est affichée. Cas d'application : réaction personnalisée en fonction de la valeur de retour
	FALSE	En cas d'erreur (<RetVal> < 0), l'exécution du programme est arrêtée et une alarme est affichée.
<RetVal> :	Indice de variables système ou message d'erreur	
	Type de données :	INT
	Plage de valeurs :	-1 ≤ x ≤ (nombre max. de composants configurés -1)
	Valeur	**Signification**
	≥ 0	Le nom recherché a été trouvé sous l'indice de variables système indiqué.
	-1	Le nom recherché n'a pas été trouvé ou une erreur est survenue.

Exemple

```
Code de programme                                                    Commentaire
DEF INT INDEX
$NP_PROT_NAME[27] = "Recouvrement"
...
INDEX = NAMETOINT("$NP_PROT_NAME","Recouvrement")                    ; INDEX == 27
```

3.11 Prévention des collisions avec chaînes cinématiques

Remarque

Zones de protection

Les zones de protection évoquées dans les chapitres suivants se rapportent à la fonction "Modélisation géométrique de la machine".

Des informations complémentaires sur cette fonction figurent dans la description fonctionnelle Surveillance et compensation.

3.11.1 Contrôle de la présence de paire de collision (COLLPAIR)

La fonction `COLLPAIR(...)` détermine si deux zones de protection forment une paire de collision.

Syntaxe

```
[<RetVal> =] COLLPAIR(<Name_1>,<Name_2>[,<NoAlarm>)])
```

Signification

`COLLPAIR`:	Vérification de l'appartenance à une paire de collision		
`<RetVal>`:	Valeur de retour de la fonction		
	Type de données :	INT	
	Valeur :	≥ 0	Les deux zones de protection forment une paire de collision. Valeur de retour == Indice de paire de collision m (voir $NP_COLL_PAIR)
		-1	Soit moins de deux chaînes ont été spécifiées, soit l'une des deux est une chaîne vide.
		-2	La zone de protection spécifiée dans le premier paramètre n'a pas été trouvée.
		-3	La zone de protection spécifiée dans le second paramètre n'a pas été trouvée.
		-4	Aucune des deux zones de protection spécifiées n'a été trouvée.
		-5	Les deux zones de protection spécifiées ont été trouvées, mais pas ensemble dans une paire de collision.
`<Name_1>`:	Nom de la première zone de protection		
	Type de données :	STRING	
	Plage de valeurs :	Noms de zone de protection paramétrés	

<Name_2>:	Nom de la seconde zone de protection		
	Type de données :	STRING	
	Plage de valeurs :	Noms de zone de protection paramétrés	
<NoAlarm>:	Inhibition des alarmes **(facultatif)**		
	Type de données :	BOOL	
	Valeur :	FALSE (Par défaut)	En cas d'erreur (<RetVal> < 0), l'exécution du programme est arrêtée et une alarme est affichée.
		TRUE	En cas d'erreur, l'exécution du programme n'est pas arrêtée et aucune alarme n'est affichée. Cas d'application : réaction personnalisée en fonction de la valeur de retour

3.11.2 Demander un nouveau calcul du modèle de machine pour la prévention des collisions (PROTA)

Lorsque des variables système $NK_..., de la modélisation géométrique de la machine ou de la prévention de collisions $NP_... sont écrites dans le programme pièce, la procédure PROTA doit ensuite être appelée afin de rendre effectives les modifications du modèle de machine interne à la CN pour la prévention de collisions.

Syntaxe

```
PROTA[(<Par>)]
```

Signification

PROTA:	Demander un nouveau calcul du modèle de machine pour la prévention des collisions		
	• Déclenche un arrêt du prétraitement des blocs. • Doit figurer seule dans le bloc.		
<Par>:	Paramètres (**option**)		
	Type de données :	STRING	
	Valeur :	---	Sans paramètre. Un nouveau calcul du modèle de machine est effectué. Les états des zones de protection restent inchangés.
		"R"	Un nouveau calcul du modèle de machine est effectué. Les zones de protection sont mises dans leur état d'initialisation en fonction de $NP_INIT_STAT.

Autres conditions à prendre en compte

Simulation

La procédure PROTA ne doit pas être utilisée dans des programmes pièce en association avec la simulation (simNC).

Exemple : Empêchement de l'appel de PROTA lorsque la simulation est active.

```
Code de programme              Commentaire
...
IF $P_SIM == FALSE             ; IF simulation inactive
   PROTA                       ; THEN recalculer le modèle de collision
ENDIF                          ; ENDIF
...
```

Voir aussi

Réglage de l'état des zone de protection (PROTS) (Page 744)

3.11.3 Réglage de l'état des zone de protection (PROTS)

La procédure PROTS(...) règle l'état des zones de protection sur la valeur indiquée.

Syntaxe

```
PROTS(<State>[,<Name_1>,...,<Name_n>])
```

Signification

PROTS :	Réglage de l'état des zones de protection		
	• Doit figurer seule dans le bloc.		
<State> :	Etat sur lequel les zones de protection spécifiées doivent être réglées		
	Type de données :	CHAR	
	Valeur :	"A" ou "a"	Etat : active
		"I" ou "i"	Etat : inactive
		"P" ou "p"	Etat : préactivé ou commandé par AP [1]
		"R" ou "r"	État : Valeur de l'état d'initialisation interne à la CN [2)]

<Name_1> ... <Name_n>:	Nom d'une ou de plusieurs zones de protection à régler sur l'état spécifié (**option**) Si aucun nom n'est spécifié, l'état spécifié est réglé pour l'ensemble des zones de protection définies.	
	Type de données :	STRING
	Plage de valeurs :	Noms de zone de protection paramétrés
	Remarque : Le nombre maximal de zones de protection pouvant être spécifiées en paramètre dépend uniquement du nombre maximal possible de caractères par ligne de programme.	
[1] L'activation/désactivation est effectuée via : DB10.DBX234.0 - DBX241.7 [2] L'état est réglé sur la valeur de l'état d'initialisation interne à la CN, c.-à-d. sur la valeur de la variable système $NP_INIT_STAT à l'instant du dernier appel de PROTA(...) (Page 743).		

3.11.4 Détermination de la distance entre deux zones de protection (PROTD)

La fonction `PROTD(...)` calcule la distance entre deux zones de protection.

Caractéristiques de la fonction :

- Le calcul de la distance est effectué indépendamment de l'état des zones de protection (activée, désactivée, pré-activée).
- Pour le calcul des distances de deux zones de protection, seuls les éléments de zone de protection marqués par $NP_USAGE = "C" ou "A" peuvent être utilisés. Les éléments de la zone de protection marqués par $NP_USAGE = "V" seront ignorés.
- Les zones de protection dont tous les éléments sont marqués par $NP_USAGE = "V" ne peuvent pas être utilisées pour le calcul des distances.
- Le calcul de la distance est effectué avec les positions valides à la fin du bloc précédent.
- Les chevauchements qui sont pris en compte dans l'exécution des blocs, p. ex. décalage DRF ou décalages d'origine externes, entrent dans le calcul de la distance avec les valeurs valides au **moment de l'interprétation** de la fonction.

Remarque

Synchronisation

Lorsque la fonction `PROTD(...)` est utilisée, l'utilisateur assume l'entière responsabilité quant à une synchronisation de l'exécution et du prétraitement des blocs le cas échéant au moyen d'un arrêt du prétraitement des blocs `STOPRE`.

Collision

En présence d'une collision entre les zones de protection spécifiées, la fonction présente une distance de 0,0. Une collision existe lorsque les deux zones de protection se touchent ou interfèrent.

La distance de sécurité pour le contrôle de collision (PM10622 $MN_COLLISION_SAFETY_DIST) n'est pas prise en compte dans le calcul de la distance.

Syntaxe

```
[<RetVal> =] PROTD([<Name_1>],[<Name_2>],VAR <Vector>[,<System>])
```

Signification

PROTD:	Calcul de la distance entre les deux zones de protection spécifiées		
	• Doit figurer seule dans le bloc.		
<RetVal>:	Valeur en retour de la fonction : Valeur absolue de la distance entre les deux zones de protection ou 0,0 en cas de collision (voir ci-dessus : paragraphe Collision)		
	Type de données :	REAL	
	Plage de valeurs :	0,0 ≤ x ≤ +max. Valeur REAL	
<Name_1>, <Name_2>:	Noms des deux zones de protection dont la distance les séparant doit être calculée (**option**)		
	Type de données :	STRING	
	Plage de valeurs :	Noms de zone de protection paramétrés	
	Valeur par défaut :	"" (chaîne vide) Si aucune zone de protection n'est spécifiée, la fonction calcule la distance actuellement la plus faible pour toutes les zones de protection activées et pré-activées contenues dans le modèle de collision.	
<Vector>:	Valeur de retour : vecteur de distance tridimensionnel de la zone de protection <Name_2> à la zone de protection <Name_1> avec : • <Vector>[0]: Cordonnée X dans le système de coordonnées mondial • <Vector>[1]: Cordonnée Y dans le système de coordonnées mondial • <Vector>[2]: Cordonnée Z dans le système de coordonnées mondial En cas de collision : <Vector> == vecteur nul		
	Type de données :	VAR REAL [3]	
	Plage de valeurs :	<Vector> [n]: 0,0 ≤ x ≤ ±valeur REAL max.	
<System>:	Système de mesure (anglo-saxon / métrique) pour la distance et le vecteur de distance (**facultatif**)		
	Type de données :	BOOL	
	Valeur :	FALSE (Par défaut)	Système de mesure correspondant à l'instruction G actuellement active du groupe G 13 (G70, G71, G700, G710).
		TRUE	Système de mesure correspondant au système de base réglé : PM52806 $MN_ISO_SCALING_SYSTEM

3.12 Transformations avec chaînes cinématiques

3.12.1 Activation de la transformation (TRAFOON)

Une transformation avec chaînes cinématiques est activée par la procédure prédéfinie TRAFOON. L'appel doit être isolé dans un bloc.

Remarque

Variante : une transformation définie avec chaînes cinématiques peut également être activée à l'aide des instructions conventionnelles, par ex. TRAORI ou TRANSMIT. Une valeur différente de zéro doit être renseignée à cet effet dans la variable système $NT_TRAFO_INDEX.

Pour plus d'informations sur $NT_TRAFO_INDEX, voir "Tables de paramètres Variables système".

Syntaxe

```
TRAFOON(<Trafoname>,<Diameter>,<k>)
```

Signification

TRAFOON:	Procédure d'activation d'une transformation avec chaînes cinématiques		
<Trafoname> :	Nom du jeu de paramètres de transformation		
	Type de données :	STRING	
	Plage de valeurs :	Tous les noms de jeu de paramètres de transformation définis dans $NT_NAME	
	Remarque : Le nom du jeu de paramètres de transformation doit être univoque Il ne doit être présent dans $NT_NAME qu'une seule fois.		
<Diameter> :	Diamètre de référence ou d'usinage (uniquement TRACYL)		
	Type de données :	REAL	
	La valeur doit être > 1.		
<k> :	Définit l'utilisation de la correction des flancs de rainure (uniquement TRACYL).		
	Type de données :	BOOL	
	Valeur :	FALSE	sans correction des flancs de rainure
		TRUE	avec correction des flancs de rainure
	Correspond au type de transformation TRACYL 514 (correction des flancs de rainure programmable). Si <k> n'est pas renseigné, le réglage paramétré de bit 10 dans $NT_CNTRL[<n>] est effectif.		

Exemple

Code de programme	Commentaire
TRAFOON["Trans_1"]	Active la transformation du nom Trans_1.

3.12.2 Modification de la transformation d'orientation après mesurage de la machine (CORRTRAFO)

Dans le cas des machines avec transformations d'orientation définies à l'aide de chaînes cinématiques, l'utilisateur peut utiliser la fonction prédéfinie CORRTRAFO, permettant, après un mesurage de la machine, de modifier les vecteurs d'offset ou les vecteurs directionnels des axes d'orientation dans le modèle cinématique de la machine.

Remarque

Les valeurs de correction écrites avec la fonction CORRTRAFO ne sont pas immédiatement effectives dans la transformation. Les valeurs de correction prennent seulement effet après désactivation de la transformation, NEWCONF et activation de la transformation.

Syntaxe

```
<Corr_Status> = CORRTRAFO(<Corr_Vect>, <Corr_Index>, <Corr_Mode>,
[ <No_Alarm>])
```

Signification

CORRTRAFO:	Appel de fonction

<Corr_Status> :	Valeur de retour de la fonction		
	Type de données :	INT	
	Valeurs :	0	La fonction a été exécutée sans erreur.
		1	Aucune transformation n'est active.
		2	La transformation actuellement active n'est pas une transformation d'orientation.
		3	La transformation d'orientation active n'a pas été définie à l'aide de chaînes cinématiques.
		10	Le paramètre d'appel <Corr_Index> est négatif.
		11	Le paramètre d'appel <Corr_Mode> est négatif.
		12	Renvoi non valide sur une section d'une chaîne partielle (position des unités dans <Corr_Index>). La valeur ne doit pas être supérieure au nombre d'axes d'orientation dans la chaîne partielle.
		13	Renvoi non valide sur l'axe d'orientation d'une chaîne partielle (position des unités dans <Corr_Index>). La valeur doit être inférieure au nombre d'axes d'orientation dans la chaîne partielle.
		14	Renvoi non valide sur une chaîne partielle (position des dizaines dans <Corr_Index>). Seules les valeurs 0 et 1 sont autorisées (renvoi sur une chaîne Part ou une chaîne Tool). Ce numéro d'erreur est également signalé lorsque la chaîne partielle figurant dans le renvoi <Corr_Index> n'existe pas.
		15	Aucun élément de correction n'est défini dans la section référencée par le paramètre <Corr_Index> ($NT_CORR_ELEM_P ou $NT_CORR_ELEM_T).
		20	Mode de correction non valide (position des unités de <Corr_Mode>). Seules les valeurs 0 et 1 sont autorisées.
		21	Mode de correction non valide (position des dizaines et/ou des centaines de <Corr_Mode>). Lors de l'écriture d'une direction d'axe, seule la position des unités peut être égale à zéro.
		30	La position des centaines de <Corr_Mode> n'est pas valide. Seules les valeurs 0 et 1 sont autorisées.
		31	La position des milliers de <Corr_Mode> n'est pas valide. Seules les valeurs 0 et 1 sont autorisées.
		40	Le vecteur directionnel devant être appliqué comme direction de l'axe est un vecteur nul. Cette erreur ne peut être signalée que si la position des milliers de <Corr_Mode> est égale à 0. Si la position des milliers de ce paramètre est égale à 1 (surveillance de la correction maximale désactivée), un vecteur nul peut également être écrit.
		41	Lors de la correction d'un vecteur d'offset, l'écart dans au moins une coordonnée par rapport à la valeur actuelle est supérieur à la valeur maximale spécifiée par la donnée de réglage SD41610 $SN_CORR_TRAFO_LIN_MAX. Le paramètre <Corr_Vect> est alors écrasé par un vecteur d'erreur. Ceci s'applique également lorsque l'exécution est annulée avec une alarme (voir paramètre <No_Alarm>). Les composantes du vecteur d'erreur, dont la valeur de correction a dépassé la limite autorisée, contiennent la différence

			(avec le signe correct) entre la valeur de correction déterminée et la limite. Le contenu des composantes n'ayant pas dépassé leur limite est égal à zéro.
		42	Lors de la correction d'un vecteur directionnel, l'écart angulaire par rapport à la direction actuelle est supérieur à la valeur maximale spécifiée par la donnée de réglage SD41611 $SN_CORR_TRAFO_DIR_MAX.
		43	La tentative d'écrire une variable système a été refusée pour cause de droits d'écriture insuffisants.
<Corr_Vect> :	Vecteur de correction Le contenu du vecteur de correction est défini par les paramètres <Corr_Index> et <Corr_Mode> suivants. Si <Corr_Status> = 41, le contenu du vecteur est écrasé (voir ci-dessus).		
	Type de données :	REAL	
<Corr_Index> :	Section dont l'élément de correction doit être modifié / indice de l'axe d'orientation dont le vecteur directionnel doit être modifié		
	Type de données :	INT	
	Le paramètre <Corr_Index> est codé en décimal (position des unités et dizaines) :		
	Position des unités :	Contient l'indice de la section ou de l'axe d'orientation dans la chaîne partielle.	
	Position des dizaines :	Identifie la chaîne partielle.	
		0x	Chaîne pièce
		1x	Chaîne outil

<Corr_Mode> :	Mode de correction		
	Type de données :	INT	
	Le paramètre <Corr_Mode> est codé en décimal (position des unités jusqu'à la position des milliers) :		
	Position des unités :	Définit l'élément à corriger.	
		xxx**0**	Correction d'un vecteur d'offset linéaire
		xxx**1**	Correction d'un vecteur directionnel d'un axe d'orientation
	Position des dizaines :	Définit la façon dont l'élément de correction référencé par le contenu de <Corr_Index> doit être modifié.	
		xx**0**x	Le vecteur de correction est directement écrit dans l'élément de correction. Cette variante peut être utilisée pour écrire directement l'élément de correction sans que l'indice <n> de la donnée système ($NK_OFF_DIR[<n>, ...]) soit nécessairement connu.
		xx**1**x	Comme 0, avec la différence que la valeur de correction transmise est interprétée en coordonnées mondiales. Une différence entre les variantes 0 et 1 peut apparaître lorsque la chaîne cinématique contient d'autres rotations en position de base (position de tous les axes d'orientation égale à 0).
		xx**2**x	Comme 1, avec la différence que la valeur de correction s'applique sur la totalité de la section. Autrement dit, la valeur renseignée dans l'élément de correction est telle que la totalité de la section atteint la longueur définie par la valeur de correction.
		Remarque : Les valeurs 1 et 2 ne sont pas autorisées pour l'écriture du vecteur directionnel d'un axe d'orientation.	
	Position des centaines :	Définit la façon dont le contenu du paramètre <Corr_Vect> doit être interprété.	
		x**0**xx	Le vecteur de correction transmis <Corr_Vect> contient la totalité de la nouvelle longueur de l'élément de correction ou de la section référencée par <Corr_Index> en association avec la position des dizaines de <Corr_Mode> (correction absolue).
		x**1**xx	Le vecteur de correction transmis <Corr_Vect> ne contient que la différence par rapport à la longueur actuelle de l'élément de correction ou de la section référencée par <Corr_Index> en association avec la position des dizaines de <Corr_Mode> (correction incrémentale).
		Remarque : Pour la correction du vecteur directionnel d'un axe d'orientation, le contenu de la position des centaines doit être égal à 0.	
	Position des milles :	Définit si la correction doit être limitée par la valeur maximale suivante : • SD41610 $SN_CORR_TRAFO_LIN_MAX ou • SD41611 $SN_CORR_TRAFO_DIR_MAX	
		0xxx	La surveillance de la correction maximale est active.

		1xxx	La surveillance de la correction maximale n'est pas active.
<No_Alarm> :	Comportement en cas d'erreur (valeur retournée > 0) (**facultatif**)		
	Type de données :	BOOL	
	Valeur :	FALSE (Default)	En cas d'erreur, l'exécution du programme est abandonnée et l'alarme 14103 est émise.
		TRUE	En cas d'erreur, l'exécution du programme n'est pas arrêtée et aucune alarme n'est affichée. Cas d'application : réaction personnalisée en fonction de la valeur de retour

Remarque

Si une erreur se produit lors de l'appel de la fonction, une alarme est émise ou un numéro d'erreur est retourné (voir paramètre <No_Alarm>), permettant à l'utilisateur de réagir de manière appropriée à l'état d'erreur. La cause de l'erreur est définie plus en détail par un paramètre d'alarme. Un numéro d'erreur retourné au lieu d'une alarme est identique au paramètre d'alarme.

Plus d'informations sur CORRTRAFO

La structure cinématique d'une machine avec transformation d'orientation est décrite par une ou deux chaînes cinématiques (chaînes partielles) qui partent de l'origine du système de coordonnées mondiales. L'une des deux chaînes, la **chaîne outil**, se termine au point de référence de l'outil, l'autre, la **chaîne pièce** dans l'origine du système de coordonnées de base.

La fonction CORRTRAFO écrit les longueurs de bras de levier et les directions d'axe des machines comportant une transformation d'orientation dans des éléments de correction spéciaux. Une chaîne cinématique est décrite entre autres par des éléments de type OFFSET définis avec $NK_TYPE.

CORRTRAFO utilise des sections

Les deux chaînes partielles peuvent être divisées chacune en quatre sections au maximum :

- La section 1 commence au point de départ de la chaîne et se termine au premier axe d'orientation.
- La section 2 est la section comprise entre l'axe d'orientation 1 et l'axe d'orientation 2.
- La section 3 est la section comprise entre l'axe d'orientation 2 et l'axe d'orientation 3.
- La section 4 est la section comprise entre l'axe d'orientation 3 et la fin de la chaîne outil ou de la chaîne pièce.

Les sections peuvent contenir chacune des éléments de chaîne constants de type OFFSET ou ROT_CONST.

Le graphique suivant présente une transformation d'orientation avec 2 axes d'orientation.

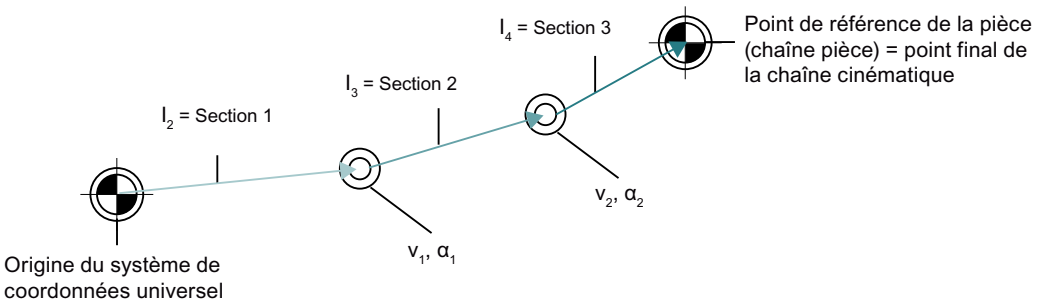


Figure 3-7 Exemple CORRTRAFO

Les sections sont définies de manière univoque : Lorsqu'on parcourt la chaîne cinématique partielle depuis son point de départ jusqu'à la fin, la première section comporte l'indice 0, la suivante l'indice 1, etc. L'indice de la dernière section est par conséquent toujours égal au nombre d'axes d'orientation.

Éléments de correction

Dans chacune de ces sections, la variable système $NT_CORR_ELEM_T[<n>, 0 ... 3] ou $NT_CORR_ELEM_P[<n>, 0 ... 3] permet de référencer un élément constant de chaîne cinématique du type $NK_TYPE[<n>] = "OFFSET". Les valeurs de correction déterminées lors du mesurage de la machine sont écrites avec la fonction CORRTRAFO dans ces éléments ainsi désignés.

Exemple avec indice de transformation = 1 :

- $NT_CORR_ELEM_T[1,0] = "C_AXIS_OFFSET" ; Le décalage de l'axe C (axe d'orientation 1), dans la section 1, est défini comme élément de correction.
- $NT_CORR_ELEM_T[1,1] = "B_AXIS_OFFSET" ; Le décalage de l'axe C (axe d'orientation 2), dans la section 2, est défini comme élément de correction.
- $NT_CORR_ELEM_T[1,2] = "BASE_TOOL_OFFSET"; Le décalage de l'axe B par rapport au point de référence d'outil, dans la section 3, est défini comme élément de correction.

L'ordre des références dans $NT_CORR_ELEM_T/P[<n>, 0 ... 3] doit correspondre aux sections susmentionnées. Autrement dit, $NT_CORR_ELEM_T/P [<n>, 0] ne peut contenir qu'un élément de chaîne se trouvant avant le premier axe d'orientation, etc.

Dans les éléments de correction ainsi définis, la fonction CORRTRAFO écrit les valeurs déterminées par le mesurage de la machine. La modification des valeurs de correction est définie dans CORRTRAFO par le paramètre <Corr_Mode>.

Fermeture de la chaîne

Lorsque le bit 7 ou le bit 8 sont mis à 1 dans la variable système $NT_CNTRL[<n>], des éléments de chaîne constants sont insérés automatiquement en interne à la fin de la chaîne pièce (bit 7) ou avant le point de départ de la chaîne outil (bit 8). Ces éléments forment une connexion entre la fin de la chaîne et l'origine machine ("Fermeture de la chaîne").

Ces éléments insérés automatiquement ne sont pas accessibles en écriture externe mais uniquement en lecture (voir également les variables système $AC_TRAFO_CORR_ELEM_P/T).

Point de fermeture de la chaîne outil

Si la variable système $NT_CLOSE_CHAIN_T n'est pas vide, la chaîne outil ne sera pas fermée à la fin de la chaîne, mais plutôt à la fin de l'élément de chaîne spécifié. D'autres éléments de chaîne qui se trouvent après ce point provoquent un décalage d'origine correspondant lorsque la transformation est activée.

Indice d'un axe d'orientation

Outre les offsets constants entre les axes d'orientation, la fonction CORRTRAFO permet également d'écrire les vecteurs directionnels des axes d'orientation. L'indice d'un axe d'orientation désigne dans ce cas l'indice résultant si l'on parcourt la chaîne cinématique partielle depuis l'origine jusqu'à la fin, en commençant le comptage à zéro. L'indice d'un axe d'orientation est alors toujours égal à l'indice de la section précédente.

L'indice d'un axe d'orientation peut également être déterminé à l'aide de la variable système $AC_TRAFO_ORIAX_LOC.

Modification maximale admissible d'un élément de chaîne

La modification maximale admissible d'un élément de chaîne peut être limitée à l'aide des deux données de réglage SD41610 $SN_CORR_TRAFO_LIN_MAX pour les vecteurs d'offset et SD41611 $SN_CORR_TRAFO_DIR_MAX pour les vecteurs directionnels des axes d'orientation. SD41610 $SN_CORR_TRAFO_LIN_MAX indique la valeur de modification maximale autorisée de chaque composante de vecteur par rapport à la valeur de référence. SD41611 $SN_CORR_TRAFO_DIR_MAX indique l'angle maximal autorisé de modification de la direction de l'axe du vecteur par rapport à la valeur de référence. La valeur de référence est toujours la valeur correspondante qui est effective lors de l'appel des transformations actives de CORRTRAFO. En d'autres termes, les contenus des données de cinématique, éventuellement modifiées dans la gestion des données après l'activation de la transformation, n'ont aucune incidence sur les effets de la fonction CORRTRAFO.

3.13 Corrections d'outils

3.13.1 Mémoire de correcteurs

Structure de la mémoire de correcteurs

Chaque tableau de données peut être appelé à l'aide d'un numéro T et d'un numéro D et contient, outre les données géométriques de l'outil, d'autres indications telles que le type d'outil.

Données de tranchant utilisateur

Des données utilisateur de tranchant peuvent être configurées à l'aide de PM. Veuillez tenir compte des indications du constructeur de machine.

Paramètres d'outil

Remarque

Les différentes valeurs de la mémoire de correcteurs

Les différentes valeurs de la mémoires de correcteurs P1 à P25 peuvent être lues et écrites par le programme grâce à des variables système. Tous les autres paramètres sont réservés.

Les paramètres d'outil $TC_DP6 à $TC_DP8, $TC_DP10 et $TC_DP11 ainsi que $TC_DP15 à $TC_DP17, $TC_DP19 et $TC_DP20 possèdent, en fonction du type d'outil, une autre signification.

Paramètre d'outil numéro (DP)	Signification des variables système	Remarque
$TC_DP1	Type d'outil	Voir liste ci-dessous
$TC_DP2	Position du tranchant	Uniquement outils de tournage
Géométrie	**Correction de longueur**	
$TC_DP3	Longueur 1	Prise en compte selon
$TC_DP4	Longueur 2	Type et niveau
$TC_DP5	Longueur 3	
Géométrie	**Rayon**	
$TC_DP6 [1)] $TC_DP6 [2)]	Rayon 1 / Longueur 1 Diamètre d	Fraise/outil indexable/meule Scie à rainurer
$TC_DP7 [1)] $TC_DP7 [2)]	Longueur 2 / Rayon de congé de fraise conique Largeur de rainure b rayon de congé	Fraises Scie à rainurer
$TC_DP8 [1)] $TC_DP8 [2)]	Rayon d'arrondi 1 pour fraises Projection k	Fraises Scie à rainurer
$TC_DP9 [1) 3)]	Rayon d'arrondi 2	Réservé
$TC_DP10 [1)]	Angle 1 face frontale de l'outil	Fraises coniques
$TC_DP11 [1)]	Angle 2 axe longitudinal de l'outil	Fraises coniques
Usure	**Correction de longueur et de rayon**	

Paramètre d'outil numéro (DP)	Signification des variables système	Remarque
$TC_DP12	Longueur 1	
$TC_DP13	Longueur 2	
$TC_DP14	Longueur 3	
$TC_DP15[1)] $TC_DP15[2)]	Rayon 1 / Longueur 1 Diamètre d	Fraise/outil indexable/meule Scie à rainurer
$TC_DP16[1)] $TC_DP16[3)]	Longueur 2 / Rayon de congé de fraise conique Largeur de rainure b rayon de congé	Fraises Scie à rainurer
$TC_DP17[1)] $TC_DP17[2)]	Rayon d'arrondi 1 pour fraises Projection k	Fraises / fraises avant 3D Scie à rainurer
$TC_DP18[1) 3)]	Rayon d'arrondi 2	Réservé
$TC_DP19[1)]	Angle 1 face frontale de l'outil	Fraises coniques
$TC_DP20[1)]	Angle 2 axe longitudinal de l'outil	Fraises coniques
Cote de base/adaptateur	**Corrections de longueur**	
$TC_DP21	Longueur 1	
$TC_DP22	Longueur 2	
$TC_DP23	Longueur 3	
Technologie		
$TC_DP24	Angle de dépouille	Uniquement outils de tournage
$TC_DP25		Réservé

1) Également valable pour les fraises destinées au fraisage en bout 3D

2) Pour le type d'outil scie à rainurer

3) Réservé

Remarques

Les grandeurs géométriques (par exemple longueur 1 ou rayon) sont constituées de plusieurs composantes. Ces dernières sont additionnées pour donner une grandeur résultante (par exemple la longueur totale 1, le rayon total) qui est alors prise en compte pour la correction.

La valeur zéro est affectée aux corrections inutiles.

Paramètres d'outil $TC-DP1 à $TC-DP23 avec outils de contour

Remarque

Les paramètres d'outil qui ne figurent pas dans le tableau, par exemple $TC_DP7, ne sont pas exploités ; autrement dit, leur contenu n'est pas significatif.

Paramètre d'outil numéro (DP)	Signification	Découper Dn		Remarque
$TC_DP1	Type d'outil			400 à 599
$TC_DP2	Position du tranchant			
Géométrie	**Correction de longueur**			

Paramètre d'outil numéro (DP)	Signification	Découper Dn		Remarque
$TC_DP3	Longueur 1			
$TC_DP4	Longueur 2			
$TC_DP5	Longueur 3			
Géométrie	**Rayon**			
$TC_DP6	Rayon			
Géométrie	**Angle limite**			
$TC_DP10	Angle limite minimal			
$TC_DP11	Angle limite maximal			
Usure	**Correction de longueur et de rayon**			
$TC_DP12	Usure Longueur 1			
$TC_DP13	Usure Longueur 2			
$TC_DP14	Usure Longueur 3			
$TC_DP15	Usure rayon			
Usure	**Angle limite**			
$TC_DP19	Usure de l'angle limite min.			
$TC_DP20	Usure de l'angle limite max.			
Cote de base/adaptateur	**Corrections de longueur**			
$TC_DP21	Longueur 1			
$TC_DP22	Longueur 2			
$TC_DP23	Longueur 3			

Valeur de base et valeur d'usure

Les grandeurs résultantes sont la somme de la valeur de base et de la valeur d'usure (par exemple $TC_DP6 + $TC_DP15 pour le rayon). Pour la longueur d'outil du premier tranchant, la cote de base ($TC_DP21 – $TC_DP23) est par ailleurs également additionnée. De plus, toutes les autres grandeurs pouvant influencer la longueur d'outil même pour un outil conventionnel agissent sur cette longueur d'outil (adaptateur, organe porte-outil orientable, données de réglage).

Angle limite 1 et 2

Les angles limites 1 ou 2 se rapportent chacun au vecteur du centre du bec de l'outil au point de référence du tranchant et sont comptés en sens anti-horaire.

3.13.2 Corrections additives

3.13.2.1 Sélection des corrections additives (DL)

Les corrections additives permettent de programmer des compensations qui dépendent du processus d'usinage. Elles se rapportent aux données géométriques d'un tranchant et font donc partie des données de tranchant d'outil.

L'accès aux corrections additives a lieu avec un numéro DL (DL : location dependent ; corrections en fonction du lieu d'utilisation) et leur saisie s'effectue sur l'interface utilisateur.

Application

Les corrections additives permettent de compenser les erreurs de cotes dues au lieu d'utilisation.

Syntaxe

```
DL=<numéro>
```

Signification

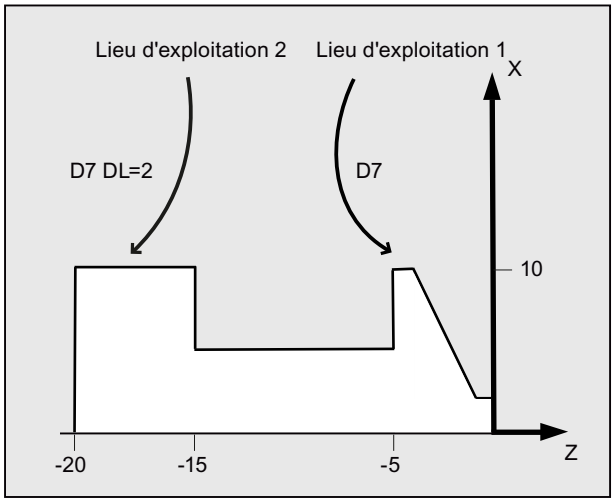
`DL` :	Instruction d'activation d'une correction additive
`<numéro>` :	Le bloc de données de correction additive à activer est indiqué dans le paramètre `<numéro>`.

Remarque

La détermination du nombre de corrections additives et leur activation ont lieu par le biais de paramètres machine (→ voir les indications du constructeur de machine).

Exemple

Le même tranchant est utilisé pour usiner 2 portées de roulement :



Code de programme	Commentaire
`N110 T7 D7`	`; La tourelle revolver est positionnée à l'emplacement 7. D7 et DL=1 sont activés et les corrections correspondantes sont exécutées dans le bloc suivant.`
`N120 G0 X10 Z1`	
`N130 G1 Z-6`	
`N140 G0 DL=2 Z-14`	`; DL=2 est activé en supplément de D7 et la correction correspondante est exécutée dans le bloc suivant.`
`N150 G1 Z-21`	
`N160 G0 X200 Z200`	`; Accostage de la position de changement d'outil.`

Code de programme	**Commentaire**
...	

3.13.2.2 Définition des valeurs d'usure et de réglage ($TC_SCPxy[t,d], $TC_ECPxy[t,d])

La lecture et l'écriture des valeurs d'usure et de réglage s'effectuent à l'aide de variables système. La structure des nouveaux paramètres système est analogue à celle des variables système correspondants pour les outils et les tranchants.

Variables système

`$TC_SCPxy[<t>,<d>]` :	Valeur d'usure associée au paramètre géométrique correspondant par le biais de xy, x étant le numéro de la valeur d'usure et y le numéro du paramètre géométrique.
`$TC_ECPxy[<t>,<d>]` :	Valeur de réglage associée au paramètre géométrique correspondant par le biais de xy, x étant le numéro de la valeur de réglage et y le numéro du paramètre géométrique.
`<t>` : numéro T de l'outil `<d>` : numéro D du tranchant de l'outil	

Remarque

Les valeurs d'usure et de réglage sont ajoutées aux valeurs des paramètres géométriques et des autres paramètres de correction (numéro D).

Exemple

La valeur d'usure associée à la longueur 1 est fixée à 1.0 pour le tranchant <d> de l'outil <t>.

Paramètre : $TC_DP**3** (longueur 1, pour outils de tournage)

Valeurs d'usure : $TC_SCP1**3** à $TC_SCP6**3**

Valeurs de réglage : $TC_ECP1**3** à $TC_ECP6**3**

$TC_SCP4**3** [<t>,<d>] = 1.0

3.13.2.3 Effacer les corrections additives (DELDL)

L'instruction `DELDL` permet d'effacer les corrections additives associées à des tranchants d'outil (libération d'espace mémoire). L'effacement concerne les valeurs d'usure aussi bien que les valeurs de réglage.

Syntaxe

```
DELDL[<t>,<d>]
DELDL[<t>]
DELDL
<Etat>=DELDL[<t>,<d>]
```

Signification

DELDL :	Instruction de suppression des corrections additives	
<t> :	Numéro T de l'outil	
<d> :	Numéro D du tranchant de l'outil	
DELDL[<t>,<d>] :	Suppression toutes les corrections additives du tranchant <d> de l'outil <t>.	
DELDL[<t>] :	Suppression de toutes les corrections additives de tous les tranchants de l'outil <t>.	
DELDL :	Suppression de toutes les corrections additives de tous les tranchants de tous les outils de l'unité TO (pour le canal dans lequel l'instruction est programmée).	
<état> :	Etat de suppression	
	Valeur :	Signification :
	0	Effacement exécuté correctement.
	-	L'effacement n'a pas eu lieu (lorsque le paramétrage désigne un tranchant précis) ou l'effacement n'a pas été entièrement effectué (lorsque le paramétrage désigne plusieurs tranchants).

Remarque

Les valeurs d'usure et de réglage d'outils actifs ne peuvent pas être effacées (par analogie avec l'effacement de correcteurs D et de données d'outil).

3.13.3 Correcteur d'outil : Interventions spéciales

Les données de réglage SD42900 à SD42960 permettent de gérer l'exploitation du signe pour la longueur d'outil et l'usure.

Ceci concerne également le comportement des composantes d'usure lors de l'application de la fonction miroir à des axes géométriques ou en cas de changement de plan de travail, mais aussi pour la compensation de température dans la direction d'outil.

Valeurs d'usure

Lorsque nous faisons référence à des valeurs d'usure dans ce qui suit, il s'agit de la somme des valeurs d'usure proprement dites ($TC_DP12 à $TC_DP20) et des corrections totales qui comprennent les valeurs d'usure ($SCPX3 à $SCPX11) et les valeurs de réglage ($ECPX3 à $ECPX11).

Des informations complémentaires sur les corrections totales figurent dans la description fonctionnelle Gestion des outils.

Données de réglage

SD42900 $SC_MIRROR_TOOL_LENGTH	Fonction miroir appliquée aux composantes de longueur d'outil et de cote de base.
SD42910 $SC_MIRROR_TOOL_WEAR	Fonction miroir appliquée aux valeurs d'usure pour les composantes de longueur d'outil.

SD42920 $SC_WEAR_SIGN_CUTPOS	Exploitation du signe des composantes d'usure en fonction de la position du tranchant.
SD42930 $SC_WEAR_SIGN	Inverse le signe des cotes d'usure.
SD42935 $SC_WEAR_TRANSFORM	Transformation des valeurs d'usure.
SD42940 $SC_TOOL_LENGTH_CONST	Affectation des composantes de longueur d'outil aux axes géométriques.
SD42950 $SC_TOOL_LENGTH_TYPE	Affectation des composantes de longueur d'outil indépendamment du type d'outil-
SD42960 $SC_TOOL_TEMP_COMP	Valeur de compensation de la température dans la direction de l'outil. Est également active pour l'orientation de l'outil actuelle.

Informations complémentaires

Prise d'effet des données de réglage modifiées

En cas de modification des données de réglage décrites, les composantes du correcteur d'outil ne sont actualisées que lors de la prochaine sélection du tranchant d'outil. Dans le cas où un outil est déjà actif et où les données de correction modifiées de cet outil doivent prendre effet, il faut appeler à nouveau cet outil.

Ceci est également valable si la longueur résultante de l'outil est modifiée du fait de l'application de la fonction miroir à un axe. L'outil doit faire l'objet d'une nouvelle sélection après la fonction miroir afin que les composantes modifiées de longueur d'outil prennent effet.

Organes porte-outil orientables et nouvelles données de réglage

Les données de réglage SD42900 à SD42940 n'ont pas d'effet sur les composantes d'un organe porte-outil orientable éventuellement actif. En présence d'un organe porte-outil orientable, l'outil est toujours pris en considération avec sa longueur totale résultante (longueur d'outil + usure + cote de base). Lors du calcul de cette longueur totale résultante, la commande tient compte de toutes les modifications provoquées par des données de réglage c'est-à-dire les vecteurs d'un organe porte-outil orientable sont donc indépendants du plan d'usinage.

Remarque

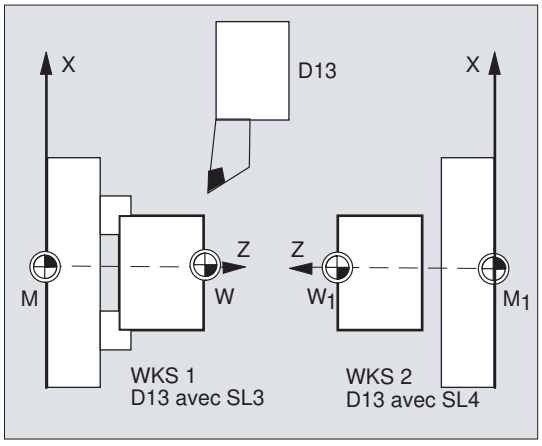
En cas de mise en œuvre d'organes porte-outil orientables, il sera souvent préférable de définir tous les outils pour un système de base non inversé par fonction miroir, même ceux utilisés uniquement pour un usinage symétrique. Pour l'usinage avec des axes inversés, l'organe porte-outil sera pivoté de telle sorte que la position effective de l'outil soit décrite correctement. Toutes les composantes de longueur d'outil agissent alors automatiquement dans le sens correct, de sorte que l'interprétation de certaines composantes par les données de réglage en fonction de l'état d'inversion d'axes est superflue.

Autres possibilités d'applications

L'utilisation de la fonctionnalité "organe porte-outil orientable" peut aussi s'avérer intéressante si la machine ne permet pas d'orienter physiquement des outils, mais dispose d'outils ayant des orientations diverses et installés à demeure. La cotation des outils peut alors être effectuée uniformément avec une orientation de base, les cotes valables pour l'usinage s'obtenant par rotation d'un organe porte-outil virtuel.

3.13.3.1 Application de la fonction miroir aux longueurs d'outil

Avec des données de réglage SD42900 $SC_MIRROR_TOOL_LENGTH et SD42910 $SC_MIRROR_TOOL_WEAR différentes de zéro, il est possible d'appliquer la fonction miroir aux composantes de longueur d'outil et de cote de base avec les valeurs d'usure par rapport à leurs axes respectifs.



SD42900 $SC_MIRROR_TOOL_LENGTH

Donnée de réglage **différente** de zéro :

Les composantes de longueur d'outil ($TC_DP3, $TC_DP4 et $TC_DP5) et les composantes de cote de base ($TC_DP21, $TC_DP22 et $TC_DP23), dont les axes associés font l'objet de la fonction miroir, sont aussi soumises à la fonction miroir - par inversion de leur signe.

La fonction miroir n'est **pas** appliquée aux valeurs d'usure. Si cela doit être le cas, vous devez régler de façon adéquate la donnée de réglage SD42910 $SC_MIRROR_TOOL_WEAR.

SD42910 $SC_MIRROR_TOOL_WEAR

Donnée de réglage **différente** de zéro :

Les valeurs d'usure pour les composantes de longueur d'outil, dont les axes associés font l'objet de la fonction miroir, sont aussi soumises à la fonction miroir - par inversion de leur signe.

3.13.3.2 Exploitation du signe de l'usure

Avec les données de réglage SD42920 $SC_WEAR_SIGN_CUTPOS et SD42930 $SC_WEAR_SIGN différentes de zéro, il est possible d'inverser l'exploitation du signe des composantes d'usure.

SD42920 $SC_WEAR_SIGN_CUTPOS

Donnée de réglage **différente** de zéro :

Dans le cas des outils à position de tranchant définie (outils de tournage et de rectification – type d'outil 400), l'exploitation du signe des composantes d'usure dans le plan de travail dépend de la position du tranchant. Dans le cas des outils sans position de tranchant définie, cette donnée de réglage n'est pas prise en considération.

Dans le tableau suivant, les cotes dont le signe est inversé par le SD42920 (différent de 0) sont repérées par X :

Position du tranchant	Longueur 1	Longueur 2
1		
2		X
3	X	X
4	X	
5		
6		
7		X
8	X	
9		

Remarque

Les modifications de signe en fonction des SD42920 et SD42910 sont indépendantes. Si le signe d'une cote est modifié par les deux données de réglage par exemple, il restera inchangé.

SD42930 $SC_WEAR_SIGN

Donnée de réglage **différente** de zéro :

Le signe de toutes les cotes d'usure est inversé. Cette donnée agit aussi bien sur la longueur d'outil que sur les autres dimensions, par exemple le rayon de l'outil, le rayon d'arrondi, etc.

S une cote d'usure positive est entrée, l'outil devient "plus court" et "plus fin". A ce sujet, voir chapitre "Correcteur d'outil : Interventions spéciales, Prise d'effet des données de réglage modifiées".

3.13.3.3 Système de coordonnées de l'usinage actif (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)

En fonction de la cinématique de la machine ou de la présence d'un organe porte-outil orientable, les valeurs d'usure mesurées dans l'un de ces systèmes de coordonnées sont transposées voire transformées dans un système de coordonnées adapté.

Systèmes de coordonnées de l'usinage actif

Des offsets de longueur d'outil peuvent résulter des systèmes de coordonnées suivants et prendre en compte les composantes d'usure de longueur d'outil avec l'instruction G correspondante du groupe 56 dans un outil actif :

- Système de coordonnées machine (SCM)
- Système de coordonnées de base (SCB)
- Système de coordonnées pièce (SCP)
- Système de coordonnées de l'outil (SCO)
- Système de coordonnées de l'outil de transformation cinématique (SCT)

Syntaxe

```
TOWSTD
TOWMCS
TOWWCS
TOWBCS
TOWTCS
TOWKCS
```

Signification

TOWSTD :	Valeur de position de base pour corrections sur la longueur d'outil de la valeur d'usure
TOWMCS :	Corrections dans la longueur d'outil dans SCM
TOWWCS :	Corrections dans la longueur d'outil dans SCP
TOWBCS :	Corrections dans la longueur d'outil dans SCB
TOWTCS :	Corrections de la longueur d'outil sur le point de référence de l'organe porte-outil (organe porte-outil orientable)
TOWKCS :	Corrections de la longueur d'outil de la tête d'outil (transformation cinétique)

Informations complémentaires

Différences

Le tableau suivant représente les principales différences :

Instruction G	Valeur d'usure	Organe porte-outil orientable actif
TOWSTD	Valeur de position initiale, longueur d'outil	Les valeurs d'usure sont assujetties à la rotation.
TOWMCS	Valeur d'usure dans SCM. TOWMCS est identique à TOWSTD lorsque aucun organe porte-outil orientable n'est actif.	Seul le vecteur de la longueur d'outil résultante tourne, sans prise en compte de l'usure.
TOWWCS	La valeur d'usure est convertie du SCP au SCM.	Le vecteur d'outil est calculé sans prise en compte de l'usure comme pour TOWMCS.
TOWBCS	La valeur d'usure est convertie dans SCB au SCM.	Le vecteur d'outil est calculé sans prise en compte de l'usure comme pour TOWMCS.
TOWTCS	La valeur d'usure est convertie dans le système de coordonnées de l'outil au SCM.	Le vecteur d'outil est calculé sans prise en compte de l'usure comme pour TOWMCS.

TOWWCS, TOWBCS, TOWTCS : Le vecteur d'usure est additionné au vecteur d'outil.

Transformation linéaire

La longueur d'outil est uniquement judicieusement définissable dans SCM lorsque le SCM est issu de SCB par une transformation linéaire.

Transformation non linéaire

Si par exemple une transformation non linéaire est active avec TRANSMIT, alors le SCB est automatiquement utilisé lorsque le SCM est indiqué comme système de coordonnées souhaité.

Pas de transformation cinématique et pas d'organe porte-outil orientable

Lorsque ni une transformation cinématique ni un organe porte-outil orientable n'est actif, alors, mis à part le SCP, les quatre autres systèmes de coordonnées coïncident. Le SCP se différencie alors de tous les autres. Du fait que seules les longueurs d'outil doivent être interprétées, les translations entre les systèmes de coordonnées n'ont plus de signification.

Prise en compte des valeurs d'usure

La donnée de réglage **SD42935 $SC_WEAR_TRANSFORM** détermine quelle composante d'usure parmi les trois composantes :

- Usure
- Corrections totales fines
- Corrections totales grossières

d'une rotation doit être assujettie par une transformation d'adaptateur ou par un organe porte-outil orientable lorsque l'une des instructions G suivantes est active :

- TOWSTD
  Position d'initialisation : Pour corrections sur la longueur d'outil
- TOWMCS
  Valeurs d'usure dans le système de coordonnées machine (SCM).
- TOWWCS
  Valeurs d'usure dans le système de coordonnées pièce (SCP)
- TOWBCS
  Valeurs d'usure dans le système de coordonnées de base (SCB).

- TOWTCS
  Valeurs d'usure dans le système de coordonnées d'outil sur le mandrin du porte-outil (référence de l'organe porte-outil T)
- TOWKCS
  Valeurs d'usure dans le système de coordonnées de la tête d'outil avec transformation cinétique.

Remarque

L'évaluation de chaque composante d'usure (affectation aux axes géométriques, exploitation du signe) est influencée par les facteurs suivants :

- Niveau actif
- Transformation d'adaptateur
- Données de réglage :
  - SD42910 $SC_MIRROR_TOOL_WEAR
  - SD42920 $SC_WEAR_SIGN_CUTPOS
  - SD42930 $SC_WEAR_SIGN
  - SD42940 $SC_TOOL_LENGTH_CONST
  - SD42950 $SC_TOOL_LENGTH_TYPE

3.13.3.4 Longueurs d'outil et changement de plan

Avec les données de réglage SD42940 $SC_TOOL_LENGTH_CONST différentes de zéro, il est possible d'affecter les composantes de longueur d'outil telles que la longueur, l'usure et la cote de base aux axes géométriques pour les outils de tournage et de rectification lors d'un changement de plan.

SD42940 $SC_TOOL_LENGTH_CONST

Donnée de réglage **différente** de zéro :

L'affectation des composantes de longueur d'outil (longueur, usure et cote de base) aux axes géométriques n'est pas modifiée en cas de changement de plan de travail (`G17` à `G19`).

Le tableau suivant indique l'affectation des composantes de longueur d'outil aux axes géométriques pour des outils de tournage et de rectification (types 400 à 599) :

Contenu	Longueur 1	Longueur 2	Longueur 3
17	Y	X	Z
*)	X	Z	Y
19	Z	Y	X
-17	X	Y	Z
-18	Z	X	Y
-19	Y	Z	X

*) Toute valeur différente de 0, qui n'est pas égale à une des six valeurs mentionnées, est interprétée comme la valeur 18.

Le tableau suivant indique l'affectation des composantes de longueur d'outil aux axes géométriques pour tous les autres outils (types < 400 ou > 599) :

Plan de travail	Longueur 1	Longueur 2	Longueur 3
*)	Z	Y	X
18	Y	X	Z
19	X	Z	Y
-17	Z	X	Y
-18	Y	Z	X
-19	X	Y	Z

*) Toute valeur différente de 0, qui n'est pas égale à une des six valeurs mentionnées, est interprétée comme la valeur 17.

Remarque

Dans ces tableaux, on suppose que les axes géométriques 1 à 3 sont désignés par X, Y, Z. Pour l'affectation d'une correction à un axe, ce n'est pas le descripteur d'axe mais l'ordre des axes qui est déterminant.

3.13.4 Correction d'outil en ligne

3.13.4.1 Définir une fonction polynôme (FCTDEF)

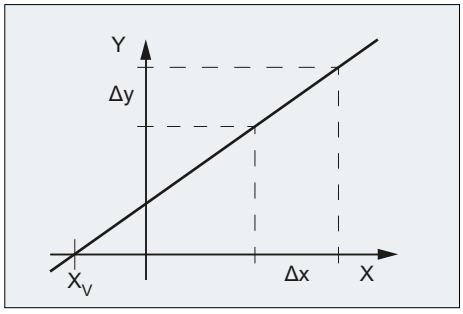
Certaines stratégies de dressage (par ex. molette de dressage) se distinguent par le fait que la meule décroît en continu (de manière linéaire) au niveau du rayon pendant la pénétration de la molette de dressage. A cet effet, une fonction linéaire est nécessaire entre la pénétration de la molette de dressage et l'écriture de la valeur d'usure de la longueur correspondante. La définition de la fonction linéaire s'effectue via la procédure FCTDEF(...) prédéfinie pour les fonctions polynôme jusqu'au 3ème degré maximum.

Equation de droite

$y = f(x) = a_0 + a_1 * x_1$

a_1 : Pente de la droite, avec $a_1 = \Delta x / \Delta y$

a_0 : Décalage de la droite le long de l'axe X, avec $a_0 = -a1 * X_V$



Syntaxe

```
FCTDEF(<Func>,<LLimit>,<ULimit>,<a0>,<a1>,<a2>,<a3>)
```

Signification

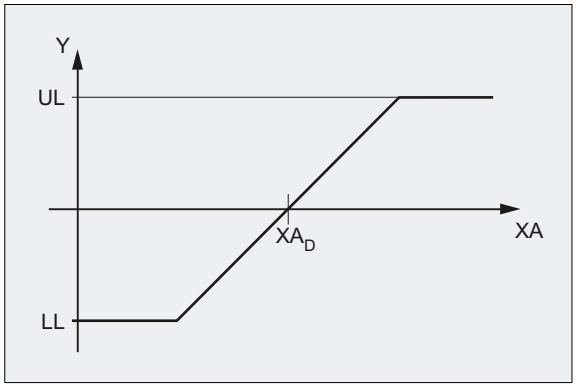
`FCTDEF(...)` :	Définition d'une fonction polynôme pour PUTFTOCF(...) : $y = f(x) = a_0 + a_1*x + a_2*x^2 + a_3*x^3$	
`<Func>` :	numéro de fonction	
	Type de données :	INT
	Plage de valeurs :	1, 2, 3
`<LLimit>` :	Valeur limite inférieure	
	Type de données :	REAL
`<ULimit>` :	Valeur limite supérieure	
	Type de données :	REAL
`<a0>,<a1>,<a2>,<a3>` :	Coefficients de la fonction polynomiale	
	Type de données :	REAL

Exemple

Règles

- Numéro de fonction : 1
- Valeurs limites inférieure et supérieure : -100, 100
- Pente de la caractéristique : $a_1 = 1$
- Le point de fonctionnement doit se situer au milieu de la caractéristique. La caractéristique doit pour cela être décalée dans le sens Y négatif à partir de la position de consigne de l'axe XA dans le SCP au moment de la définition de la fonction dans le programme CN. $a_0 = -a_1 * XA_D = -1 * \$AA\_IW$
- $a_2 = a_3 = 0$

Caractéristique



UL Valeur limite supérieure

LL Valeur limite inférieure

XA_D Consigne de l'axe XA au moment de la définition de la fonction dans le programme CN

Programmation

Code de programme	Commentaire
`FCTDEF(1,-100,100,-$AA_IW[XA],1)`	`; Définition de la fonction`

3.13.4.2 Ecriture en continu de la correction d'outil en ligne (PUTFTOCF)

La procédure PUTFTOCF(...) prédéfinie permet d'effectuer une correction d'outil en ligne via une fonction polynomiale définie au préalable avec FCTDEF(...) (Page 768).

Remarque

La correction d'outil en ligne peut aussi être réalisée par le biais d'une action synchrone.

Pour plus d'informations, voir Description fonctionnelle Actions synchrones.

Syntaxe

```
PUTFTOCF(<Func>,<RefVal>,<ToolPar>,<Chan>,<Sp>)
```

Signification

`PUTFTOCF(...)` :	Ecriture continue modale de la correction d'outil en ligne à l'aide de la fonction polynomiale définie avec FCTDEF(...)	
`<Func>` :	Numéro de fonction, spécifié par la définition de fonction avec FCTDEF(...)	
	Type de données :	INT
	Plage de valeurs :	1, 2, 3

<RefVal> :	Valeur de référence à partir de laquelle la correction doit être déduite (p. ex. consigne d'un axe)	
	Type de données :	VAR REAL
<ToolPar> :	Numéro du paramètre d'usure (longueur 1, 2 ou 3) dans lequel la valeur de correction doit être prise en compte	
	Type de données :	INT
<Chan> :	Numéro du canal dans lequel la correction d'outil en ligne doit être effective **Remarque** : Uniquement requis lorsque la correction ne doit pas être effective dans le canal actif.	
	Type de données :	INT
<Sp> :	Numéro de la broche pour laquelle la correction d'outil en ligne doit être effective **Remarque :** Uniquement requis lorsqu'il s'agit de corriger une meule non active à la place de l'outil actif mis en œuvre.	
	Type de données :	INT

3.13.4.3 Ecriture discrète de la correction d'outil en ligne (PUTFTOC)

Fonction

La procédure PUTFTOC(...) prédéfinie permet d'effectuer une correction d'outil en ligne avec une valeur de correction fixe.

Syntaxe

```
PUTFTOC(<CorrVal>,<ToolPar>,<Chan>,<Sp>)
```

Signification

PUTFTOC(...) :	Ecriture de la correction d'outil en ligne	
<CorrVal> :	Valeur de correction ajoutée au paramètre d'usure	
	Type de données :	REAL
<ToolPar> :	Numéro du paramètre d'usure (longueur 1, 2 ou 3) dans lequel la valeur de correction doit être prise en compte	
	Type de données :	INT
<Chan> :	Numéro du canal dans lequel la correction d'outil en ligne doit être effective **Remarque :** Uniquement requis lorsque la correction ne doit pas être effective dans le canal actif.	
	Type de données :	INT

<Sp> :	Numéro de la broche pour laquelle la correction d'outil en ligne doit être effective **Remarque :** Uniquement requis lorsqu'il s'agit de corriger une meule non active à la place de l'outil actif mis en œuvre.	
	Type de données :	INT

3.13.4.4 Activer/désactiver la correction d'outil en ligne (FTOCON/FTOCOF)

Les instructions G FTOCON et FTOCOF permettent d'activer ou de désactiver la correction d'outil en ligne.

Syntaxe

```
FTOCON
...
FTOCOF
```

Signification

FTOCON :	Activer la correction d'outil en ligne L'instruction doit être programmée dans le canal dans lequel la correction d'outil en ligne doit être activée.
FTOCOF :	Désactiver la correction d'outil en ligne L'instruction doit être programmée dans le canal dans lequel la correction d'outil doit être désactivée. **Remarque :** Avec FTOCOF, la correction d'outil n'est plus appliquée. Cependant, la valeur prise en considération avec PUTFTOC/PUTFTOCF est conservée dans les données de correction spécifiques au tranchant. Pour la désactivation définitive de la correction d'outil en ligne, FTOCOF doit être suivie d'une sélection/désélection de l'outil (T...).

3.13.5 Correction du rayon d'outil 3D

3.13.5.1 Sélection de la correction 3D du rayon d'outil pour le fraisage périphérique 3D (CUT3DC, CUT3DCD, ISD)

La correction 3D du rayon d'outil (CRO 3D) pour le fraisage périphérique 3D, sans prise en compte de surfaces de délimitation, est sélectionnée avec l'instruction G à effet modal CUT3DC ou CUT3DCD.

L'activation proprement dite s'effectue avec G41 ou G42. G40 désactive la correction du rayon d'outil.

Syntaxe

```
G41/G42 ORIC/ORID ISD=... CUT3DC/CUT3DCD CDOF2 X... Y... Z...
...
G40 X... Y... Z...
```

Signification

CUT3DC :	CRO 3D pour le fraisage périphérique (uniquement si la transformation 5 axes est activée).	
CUT3DCD :	CRO 3D rapportée à un outil différentiel pour le fraisage périphérique (uniquement si la transformation 5 axes est activée). La différence de rayon est spécifiée par le paramètre outil $TC_DP15.	
G41/G42 X... Y... Z... :	Activer la correction de rayon d'outil	
	G41 :	Correction du rayon d'outil à gauche du contour
	G42 :	Correction du rayon d'outil à droite du contour
	Remarque : L'activation doit intervenir dans un bloc linéaire (G0/G1).	
CDOF2 :	Désactiver la détection des violations du contour pour le fraisage périphérique 3D	
ORIC/ORID :	Les instructions G ORIC et ORID déterminent le comportement lors de changements d'orientation aux angles saillants.	
	ORIC :	Les changements d'orientation aux angles saillants sont superposés au bloc circulaire à insérer.
	ORID :	Les changements d'orientation aux angles saillants sont exécutés avant le bloc circulaire devant être inséré.
ISD=<Value> :	Lors du fraisage périphérique avec la correction du rayon d'outil 3D **active**, l'adresse ISD (InSertion Depth) permet de modifier la profondeur de pénétration de l'outil.	
	<Value> :	Longueur de la profondeur de pénétration
G40 X... Y... Z... :	Désactiver la correction de rayon d'outil **Remarque :** La désactivation doit intervenir dans un bloc linéaire (G0/G1) avec déplacements d'axes géométriques.	

Remarque

Les instructions G pour la sélection de CRO 3D sont évaluées dans le bloc d'accostage, donc typiquement dans le bloc contenant G41 ou G42.

G41 ou G42 peut également être programmée dans des blocs sans déplacement pour les axes géométriques concernés par la correction. Dans ce cas, le bloc d'accostage est le premier bloc de déplacement après ce type de bloc.

Tout changement de la variante CRO 3D, alors que la correction du rayon d'outil est active, est ignoré sans alarme.

Exemple

Code de programme	**Commentaire**
	; Définition de l'outil D1 :
$TC_DP1[1,1]=120	; Type (fraise à queue)
$TC_DP3[1,1]=20	; Vecteur de correction de longueur
$TC_DP6[1,1]=8	; Rayon
N10 X0 Y0 Z0 T1 D1 F12000	; Sélection de l'outil.
N20 TRAORI(1)	; Activation de la transformation.
N30 G42 ORIC ISD=10 **CUT3DC** G64 X30	; Activation du fraisage périphérique 3D,
	; changement d'orientation en continu au niveau des angles saillants,
	; profondeur de pénétration : 10 mm
N40 ORIWKS A30 B15	; Changement d'orientation au niveau d'un angle en spécifiant les positions d'axe.
N50 Y20 A3=1 C3=1	; Bloc de déplacement avec changement d'orientation,
	; spécification de l'orientation avec vecteur directionnel.
N60 X50 Y30	; Bloc de déplacement avec orientation constante.
N70 Y50 A3=0.5 B3=1 C3=5	; Bloc de déplacement avec changement d'orientation.
N80 M63	; Bloc sans information de déplacement.
N90 X0 ISD=20	; Bloc de déplacement avec changement de la profondeur de pénétration.
N100 G40 Y0	; Désactivation de la correction du rayon d'outil.
N110 M30	

Informations complémentaires

Trajectoire et orientation

La variante de fraisage périphérique utilisée ici est réalisée par spécification d'un contour (ligne directrice) et de l'orientation correspondante. Dans ce type d'usinage, la forme de l'outil n'a aucune importance pour la trajectoire. Seul le rayon est déterminant au point d'attaque de l'outil.

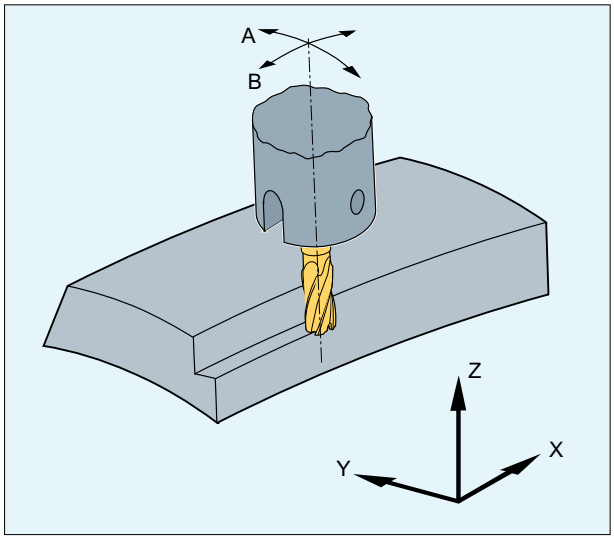


Figure 3-8 Fraisage périphérique

Comportement à l'accostage

Le comportement à l'accostage est toujours NORM pour les variantes 3D de la correction du rayon d'outil.

Comportement aux angles saillants

Lors du fraisage périphérique avec CRO 3D au niveau des angles saillants, les instructions G du groupe 18 (correction d'outil pour comportement aux angles) sont évaluées de manière analogique aux conditions de la CRO 2½D.

- G450 : Arc de raccordement (l'outil contourne les angles de la pièce sur une trajectoire circulaire)
  Contrairement aux solutions de la CRO 2½D, l'élément de contour inséré au niveau d'un angle saillant est toujours un cercle de rayon 0 sur lequel la correction du rayon d'outil agit de la même manière que sur toute autre trajectoire programmée. Il n'est pas possible d'insérer des coniques à la place des cercles. Dans ce cas, l'adresse DISC est sans signification et n'est pas évaluée.
- G451 : Point d'intersection des équidistantes (l'outil s'arrête aux angles de la pièce)
  Le point d'intersection est déterminé en prolongeant les trajectoires décalées des deux blocs concernés dans le plan perpendiculaire à la direction de l'outil au niveau de l'angle.

La méthode du point d'intersection (G451) n'est pas utilisée lorsqu'au moins un bloc contenant un changement de l'orientation de l'outil est inséré entre les blocs de déplacement concernés. Dans un tel cas, c'est toujours un cercle qui est inséré au niveau de l'angle.

Comportement lors de changements d'orientation au niveau des angles saillants

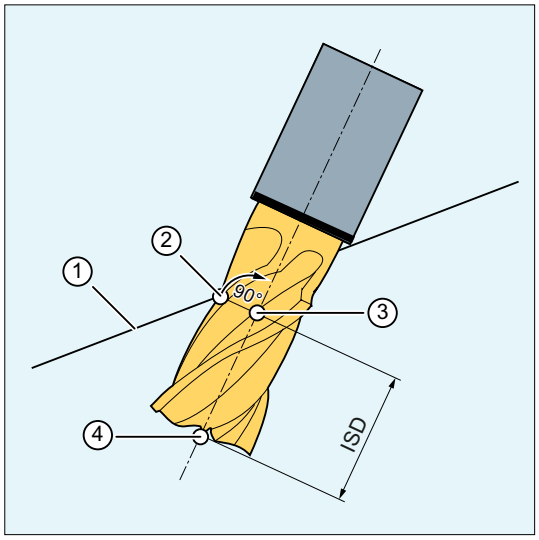
Les instructions G ORIC et ORID permettent de définir si les changements d'orientation qui ont été programmés entre les deux blocs de déplacement formant l'angle seront effectués avant le début du bloc à interpolation circulaire inséré (ORID) ou en même temps que celui-ci (ORIC).

Profondeur de pénétration

La profondeur de pénétration de la fraise correspond à la distance entre le point auxiliaire de la fraise et la pointe de l'outil.

Le point auxiliaire de la fraise est la projection orthogonale du point d'usinage de la fraise sur la trajectoire programmée sur l'axe longitudinal de l'outil.

Ainsi, la profondeur de pénétration règle la position du point d'usinage sur la surface latérale de l'outil.



① Trajectoire programmée
② Point d'usinage de la fraise
③ Point auxiliaire de la fraise
④ Pointe de la fraise
ISD Profondeur de pénétration (InSertion Depth)

Figure 3-9 Profondeur de pénétration

Correction du rayon d'outil par rapport à un outil différentiel

La CRO 3D rapportée à un outil différentiel pour le fraisage périphérique est activée au moyen de l'instruction CUT3DCD. Elle doit être utilisée lorsque le contour programmé se rapporte à la trajectoire du centre d'un outil standard et que l'usinage s'effectue avec un outil différent. Ainsi, lors du calcul de la correction du rayon d'outil 3D, seuls la valeur d'usure du rayon de l'outil actif ($TC_DP15) et les offsets de correction d'outil OFFN et TOFFR/TOFFLR éventuellement programmés sont pris en compte. Le rayon de base ($TC_DP6) de l'outil actif n'est **pas** pris en compte.

Fraisage de poche avec faces latérales obliques pour fraisage périphérique avec CUT3DC

Dans le cas de cette correction de rayon d'outil 3D, un écart du rayon de fraise est compensé par un déplacement perpendiculaire à la surface à usiner. Le plan dans lequel se trouve la surface d'attaque de la fraise reste inchangé lorsque la profondeur de pénétration ISD reste inchangée. Une fraise dotée par exemple d'un rayon plus petit par rapport à un outil standard n'atteindrait pas le fond de la poche, qui représente également la surface de délimitation. Pour permettre une pénétration automatique de l'outil, la commande doit connaître cette surface de délimitation, voir chapitre "Fraisage périphérique 3D avec prise en compte d'une surface de délimitation (CUT3DCC, CUT3DCCD) (Page 783)".

Advanced Surface / Top Surface

Remarque

Pour l'utilisation de la correction du rayon d'outil CUT3DCD en association avec l'option soumise à la licence "Advanced Surface" ou "Top Surface", tenir compte des recommandations de réglages relatives à "Advanced Surface" / "Top Surface".

Pour vérifier les paramètres réglés, un programme de vérification spécial est disponible sur le portail SIOS.

→ Programme de vérification pour Advanced Surface / Top Surface (https://support.industry.siemens.com/cs/ww/fr/view/109738423/en)

3.13.5.2 Sélection de la correction 3D du rayon d'outil pour le fraisage en bout 3D (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD)

La correction 3D du rayon d'outil (CRO 3D) pour le fraisage en bout 3D est sélectionnée avec l'instruction G à effet modal CUT3DF, CUT3DFS, CUT3DFF ou CUT3DFD.

L'activation proprement dite s'effectue avec G41 ou G42.

Pour pouvoir calculer la correction du rayon d'outil en mode fraisage en bout 3D, la définition du vecteur normal à la surface du plan à usiner est nécessaire. Cette définition doit être effectuée avec G41 ou G42 et les adresses A4, B4, C4 et A5, B5, C5.

G40 désactive la correction du rayon d'outil.

Syntaxe

```
G41/G42 ORIC/ORID CUT3DF/CUT3DFS/CUT3DFF/CUT3DFD X... Y... Z... A4=... B4=...
C4=... A5=... B5=... C5=...
...
G40 X... Y... Z...
```

Signification

CUT3DFS :	CRO 3D pour le fraisage en bout avec orientation d'outil constante. L'orientation de l'outil est définie par G17 - G19 et n'est pas influencée par les frames.
CUT3DFF :	CRO 3D pour le fraisage en bout avec orientation d'outil constante. L'orientation de l'outil est la direction définie par G17 - G19 et, le cas échéant, par la composante rotationnelle d'un frame.
CUT3DF :	CRO 3D pour le fraisage en bout avec modification de l'orientation d'outil (uniquement si la transformation 5 axes est activée).

CUT3DFD:	La CRO 3D rapportée à un outil différentiel pour le fraisage en bout avec modification de l'orientation d'outil (uniquement si la transformation 5 axes est activée). La différence de rayon est spécifiée par le paramètre outil $TC_DP15. **Remarque** : CUT3DFD n'est possible qu'en association avec le "lissage des normales à la surface en fraisage en bout 3D". Celui-ci est activé par l'appel de la fonction "Top Surface" (soumise à licence) avec CYCLE832(...).	
G41/G42 X... Y... Z...:	Activer la correction d'outil Le comportement pour G41 et pour G42 est identique en mode fraisage en bout 3D. **Remarque :** L'activation doit intervenir dans un bloc linéaire (G0/G1).	
A4/5=... B4/5=... C4/5=...:	Définition du vecteur normal à la surface du plan à usiner	
	A4=... B4=... C4=...:	Définition en début de bloc
	A5=... B5=... C5=...:	Définition en fin de bloc
ORIC/ORID:	Les instructions G ORIC et ORID déterminent le comportement lors de changements d'orientation aux angles saillants.	
	ORIC:	Les changements d'orientation aux angles saillants sont superposés au bloc circulaire à insérer.
	ORID:	Les changements d'orientation se font avant l'exécution du bloc circulaire inséré.
G40 X... Y... Z...:	Désactiver la correction de rayon d'outil **Remarque :** La désactivation doit intervenir dans un bloc linéaire (G0/G1) avec déplacements d'axes géométriques.	

Remarque

G41 ou G42 peut également être programmée dans des blocs sans déplacement pour les axes géométriques concernés par la correction. Dans ce cas, le bloc d'accostage est le premier bloc de déplacement après ce type de bloc.

Tout changement de la variante CRO 3D, alors que la correction du rayon d'outil est active, est ignoré sans alarme.

Exemples

Exemple 1 : Fraisage en bout 3D avec CUT3DF

```
Code de programme                          Commentaire
N10                                        ; Définition de l'outil D1 :
N20 $TC_DP1[1,1]=121                       ; Type d'outil (fraise convexe demi-cercle)
N30 $TC_DP3[1,1]=20                        ; Correction de longueur
N40 $TC_DP6[1,1]=5                         ; Rayon
```

Code de programme	**Commentaire**
N50 $TC_DP7[1,1]=3	; Rayon d'arrondi
N60	
N70	
N80 X0 Y0 Z0 A0 B0 C0 G17 T1 D1 F12000	; Sélection de l'outil.
N90 TRAORI(1)	; Sélectionner la transformation d'orientation.
N100 B4=-1 C4=1	; Définition du plan.
N110 **G41** ORID **CUT3DF** G64 X10 Y0 Z0	; Activer la correction d'outil.
N120 X30	
N130 Y20 A4=1 C4=1	; Angle saillant, nouvelle définition du plan.
N140 B3=1 C3=5	; Changement d'orientation avec ORID.
N150 B3=1 C3=1	; Changement d'orientation avec ORID.
N160 X-10 A5=1 C5=2 ORIC	
N170 A3=-2 C3=1	; Changement d'orientation avec ORIC
N180 A3=-1 C3=1	; Changement d'orientation avec ORIC
N190 Y-10 A4=-1 C4=3	; Nouvelle définition du plan.
N200 X-20 Y-20 Z10	; Angle rentrant avec le bloc précédent.
N210 X-30 Y10 A4=1 C4=1	; Angle rentrant, nouvelle définition du plan.
N220 A3=1 B3=0.5 C3=1.7	; Changement d'orientation avec ORIC.
N230 X-20 Y30 A4=1 B4=-2 C4=3 ORID	
N240 A3 = 0.5 B3=-0.5 C3=1	; Changement d'orientation.
N250 X0 Y30 C4=1	; Déplacement avec interpolation, nouveau plan, ; orientation avec programmation relative.
N260 BSPLINE X20 Z15	; Début de la courbe de type spline, programmation relative de l'orientation
N270 X30 Y25 Z18	; Reste active pendant la fonction spline.
N280 X40 Y20 Z13	
N290 X45 Y0 PW=2 Z8	
N300 Y-20	
N310 G2 ORIMKS A30 B45 I-20 X25 Y-40 Z0	; Hélice, orientation avec programmation d'axe.
N320 G1 X0 A3=-0.123 B3=0.456 C3=2.789 B4=-1 C4=5 B5=-1 C5=2	; Déplacement avec interpolation, orientation, plan non constant.
N330 X-20 G40	; Désactivation de la correction du rayon d'outil
N340 M30	

Exemple 2 : Programme CN (extrait) avec CUT3DFD, généré depuis un système CAO

Code de programme	**Commentaire**
N01 G710	
N03 T="12"	
N06 S5305 M03	
N07 G642	
; Accostage de la position de départ dans le système de coordonnées machine en fonction de la longueur d'outil.	
G00 G90 X-250.62787 Y-38.37944 A=DC(253.12719) B-12.49543	

Code de programme	**Commentaire**
G00 G90 Z251.80052	
; Fin du positionnement dans le système de coordonnées machine.	
;	
TRAORI(1)	; Sélectionner la transformation d'orientation.
G500	
D1	
CYCLE832(0.01, _TOP_SURFACE_SMOOTH_ON + _ORI_FINISH, 1)	; Appel de CYCLE832 avec : ; tolérance de contour = 0,01 mm, ; type d'usinage : Top Surface avec lissage, ; finition avec saisie d'une tolérance d'orientation, ; tolérance d'orientation = 1 degré
CUT3DFD	
N08 G90 G94	
N09 G00 X-269.21195 Y128.32027 Z1.18577 A3=-.216361688 B3=.934284397 C3=-.283373051	; Les blocs N09 à N10 constituent la part rapide du mouvement d'accostage de la pièce à orientation constante.
N10 G00 X-251.90301 Y53.57752 Z23.85561	
N11 G01 X-247.57578 Y34.89183 Z29.52308 F50000.00000	; La part lente du mouvement d'accostage est réalisée dans les blocs N11 à N21. L'outil est déjà près de la pièce et les réglages d'usinage de moule (par exemple COMPSURF) de CYCLE832 sont alors actifs (en raison de G01 actif). La longueur de trajectoire de cette phase transitoire du comportement d'usinage de moule doit être d'environ 1000 fois la tolérance de contour (10 mm dans cet exemple).
N12 X-247.69126 Y33.82182 Z24.78219 F1061.00000	
N13 X-247.76560 Y33.13299 Z21.73022	
N14 X-247.82755 Y32.55897 Z19.18691	
N15 X-247.87918 Y32.08062 Z17.06748	
N16 X-247.92220 Y31.68200 Z15.30129	
N17 X-247.95805 Y31.34981 Z13.82947 A3=-.216361686 B3=.934284391 C3=-.283373071	
N18 X-247.98792 Y31.07299 Z12.60295 A3=-.216360662 B3=.934280801 C3=-.283385691	
N19 X-248.01282 Y30.84230 Z11.58085 A3=-.216336015 B3=.934194446 C3=-.283689030	
N20 X-248.03357 Y30.65006 Z10.72910 A3=-.216233089 B3=.933833626 C3=-.284952647	
N21 X-248.05086 Y30.48986 Z10.01931 A5=-.060687572 B5=.974940255 C5=-.214029243 A3=-.215712821 B3=.932005189 C3=-.291263295	

Code de programme	**Commentaire**
N22 **G41** X-248.06237 Y30.32400 Z9.36695 A5=-.060431854 B5=.973045457 C5=-.222554556 A3=-.214974689 B3=.929398552 C3=-.300007025 F1061.03295	; À partir de N22, la normale à la surface est définie pour la première fois sur l'ensemble du bloc (autrement dit, la normale à la surface à la fin du bloc précédent N21 est disponible et, de ce fait, la normale à la surface au début du bloc N22 et la normale à la surface à la fin du bloc N22). La condition d'activation de la correction d'outil avec G41/G42 est donc remplie.
N23 X-248.07130 Y30.15119 Z8.71082 A5=-.060165696 B5=.971048883 C5=-.231179920 A3=-.214177198 B3=.926684940 C3=-.308841625	
N24 X-248.07829 Y29.97126 Z8.05094 A5=-.059884286 B5=.968941717 C5=-.239928784 A3=-.213318480 B3=.923853466 C3=-.317789237	
N25 X-248.08317 Y29.78487 Z7.38844 A5=-.059584206 B5=.966718449 C5=-.248807482 A3=-.212397895 B3=.920898045 C3=-.326854594	
N26 X-248.08578 Y29.59254 Z6.72679 A5=-.059263963 B5=.964380907 C5=-.257793037 A3=-.211418355 B3=.917822366 C3=-.336012474	
...	

Remarque

Pour le fraisage en bout 3D avec CUT3DFD, la définition des normales à la surface est nécessaire pour l'activation de la correction d'outil avec G41/G42. La programmation de G41/G42 sans définition des normales à la surface provoque une alarme.

Voir aussi

CYCLE832 - Réglages High Speed (Page 1147)

Informations complémentaires

Fraisage en bout 3D

Ce type de fraisage 3D nécessite la description des différentes trajectoires 3D qui balayent la surface de la pièce, ligne après ligne. Les calculs sont généralement réalisés à l'aide d'un système de FAO, en tenant compte de la forme et des dimensions de l'outil. Dans le programme pièce, le postprocesseur écrit non seulement les blocs CN, mais également les orientations d'outil (lorsque la transformation 5 axes est active) et l'instruction G de la correction d'outil 3D souhaitée. Ainsi l'opérateur de la machine a la possibilité d'utiliser des outils légèrement plus petits par rapport à l'outil utilisé pour le calcul des trajectoires CN.

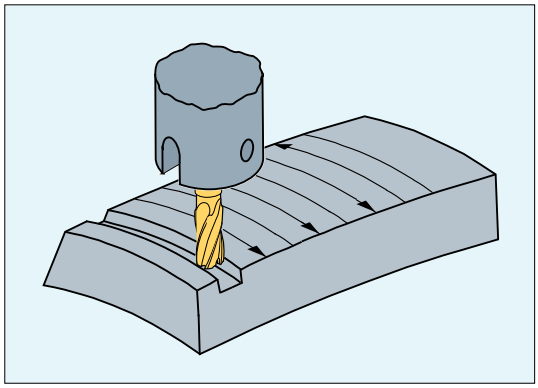


Figure 3-10 Fraisage en bout

Comportement à l'accostage

Le comportement à l'accostage est toujours NORM pour les variantes 3D de la correction du rayon d'outil.

Comportement aux angles saillants

En mode fraisage en bout, les angles saillants sont traités en tant que cercles de rayon 0, sachant que le plan du cercle est déterminé par la tangente au point final du premier bloc et la tangente du point de départ du deuxième bloc. Ainsi un changement d'orientation est possible à la transition de blocs. Par conséquent, un cercle est toujours inséré en tant qu'élément de contour au niveau d'un angle saillant. La méthode du point d'intersection n'est pas disponible en mode fraisage en bout.

Comportement lors de changements d'orientation au niveau des angles saillants

Les instructions G ORIC et ORID permettent de définir si les changements d'orientation qui ont été programmés entre les deux blocs de déplacement formant l'angle seront effectués avant le début du bloc à interpolation circulaire inséré (ORID) ou en même temps que celui-ci (ORIC).

Correction du rayon d'outil par rapport à un outil différentiel

La correction du rayon d'outil 3D rapportée à un outil différentiel est activée avec l'instruction CUT3DFD. Elle doit être utilisée lorsque le contour programmé se rapporte à la trajectoire du centre d'un outil standard et que l'usinage s'effectue avec un outil différent. Ainsi, lors du calcul de la correction du rayon d'outil 3D, seuls la valeur d'usure du rayon de l'outil actif ($TC_DP15) et les offsets de correction d'outil OFFN et TOFFR/TOFFLR éventuellement programmés sont pris en compte. Le rayon de base ($TC_DP6) de l'outil actif n'est **pas** pris en compte.

Le fraisage en bout 3D avec CUT3DFD n'est possible qu'en association avec le "lissage de la normale à la surface en fraisage en bout 3D". Celui-ci est activé par l'appel de la fonction "Top Surface" (soumise à licence) avec CYCLE832(...). L'activation doit avoir lieu **avant** l'activation de la correction d'outil avec G41/G42, et cela non pas juste avant l'attaque de l'outil, mais une longueur de trajectoire plut tôt, cette longueur étant d'environ 1000 fois la tolérance de contour (par exemple 1000 x 0,01 mm = 10 mm). Pour la désactivation, il faut suivre la procédure inverse : d'abord désactiver la correction d'outil avec G40, puis, après une longueur de trajectoire correspondant environ à 1000 fois la tolérance de contour, procéder à la désactivation avec p. ex. CUT2D (ou autre).

Pour pouvoir appliquer le "lissage des normales à la surface en fraisage en bout 3D", le déblocage de la fonction "Interpolation des normales à la surface par polynômes" est nécessaire :

MD28291 $MC_MM_SMOOTH_SURFACE_NORMALS = TRUE

Remarque

Pour le fraisage en bout 3D avec CUT3DFD en association avec "Top Surface", tenir compte des recommandations de réglages relatives à "Top Surface".

Pour vérifier les paramètres réglés, un programme de vérification spécial est disponible sur le portail SIOS.

→ Programme de vérification pour Top Surface (https://support.industry.siemens.com/cs/ww/fr/view/109738423/en)

Voir aussi

CYCLE832 - Réglages High Speed (Page 1147)

3.13.5.3 Fraisage périphérique 3D avec prise en compte d'une surface de délimitation (CUT3DCC, CUT3DCCD)

Dans le cas du fraisage périphérique 3D avec modification continue ou constante de l'orientation de l'outil, la trajectoire du centre de l'outil est souvent programmée pour un outil standard défini. Compte tenu que dans bien des cas réels, on ne dispose pas des outils standard adéquats, on peut utiliser un outil ne différant pas trop sensiblement (≤ 5 %) d'un outil standard.

Avec CUT3DCCD, une surface de délimitation, qui décrirait l'outil standard programmé, est prise en compte pour un outil différentiel réel. Le programme CN décrit la trajectoire centrale de l'outil standard.

Avec CUT3DCC, une surface de délimitation, qui aurait atteint l'outil standard programmé, est prise en compte dans le cas de l'utilisation d'outils cylindriques. Le programme CN décrit le contour de la surface d'usinage.

Le vecteur normal à la surface de délimitation est spécifié comme en mode fraisage en bout 3D par A4, B4, C4 et A5, B5, C5.

Syntaxe

```
G41/G42 CUT3DCCD/CUT3DCC CDOF2 X... Y... Z... A4=... B4=... C4=... A5=... B5=...
C5=...
...
G40 X... Y... Z...
```

Signification

`CUT3DCCD :`	CRO 3D pour le fraisage périphérique en tenant compte d'une surface de délimitation avec outil différentiel sur la trajectoire du centre de l'outil : déplacement perpendiculaire jusqu'à la surface de délimitation	
`CUT3DCC :`	CRO 3D pour le fraisage périphérique en tenant compte d'une surface de délimitation avec correction de rayon 3D : contour de la surface d'usinage	
`G41/G42 X... Y... Z... :`	Activer la correction de rayon d'outil	
	`G41 :`	Correction du rayon d'outil à gauche du contour
	`G42 :`	Correction du rayon d'outil à droite du contour
	Remarque : L'activation doit intervenir dans un bloc linéaire (G0/G1).	
`CDOF2 :`	Désactiver la détection des violations du contour pour le fraisage périphérique 3D	
`A4/5=... B4/5=... C4/5=... :`	Définition du vecteur normal à la surface de délimitation	
	`A4=... B4=... C4=... :`	Définition en début de bloc
	`A5=... B5=... C5=... :`	Définition en fin de bloc
`G40 X... Y... Z... :`	Désactiver la correction de rayon d'outil **Remarque :** La désactivation doit intervenir dans un bloc linéaire (G0/G1) avec déplacements d'axes géométriques.	

Remarque

Les instructions G pour la sélection de CRO 3D sont évaluées dans le bloc d'accostage, donc typiquement dans le bloc contenant G41 ou G42.

G41 ou G42 peut également être programmée dans des blocs sans déplacement pour les axes géométriques concernés par la correction. Dans ce cas, le bloc d'accostage est le premier bloc de déplacement après ce type de bloc.

Tout changement de la variante CRO 3D, alors que la correction du rayon d'outil est active, est ignoré sans alarme.

Exemple

```
Code de programme                                  Commentaire
N10 $TC_DP1[1,1]=120                               ; Fraise cylindrique
N20 $TC_DP6[1,1]=10
N30 $TC_DP15[1,1]=-3
...
; Usinage avec fraise cylindrique et CUT3DCCD
N110 TRAORI                                        ; Activation de la transformation.
N120 A4=0 B4=0 C4=1                                ; Définition de la normale à la surface
                                                   de délimitation en début de bloc.
N130 X0 Y0 Z0 A0 C0 T1 D1 F20000
```

Code de programme	**Commentaire**
N140 X10 Y0 Z0 G41 **CUT3DCCD** CDOF2 G64	; Activation du fraisage périphérique 3D en fonction de la surface de délimitation + désactivation de la détection des violations du contour.
N150 X20	
N160 X30 A45	; Angle obtus ==> pas de pénétration
N170 X40 A-45	; Angle aigu ==> pénétration
N180 X55	
N190 Y10 Z10	; Mouvement dans la direction de l'outil.
N200 Y20	
N210 C45	; Simple changement d'orientation.
N220 Y30 C90	
N230 A5=-1 B5=0 C5=2 Y40	; Changement de la surface.
N240 Y50 G40	; Désactivation de la correction du rayon d'outil.
...	

Informations complémentaires

Type d'outil

Le type d'outil (paramètre d'outil $TC_DP1) est évalué. Seules les fraises à corps cylindrique (fraises cylindriques, convexes ou demi-cercle et, cas limite, la fraise à bout hémisphérique) sont autorisées. Cela correspond aux types d'outil 1 à 399 à l'exception des numéros 111 et 155 à 157.

Outils standard avec arrondi

L'arrondi de l'outil standard est défini par le paramètre d'outil $TC_DP7. L'écart de l'arrondi de l'outil réel par rapport à l'outil standard est défini par le paramètre d'outil $TC_DP16.

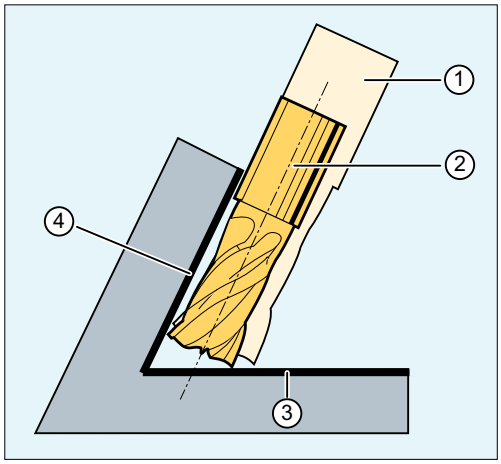
Exemple : Fraise convexe demi-cercle avec un rayon plus petit par rapport à l'outil standard.

Type d'outil	Rayon de tige (R)	Rayon de congé (r)
Outil standard avec arrondi	R = $TC_DP6	r = $TC_DP7
Outil réel avec arrondi Types d'outil 121 et 131 fraises convexes demi-cercle (fraises deux tailles à queue avec arrondi)	R' = $TC_DP6 + $TC_DP15 + OFFN	r' = $TC_DP7 + $TC_DP16

Dans cet exemple, $TC_DP15 + OFFN ainsi que $TC_DP16 sont négatifs.

CRO 3D avec CUT3DCCD : Trajectoire du centre de l'outil avec déplacement jusqu'à la surface de délimitation

En cas d'utilisation d'un outil avec un rayon plus petit que celui de l'outil standard correspondant, une fraise pénétrant dans le sens longitudinal avance jusqu'à ce qu'elle touche de nouveau le fond de la poche. Ainsi l'angle formé par la surface d'usinage et la surface de délimitation est évidé autant que l'outil le permet. Il s'agit d'un mode d'usinage mixte par fraisage périphérique et par fraisage en bout. De manière analogue à l'outil avec un rayon réduit, le déplacement s'effectue dans la direction inverse dans le cas d'un outil avec un rayon plus grand.



① Outil standard
② Outil avec un rayon plus petit avec pénétration jusqu'à la surface de délimitation
③ Surface de délimitation
④ Surface d'usinage

Par rapport à toutes les autres corrections d'outil du groupe d'instructions G 22, un paramètre d'outil $TC_DP6 spécifié pour CUT3DCCD est sans signification pour le rayon d'outil et n'influence pas la correction résultante. Le décalage de correction résulte de la somme de la valeur d'usure du rayon de l'outil (paramètre d'outil $TC_DP15) et d'un décalage d'outil OFFN programmé pour le calcul du décalage perpendiculaire à la surface de délimitation.

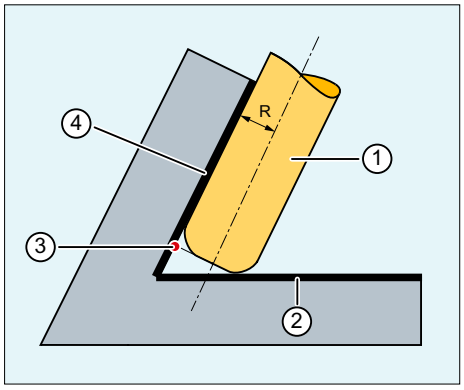
Le programme pièce généré ne permet pas de savoir si la surface usinée se situe à gauche ou à droite de la trajectoire. On part donc d'un rayon positif et d'une valeur d'usure négative de l'outil original. Une valeur d'usure négative décrit toujours un outil de diamètre réduit.

Utilisation d'outils cylindriques

En cas d'utilisation d'outils cylindriques, un déplacement est uniquement nécessaire si la surface d'usinage et la surface de délimitation forment un angle aigu (moins de 90 degrés). L'utilisation de fraises convexes demi-cercle (fraises deux tailles à queue avec arrondi) exige un déplacement de l'outil dans le sens longitudinal aussi bien en présence d'un angle aigu que d'un angle obtus.

CRO 3D avec CUT3DCC : contour de la surface d'usinage

Si CUT3DCC est actif avec une fraise convexe demi-cercle, la trajectoire programmée se rapporte à une fraise cylindrique fictive de diamètre identique. La référence de trajectoire qui en résulte est représentée sur la figure suivante pour l'utilisation d'une fraise convexe demi-cercle.



① Fraise convexe demi-cercle
② Surface de délimitation
③ Référence de trajectoire
④ Surface d'usinage
R Rayon de tige (rayon d'outil)

L'angle formé par la surface d'usinage et la surface de délimitation peut également passer d'un angle aigu à un angle obtus, et vice versa, à l'intérieur d'un bloc.

L'outil utilisé réellement peut être aussi bien plus grand que plus petit que l'outil standard. Cependant le rayon d'arrondi résultant ne doit pas devenir négatif et le signe du rayon d'outil résultant doit rester inchangé.

Avec CUT3DCC, le programme pièce CN se rapporte au contour de la surface d'usinage. Comme pour la correction de rayon d'outil conventionnelle, on tient compte du rayon total constitué des composantes suivantes :

- Rayon d'outil (paramètre d'outil $TC_DP6)
- Valeur d'usure (paramètre d'outil $TC_DP15)
- Décalage d'outil OFFN programmé pour le calcul du décalage perpendiculaire à la surface de délimitation.

La position de la surface de délimitation est déterminée à partir de la différence suivante :

Dimensions de l'outil standard - rayon d'outil (paramètre d'outil $TC_DP6)

Advanced Surface / Top Surface

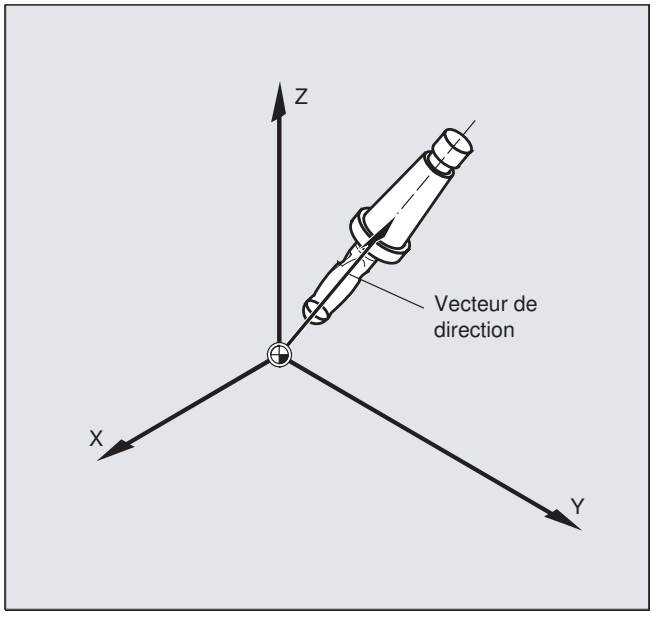
Remarque

Pour l'utilisation de la correction du rayon d'outil CUT3DCC / CUT3DCCD en association avec la fonction soumise à la licence "Advanced Surface" ou "Top Surface", tenir compte des recommandations de réglages relatives à "Advanced Surface" / "Top Surface". Pour vérifier les paramètres réglés, un programme de vérification spécial est disponible sur le portail SIOS.

→ Programme de vérification pour Advanced Surface / Top Surface (https://support.industry.siemens.com/cs/ww/fr/view/109738423/en)

3.13.6 Orientation de l'outil (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST)

On entend par orientation de l'outil l'orientation géométrique de l'outil dans l'espace. Sur une machine-outil à cinq axes, l'orientation de l'outil peut être spécifiée par des instructions du programme.



Des déplacements d'arrondissement de l'orientation activés par `OSD` et `OST` sont formés différemment selon le type d'interpolation pour l'orientation de l'outil.

En cas d'interpolation vectorielle active, le tracé lissé de l'orientation est également interpolé par l'interpolation vectorielle. Par contre, en cas d'interpolation d'axe rotatif active, l'orientation est directement lissée par des déplacements d'axe rotatif.

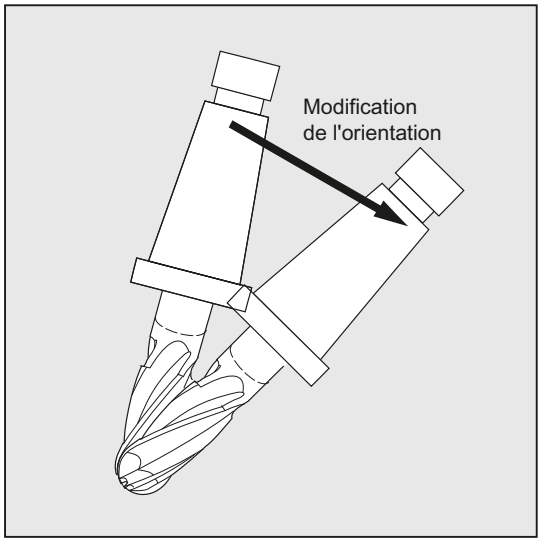
Programmation

Programmation du changement d'orientation :

On peut programmer un changement d'orientation de l'outil par :

- programmation directe des axes rotatifs `A`, `B`, `C` (interpolation d'axe rotatif)
- Angles d'Euler ou RPY
- Vecteur de direction (interpolation vectorielle par indication de `A3` ou de `B3` ou de `C3`)
- `LEAD/TILT` (fraisage en bout)

Le système de coordonnées de référence est soit le système de coordonnées machine (`ORIMKS`), soit le système de coordonnées pièce courant (`ORIWKS`).



Programmation de l'orientation de l'outil :

ORIC :	orientation et déplacement le long de la trajectoire en parallèle
ORID :	orientation et déplacement le long de la trajectoire successifs
OSOF :	aucun lissage de l'orientation
OSC :	orientation constante
OSS :	lissage de l'orientation seulement en début du bloc
OSSE :	lissage de l'orientation en début et en fin de bloc
ORIS :	Vitesse, en degrés par mm, du changement d'orientation lors de l'activation du lissage de l'orientation (s'applique à OSS et OSSE)
OSD :	Arrondissement de l'orientation par prescription de la longueur d'arrondissement avec la donnée de réglage : SD42674 $SC_ORI_SMOOTH_DIST
OST :	Arrondissement de l'orientation par prescription de la tolérance d'angle en degrés en cas d'interpolation vectorielle avec la donnée de réglage : SD42676 $SC_ORI_SMOOTH_TOL En cas d'interpolation d'axe rotatif, la tolérance prescrite est prise comme étant l'écart maximal des axes d'orientation.

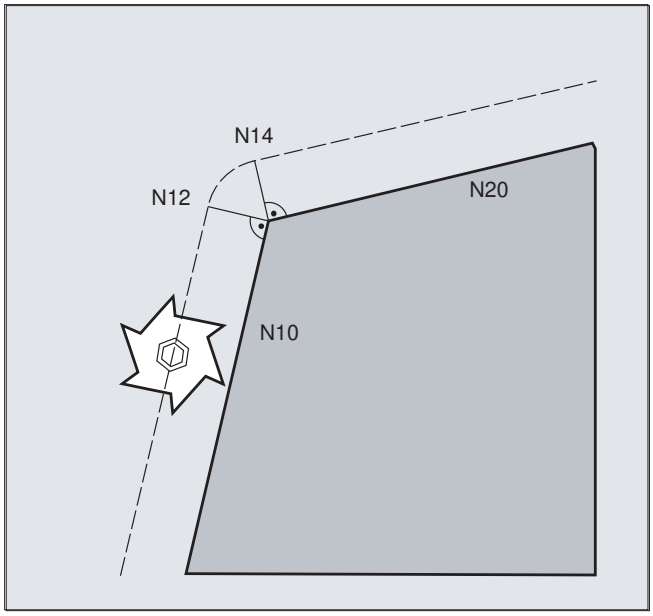
Remarque

Toutes les instructions réalisant un arrondissement de l'orientation de l'outil (OSOF, OSC, OSS, OSSE, OSD et OST) sont regroupées dans le groupe G 34. Elles sont à effet modal, autrement dit, une seule de ces instructions peut être exécutée à la fois.

Exemples

Exemple 1 : ORIC

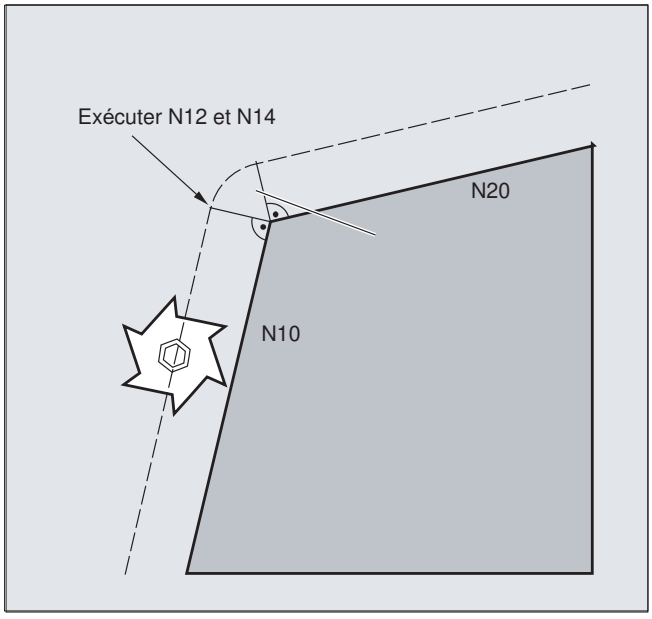
Si, entre les blocs de déplacement N10 et N20, deux blocs ou plus avec des changements d'orientation (par ex. A2=... B2=... C2=) sont programmés et que ORIC est activée, le bloc à interpolation circulaire inséré est réparti sur ces blocs intermédiaires conformément à la valeur des variations angulaires.



Code de programme	Commentaire
ORIC	
N8 A2=… B2=… C2=…	
N10 X… Y… Z…	
N12 C2=… B2=…	; Le bloc à interpolation circulaire qui est inséré à l'angle saillant est réparti sur N12 et N14 en fonction du changement d'orientation. Le mouvement circulaire et le changement d'orientation sont exécutés en parallèle.
N14 C2=… B2=…	
N20 X =…Y=… Z=… G1 F200	

Exemple 2 : ORID

Si ORID est activée, tous les blocs situés entre les deux blocs de déplacement sont exécutés à la fin du premier bloc de déplacement. Le bloc à interpolation circulaire avec orientation constante est exécuté directement avant le deuxième bloc de déplacement.



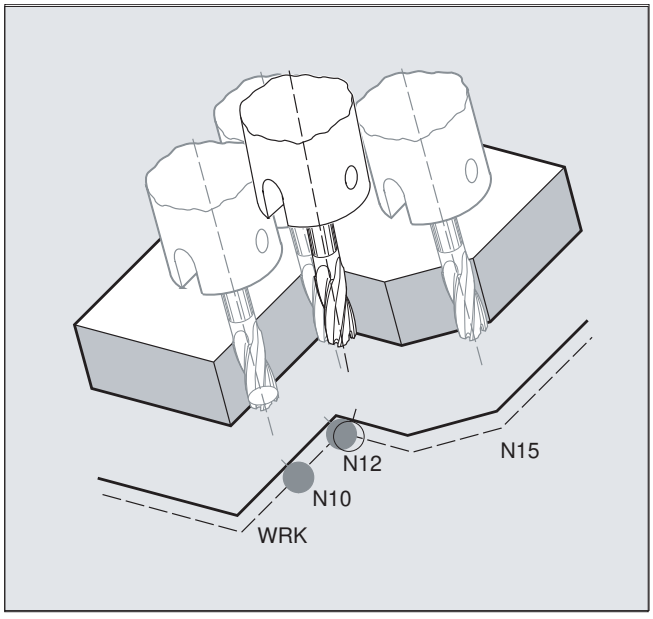
Code de programme	Commentaire
ORID	
N8 A2=… B2=… C2=…	
N10 X… Y… Z…	
N12 A2=… B2=… C2=…	; Le bloc N12 et N14 est exécuté à la fin de N10. Puis le bloc à interpolation circulaire se retire avec l'orientation courante.
N14 M20	; Fonctions auxiliaires, etc.
N20 X… Y… Z…	

Remarque

L'instruction de programme qui est active dans le premier bloc de déplacement d'un angle saillant est déterminante pour le type de changement d'orientation à cet angle saillant.

Sans changement d'orientation : Si l'orientation n'est pas modifiée à la fin du bloc, la section de l'outil est un cercle en contact avec les deux contours.

Exemple 3 : changement d'orientation à un angle rentrant



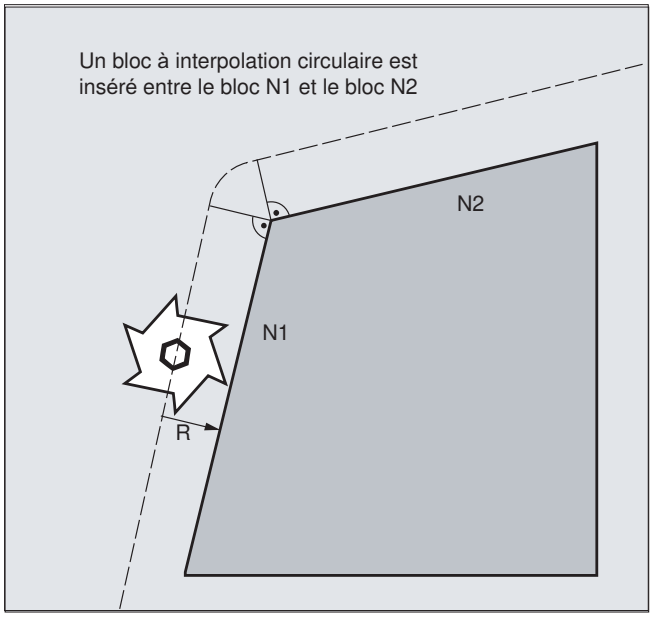
Code de programme
ORIC
N10 X …Y… Z… G1 F500
N12 X …Y… Z… A2=… B2=… C2=…
N15 X …Y… Z… A2=… B2=… C2=…

Informations complémentaires

Comportement aux angles saillants

Un bloc à interpolation circulaire avec un rayon égal à celui de la fraise est toujours inséré aux angles saillants.

Les instructions de programme `ORIC` et `ORID` permettent de définir si les changements d'orientation qui ont été programmés entre le bloc `N1` et le bloc `N2` seront effectués avant le début du bloc à interpolation circulaire inséré ou en même temps que celui-ci.



Si un changement d'orientation est nécessaire aux angles saillants, celui-ci peut s'effectuer parallèlement à l'interpolation ou bien séparément, en même temps que le déplacement le long de la trajectoire.

Avec `ORID`, les blocs insérés sont exécutés d'abord sans déplacement le long de la trajectoire. Le bloc de raccordement par interpolation circulaire est inséré directement avant le second des deux blocs de déplacement qui forment l'angle.

Si plusieurs blocs d'orientation sont insérés à un angle saillant et si `ORIC` est actif, le mouvement circulaire est réparti sur les différents blocs insérés, conformément aux valeurs des changements d'orientation de ces blocs.

Arrondissement de l'orientation avec OSD ou OST

Lors de l'arrondissement avec `G642`, l'écart maximal pour les axes de contour et les axes d'orientation ne peut pas varier beaucoup. La tolérance, la plus petite des deux, détermine la forme du déplacement d'arrondissement et de la tolérance d'angle et permet de lisser assez fortement le tracé de l'orientation sans écarts de contour importants.

L'activation de `OSD` ou `OST` permet, avec une longueur d'arrondissement ou une tolérance d'angle prescrites, de lisser "généreusement" de très faibles écarts du tracé de l'orientation sans écarts de contour importants.

Remarque

A la différence de l'arrondissement du contour (et du tracé de l'orientation) avec `G642`, aucun bloc séparé n'est formé lors de l'arrondissement de l'orientation avec `OSD` ou `OST`, mais le déplacement d'arrondissement est directement introduit dans les blocs d'origine programmés.

Avec `OSD` ou `OST`, aucune transition entre blocs ne peut être arrondie pour lesquels un changement du type d'interpolation pour l'orientation de l'outil (vecteur → axe rotatif, axe rotatif → vecteur) a lieu. Ces transitions entre blocs peuvent éventuellement être arrondies avec les fonctions d'arrondissement conventionnelles `G641`, `G642` ou `G643`.

3.13.7 Numéros D libres, numéro de tranchant

3.13.7.1 Numéros D libres, numéro de tranchant (adresse CE)

Numéro D

Les numéros D peuvent être utilisés en tant que numéros de correcteur. En outre, CE permet l'adressage du numéro de tranchant. L'écriture du numéro de tranchant est possible à l'aide de la variable système $TC_DPCE.

Préréglage : N° de correcteur == n° de tranchant

Les nombres maximaux de numéros D (numéros de tranchant) et de tranchants par outil sont définis via les paramètres machine (→ constructeur de machines). Les instructions suivantes sont recommandées uniquement lorsque le nombre maximal de numéros de tranchant (MD18105) est plus grand que le nombre de tranchants par outil (MD18106). Tenir compte des indications du constructeur de machines.

Informations complémentaires

Description fonctionnelle Outils

3.13.7.2 Libre affectation des numéros D : contrôle des numéros D (CHKDNO)

L'instruction `CKKDNO` permet un contrôle d'univocité de l'attribution des numéros D. Les numéros D de tous les outils définis dans une unité TO ne peuvent être attribués qu'une fois. La vérification ne porte pas sur les outils de rechange.

Syntaxe

```
state=CHKDNO(Tno1,Tno2,Dno)
```

Signification

`state` :	=TRUE :	Les numéros D ont été attribués de façon univoque dans le domaine contrôlé.
	=FALSE :	Une collision entre des numéros D a été constatée ou le paramétrage est incorrect. Tno1, Tno2 et Dno sont les paramètres ayant conduit à la collision. Ces données peuvent être exploitées dans le programme pièce.
`CHKDNO(Tno1,Tno2)` :	Tous les numéros D des outils indiqués sont comparés.	
`CHKDNO(Tno1)` :	Tous les numéros D de l'outil Tno1 sont comparés à tous les numéros D des autres outils.	
`CHKDNO` :	Tous les numéros D de tous les outils sont comparés à tous les numéros D de tous les autres outils.	

3.13.7.3 Libre affectation des numéros D : modification des numéros D (GETDNO, SETDNO)

L'attribution des numéros D doit être univoque. Deux tranchants distincts d'un outil ne peuvent pas posséder le même numéro D.

GETDNO

Cette instruction fournit le numéro D d'un certain tranchant (ce) d'un outil avec le numéro T t. S'il n'existe pas de numéro D correspondant aux paramètres indiqués, la valeur d=0 est attribuée. Si le numéro D n'est pas valable, d contient une valeur supérieure à 32000.

SETDNO

Avec cette instruction, vous attribuez la valeur d au numéro D du tranchant ce de l'outil t. ℸLe paramètre state fournit le résultat de l'instruction (TRUE ou FALSE). S'il n'existe pas d'enregistrement correspondant aux paramètres indiqués, le résultat de l'instruction est FALSE. Les erreurs de syntaxe génèrent une alarme. La valeur 0 ne peut pas être attribuée explicitement à un numéro D.

Syntaxe

```
d = GETDNO (t,ce)
state = SETDNO (t,ce,d)
```

Signification

d :	Numéro D du tranchant de l'outil
t :	Numéro T de l'outil
ce :	Numéro du tranchant (numéro CE) de l'outil
state :	Indique si l'instruction a pu être exécutée sans erreur (TRUE ou FALSE).

Exemple de changement de nom d'un numéro D

Programmation	**Commentaire**
`$TC_DP2[1,2] = 120`	
`$TC_DP3[1,2] = 5.5`	
`$TC_DPCE[1,2] = 3`	`; Numéro de tranchant CE`
`...`	
`N10 def int N°DAncien, N°DNouveau = 17`	
`N20 N°DAncien = GETDNO(1,3)`	
`N30 SETDNO(1,3,N°DNouveau)`	

Le nouveau correcteur D17 est affecté au tranchant CE=3. L'accès aux données de ce tranchant a dorénavant lieu avec le numéro D17 ; ceci concerne les paramètres système aussi bien que la programmation avec l'adresse CN.

3.13.7.4 Libre affectation des numéros D : détermination du numéro T correspondant au numéro D prescrit (GETACTTD)

La fonction prédéfinie GETACTTD détermine le numéro T correspondant à un numéro D absolu. Il n'y a pas de contrôle d'univocité. En présence de plusieurs numéros D identiques dans une unité TO, l'instruction fournit le numéro T du premier outil trouvé.

Syntaxe

```
<status>=GETACTTD(<TNr>,<DNr>)
```

Signification

GETACTTD() :	Appel de fonction		
<DNr> :	Numéro D pour lequel le numéro T doit être recherché.		
	Type de données :	INT	
<TNr> :	Numéro T trouvé		
	Type de données :	VAR INT	
<status> :	Résultat		
	Type de données :	INT	
	Valeur :	0	Le numéro T a été trouvé. <TNr> contient sa valeur.
		-1	Il n'existe pas de numéro T pour le numéro D indiqué, <TNr>=0.
		-2	Le numéro D n'est pas univoque. <TNr> contient la valeur du premier outil trouvé comportant le numéro D dont la valeur est <DNr>.
		-5	La fonction n'a pas pu être exécutée pour une autre raison.

3.13.7.5 Libre affectation des numéros D : numéros D déclarés non valides (DZERO)

L'instruction DZERO sert à faciliter le réoutillage. Les correcteurs déclarés non valides ne font plus l'objet d'un contrôle par l'instruction CHKDNO. Pour avoir de nouveau accès aux correcteurs, il faut procéder à une nouvelle affectation des numéros D avec SETDNO.

Syntaxe

```
DZERO
```

Signification

DZERO :	Déclare tous les numéros D de l'unité TO non valides.

3.13.8 Cinématique de l'organe porte-outil

Conditions

Un organe porte-outil ne peut orienter un outil dans toutes les directions de l'espace que

- si deux axes rotatifs V_1 et V_2 sont présents.
- si les axes rotatifs sont perpendiculaires les uns par rapport aux autres.
- si l'axe longitudinal de l'outil est perpendiculaire au deuxième axe rotatif V_2.

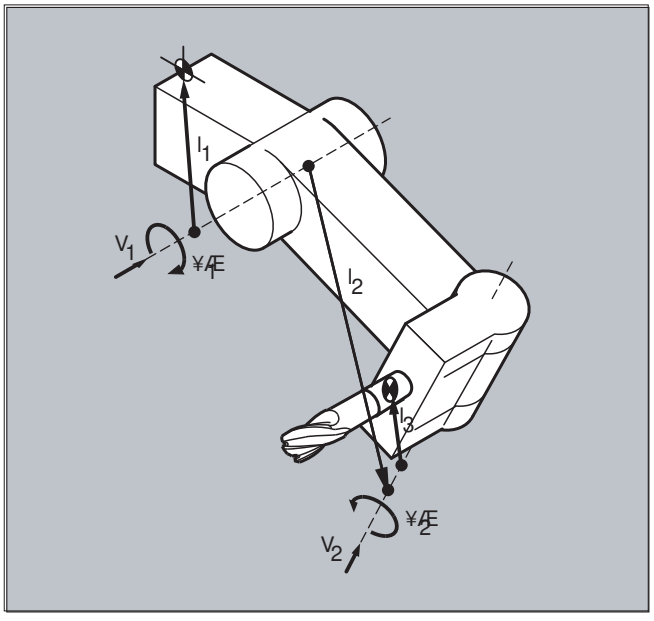
En outre, la condition ci-dessous doit être satisfaite dans le cas des machines sur lesquelles il doit être possible de régler toutes les orientations :

- - l'orientation de l'outil doit être perpendiculaire au premier axe rotatif V_1.

Fonction

La cinématique d'un organe porte-outil ayant deux axes rotatifs au maximum $v_{1\ ou}\ v_2$ est décrite à l'aide des 17 variables système allant de $TC_CARR1[m] à $TC_CARR17[m]. Cette description comprend :

- le vecteur distance entre le premier axe rotatif et le point de référence de l'organe porte-outil I_1, le vecteur distance entre le premier et le second axe rotatif I_2, le vecteur distance entre le second axe rotatif et le point de référence de l'outil I_3.
- le vecteur de direction des deux axes rotatifs V_1, V_2.
- les angles rotatifs α1, α2 autour des deux axes. Les angles rotatifs sont positifs s'ils provoquent une rotation en sens horaire pour un observateur regardant dans la direction des axes rotatifs.



Pour les machines à **cinématique décomposée** (l'outil et la pièce sont orientables), les variables système ont été complétées par $TC_CARR18[m] bis $TC_CARR23[m].

Paramètre

Fonction des variables système pour organes porte-outil orientables			
Désignation	**Composante x**	**Composante y**	**Composante z**
Vecteur offset l_1	$TC_CARR1[m]	$TC_CARR2[m]	$TC_CARR3[m]
Vecteur offset l_2	$TC_CARR4[m]	$TC_CARR5[m]	$TC_CARR6[m]
Axe rotatif v_1	$TC_CARR7[m]	$TC_CARR8[m]	$TC_CARR9[m]
Axe rotatif v_2	$TC_CARR10[m]	$TC_CARR11[m]	$TC_CARR12[m]
Angle rotatif α_1 Angle rotatif α_2	$TC_CARR13[m] $TC_CARR14[m]		
Vecteur offset l_3	$TC_CARR15[m]	$TC_CARR16[m]	$TC_CARR17[m]

Extension des variables système pour organes porte-outil orientables			
Désignation	**Composante x**	**Composante y**	**Composante z**
Vecteur offset l_4	$TC_CARR18[m]	$TC_CARR19[m]	$TC_CARR20[m]
Descripteur d'axe Axe rotatif v_1 Axe rotatif v_2	Descripteur d'axe pour les axes rotatifs $v_1$et v_2 (préréglage avec zéro) $TC_CARR21[m] $TC_CARR22[m]		
Type de cinématique **T**ool **P**art **M**ixed mode	$TC_CARR23[m]		
	Type de cinématique T ->	Type de cinématique P ->	Type de cinématique M
	seul l'outil est orientable (préréglage).	seule la pièce est orientable.	pièce et outil sont orientables.
Offset des axes rotatifs v_1 et v_2	Angle en degrés des axes rotatifs $v_1$et v_2 en position initiale $TC_CARR24[m] $TC_CARR25[m]		
Décalage angulaire des axes rotatifs v_1 et v_2	Offset en degrés de la denture Hirth des axes rotatifs $v_1$et v_2 $TC_CARR26[m] $TC_CARR27[m]		
Incrément angulaire v_1 Axe rotatif v_2 Axe rotatif	Incrément en degrés de la denture Hirth des axes rotatifs $v_1$et v_2 $TC_CARR28[m] $TC_CARR29[m]		
Position min. Axe rotatif v_1 Axe de rotatif v_2	Limite logicielle pour position minimale des axes rotatifs $v_1$et v_2 $TC_CARR30[m] $TC_CARR31[m]		
Position max. Axe rotatif v_1 Axe rotatif v_2	Limite logicielle pour position maximale des axes rotatifs $v_1$et v_2 $TC_CARR32[m] $TC_CARR33[m]		
Nom de l'organe porte-outil	À la place d'un nombre, un organe porte-outil peut se voir affecter un nom. $TC_CARR34[m]		
Utilisateur : Nom d'axe 1 Nom d'axe 2 Identificateur **Position**	Utilisation prévue par l'utilisateur à l'intérieur des cycles de mesure. $TC_CARR35[m] $TC_CARR36[m] $TC_CARR37[m]		
	$TC_CARR38[m]	$TC_CARR39[m]	$TC_CARR40[m]

Extension des variables système pour organes porte-outil orientables			
Décalage fin	Paramètres pouvant être ajoutés **aux valeurs contenues dans les paramètres de base**.		
Vecteur offset l_1	$TC_CARR41[m]	$TC_CARR42[m]	$TC_CARR43[m]
Vecteur offset l_2	$TC_CARR44[m]	$TC_CARR45[m]	$TC_CARR46[m]
Vecteur offset l_3	$TC_CARR55[m]	$TC_CARR56[m]	$TC_CARR57[m]
Vecteur offset l_4	$TC_CARR58[m]	$TC_CARR59[m]	$TC_CARR60[m]
Axe rotatif v_1	$TC_CARR64[m]		
Axe rotatif v_2	$TC_CARR65[m]		

Remarque

Explication des paramètres

"m" est le numéro de l'organe porte-outil à décrire.

Les paramètres $TC_CARR47 jusqu'à $TC_CARR54 ainsi que $TC_CARR61 jusqu'à $TC_CARR63 ne sont pas définis et génèrent une alarme en cas de tentative d'accès en lecture ou en écriture.

Vous pouvez choisir librement les points initiaux et finals des vecteurs distances sur les axes. Les angles rotatifs α_1, $\alpha_2$autour des deux axes sont égaux à 0° lorsque l'organe porte-outil est en position initiale. Il existe donc un grand nombre de possibilités pour décrire la cinématique d'un organe porte-outil.

Vous pouvez décrire des organes porte-outil n'ayant qu'un axe rotatif ou aucun axe rotatif en annulant les composantes du vecteur de direction d'un ou des deux axes rotatifs.
Dans le cas d'un organe porte-outil sans axe rotatif, les vecteurs distances ont le même effet que des corrections d'outil supplémentaires, dont les composantes ne sont pas influencées par le basculement entre les plans d'usinage (G17 à G19).

Extension des paramètres

Paramètres des axes rotatifs

Les variables système $TC_CARR24[m] à $TC_CARR33[m], dont les significations sont indiquées ci-dessous, sont introduites :

Offset des axes rotatifs v_1 et v_2	Changement de position de l'axe rotatif v_1 ou v_2 lorsque l'organe porte-outil orientable est en position initiale.
Décalage **angulaire/ incrément** angulaire des axes rotatifs $v_1$et v_2	Offset ou incrément angulaire de la denture Hirth des axes rotatifs v_1 et v_2. L'angle programmé ou calculé est arrondi à la valeur la plus proche qui s'obtient à l'aide de la formule phi = s + n * d, n étant un nombre entier.
Position minimale et maximale Axes rotatifs v_1, v_2	Position minimale/maximale des axes rotatifs Angles limites (limite logicielle) des axes rotatifs v1 et v2.

Paramètres pour l'utilisateur

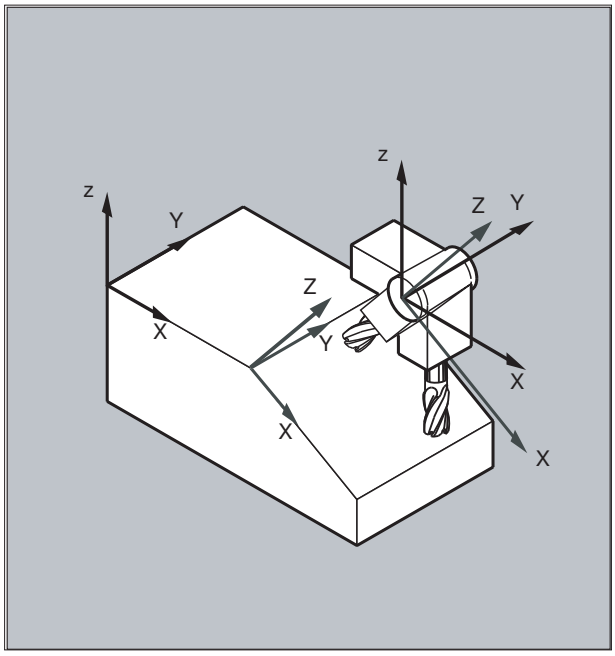
$TC_CARR34 à $TC_CARR40 contiennent des paramètres qui sont mis à la libre disposition de l'utilisateur et qui, jusqu'à la version de logiciel 6.4, ne sont pas exploités en standard à l'intérieur du NCK ou qui sont sans signification.

Paramètres de décalage fin

$TC_CARR41 à $TC_CARR65 contiennent des paramètres de décalage fin pouvant être ajoutés aux valeurs contenues dans les paramètres de base. On obtient une valeur de décalage fin affectée à un paramètre de base lorsque la valeur 40 est ajoutée au numéro de paramètre.

Exemple

L'organe porte-outil utilisé dans l'exemple suivant peut être décrit entièrement par une rotation autour de l'axe Y.



Code de programme	**Commentaire**
N10 $TC_CARR8[1]=1	; Définition de la composante Y du premier axe rotatif de l'organe porte-outil 1.
N20 $TC_DP1[1,1]=120	; Définition d'une fraise cylindrique.
N30 $TC_DP3[1,1]=20	; Définition d'une fraise cylindrique d'une longueur de 20 mm.
N40 $TC_DP6[1,1]=5	; Définition d'une fraise cylindrique d'un rayon de 5 mm.
N50 ROT Y37	; Définition d'un frame avec rotation de 37° autour de l'axe Y.
N60 X0 Y0 Z0 F10000	; Accostage du point de départ.

Code de programme	Commentaire
N70 G42 CUT2DF TCOFR TCARR=1 T1 D1 X10	; Réglage correction de rayon, correction de longueur d'outil dans le frame pivoté, sélection organe porte-outil 1, outil 1.
N80 X40	; Effectuer l'usinage sous un angle rotatif de 37°.
N90 Y40	
N100 X0	
N110 Y0	
N120 M30	

Informations complémentaires

Cinématique décomposée

Pour les machines à cinématique décomposée (l'outil et la pièce sont orientables), les variables système supplémentaires `$TC_CARR18[m]` à `$TC_CARR23[m]` permettent de définir :

l'organe porte-pièce orientable à l'aide :

- du vecteur distance entre le second axe rotatif V_2 et le point de référence d'une table porte-pièce pivotante I_4 du troisième axe rotatif.

les axes rotatifs à l'aide :

- des deux descripteurs d'axes de canal associés aux axes rotatifs V_1 et V_2, et dont les positions peuvent être interrogées pour la détermination de l'orientation.

le type de cinématique à l'aide d'une des valeurs T, P ou M :

- type de cinématique T : seul l'outil est orientable.
- type de cinématique P : seule la pièce est orientable.
- type de cinématique M : l'outil et la pièce sont orientables.

Effacement des données d'organe porte-outil

Avec `$TC_CARR1[0] = 0`, vous pouvez effacer les données de tous les organes porte-outil.

La valeur du paramètre pour le type de cinématique `$TC_CARR23[T] = T` doit être une des trois lettres majuscules ou minuscules (T, P, M) ; ce paramètre ne doit donc pas être effacé.

Modification des données d'organe porte-outil

Vous pouvez modifier chacune des valeurs en lui affectant une nouvelle valeur dans le programme pièce. Tout autre caractère que T, P ou M conduit à une alarme lors de la tentative d'activer l'organe porte-outil orientable.

Lecture des données d'organe porte-outil

Vous pouvez lire chacune des valeurs en l'affectant à une variable dans le programme pièce.

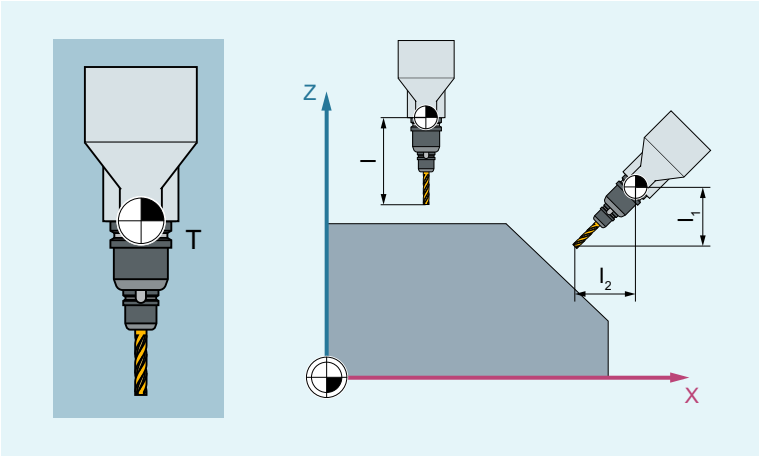
Décalage fin

Une valeur de décalage fin non autorisée n'est détectée que lorsqu'un organe porte-outil orientable contenant une telle valeur est activé et que, dans le même temps, la donnée de réglage SD42974 $SC_TOCARR_FINE_CORRECTION = TRUE.

La valeur absolue du décalage fin autorisée est limitée par les paramètres machine à une valeur maximale admissible.

3.13.9 Correction de longueur d'outil pour organes porte-outils orientables (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)

Lorsque l'orientation dans l'espace de l'organe porte-outil, et donc de l'outil, change, leurs composantes de longueur d'outil varient également :



T	Point de référence de l'organe porte-outil
l_1, l_2	Composantes de longueur d'outil

Par conséquent, après un changement, par réglage manuel ou remplacement de l'organe porte-outil à orientation spatiale fixe par exemple, les composantes de longueur d'outil sont à redéfinir. Pour cela, on utilise les instructions du groupe G 42 "Organe porte-outil".

Syntaxe

```
TCARR=<m>
TCOABS
TCOFR
TCOFRZ/TCOFRY/TCOFRX
```

Signification

Élément	Type	Signification	
TCARR=<m>	Adresse	Demander un organe porte-outil	
		<m>	Numéro de l'organe porte-outil

Élément	Type	Signification	
TCOABS	Instruction G	Calculer les composantes de longueur d'outil à partir de l'orientation de l'organe porte-outil.	
TCOFR	Instruction G	Détermination des composantes de longueur d'outil à partir de l'orientation du frame actif.	
TCOFRZ/TCOFRY/ TCOFRX	Instruction G	L'instruction TCOFRX/TCOFRY/TCOFRZ émane d'un outil orienté dans la direction correspondante (X/Y/Z) et calcule les angles de réglage de l'organe porte-outil orientable de sorte que l'outil soit orienté dans la même direction dans le frame actif.	
		TCOFRZ	L'outil orienté dans la direction Z est orienté de manière à être également orienté dans la direction Z dans le frame actif.
		TCOFRY	L'outil orienté dans la direction Y est orienté de manière à être également orienté dans la direction Y dans le frame actif.
		TCOFRX	L'outil orienté dans la direction X est orienté de manière à être également orienté dans la direction X dans le frame actif.

Informations complémentaires

Sélection de l'organe porte-outil (TCARR)

L'organe porte-outil est associé à un bloc de données d'organe porte-outil qui décrit sa géométrie. Si un outil est actif au moment de la sélection de l'organe porte-outil, il est affecté au nouvel organe porte-outil sélectionné et les données géométriques de cet organe porte-outil prennent effet immédiatement. Il n'est pas nécessaire de changer l'outil actif ou de le reprogrammer. Lorsqu'un outil est réactivé, il est traité comme s'il était monté sur l'organe porte-outil actif.

TCARR=0 permet de désélectionner l'organe porte-outil actif.

Remarque

Les données géométriques actuelles de l'organe porte-outil peuvent aussi être définies dans le programme CN avec les variables système correspondantes (voir "Accès aux blocs de données d'organe porte-outil").

Remarque

Du point de vue de la commande, les numéros d'organe porte-outil <m> et les numéros d'outil <n> peuvent être combinés librement. Dans l'application réelle, certaines combinaisons peuvent être exclues pour des raisons d'usinage et des raisons mécaniques. La commande ne vérifie pas la pertinence des combinaisons.

Accès aux blocs de données d'organe porte-outil

Procéder comme suit pour accéder aux données géométriques actuelles d'un organe porte-outil depuis le programme CN :

- Écriture :
  `$TC_CARR<n>[<m>]=<valeur>`
- Lecture :
  `<valeur>=$TC_CARR<n>[<m>]`
  (<valeur> doit être une variable du type de données REAL)

Si l'organe porte-outil référencé n'est pas défini, une alarme apparaît.

Mise à zéro des blocs de données d'organe porte-outil

Depuis le programme CN, toutes les valeurs de tous les blocs de données d'organe porte-outil peuvent être supprimées à l'aide d'un ordre :

```
$TC_CARR1[0] = 0
```

Les différents blocs de données d'organe porte-outil peuvent être supprimés individuellement à l'aide de la procédure prédéfinie DELTC.

Effet de la sélection de l'organe porte-outil

Un organe porte-outil devient effectif dès qu'un organe porte-outil et un outil ont été activés. La seule sélection d'un organe porte-outil n'a aucun effet.

L'effet de la sélection d'un organe porte-outil dépend de l'instruction du groupe G 42 (Organe porte-outil) qui est programmée en même temps que TCARR.

Un changement de l'instruction G depuis ce groupe entraîne un recalcul des composantes de longueur d'outil lorsqu'un organe porte-outil est actif.

Calcul de la correction de longueur d'outil à partir de l'orientation de l'organe porte-outil (TCOABS)

TCOABS calcule la correction de longueur d'outil sur la base des angles d'orientation de l'organe porte-outil, qui sont stockés dans les variables système $TC_CARR13 et $TC_CARR14. L'orientation prise en compte est indépendante de l'orientation du frame actif.

Pour la définition de la cinématique de l'organe porte-outil avec des variables système, voir chapitre "Cinématique de l'organe porte-outil (Page 797)" dans le manuel de programmation Programmation CN.

Détermination des composantes de longueur d'outil à partir de l'orientation du frame actif (TCOFR)

TCOFR permet de déterminer les angles d'orientation de l'organe porte-outil à partir de l'orientation du frame actif. Les valeurs stockées dans les données de l'organe porte-outil restent toutefois inchangées. Elles sont utilisées en outre pour lever les ambiguïtés pouvant apparaître lors du calcul des angles de rotation à partir d'un frame (voir le paragraphe suivant).

Ambiguïtés

Une orientation de l'outil déterminée, définie par un frame, peut en général être réglée avec deux axes par **deux** paires d'angles de rotation différentes. Entre ces deux positions possibles,

la commande sélectionne celle pour laquelle les angles de rotation sont les plus proches possible des angles de rotation programmés.

Remarque

En cas d'ambiguïté, il est en général nécessaire de stocker les angles escomptés à partir du frame dans les données de l'organe porte-outil.

Orientation de l'outil dans le frame actif (TCOFRZ/TCOFRY/TCOFRX)

Pour le calcul des angles de réglage de l'organe porte-outil, l'instruction TCOFRX/TCOFRY/TCOFRZ se base sur une orientation dans la direction X/Y/Z, indépendamment de l'orientation active. La sélection du plan d'usinage (G17/G18/G19) n'influence aucunement l'orientation dans le frame actif. Elle influence toutefois le calcul de la correction de longueur d'outil.

Changement de frame

Après la sélection de l'outil, l'utilisateur peut changer de frame. Cela n'a aucune influence sur les composantes de la correction de longueur d'outil.

Les angles de rotation stockés dans les données de l'organe porte-outil ne sont pas influencés par les angles de rotation définis avec les frames. En cas de passage de TCOFR à TCOABS, les angles de rotation (programmés) initiaux figurant dans les données de l'organe porte-outil sont à nouveau effectifs.

Recalcul de la correction de longueur d'outil

Pour que la correction de longueur d'outil soit recalculée après un changement de frame, l'outil doit être à nouveau sélectionné. Lors du calcul de la correction de longueur d'outil, les angles de rotation de l'organe porte-outil font également l'objet d'un calcul. Étant donné que les organes porte-outil à deux axes de rotation ont généralement deux paires d'angles de rotation, avec lesquels on peut adapter l'orientation de l'outil au frame actif, les valeurs des angles de rotation stockées dans les données de l'organe porte-outil doivent correspondre au moins approximativement aux angles de rotation réglés mécaniquement.

Remarque

Dans pratiquement tous les cas d'ambiguïté, il est nécessaire de stocker les angles escomptés à partir du frame dans les données de l'organe porte-outil.

Remarque

L'orientation de l'outil doit être adaptée manuellement au frame actif.

Remarque

La commande n'est pas en mesure de contrôler la possibilité de réglage sur la machine des angles de rotation calculés sur la base de l'orientation du frame. Si la machine est conçue de sorte qu'il est impossible pour les axes de rotation de l'organe porte-outil d'atteindre l'orientation de l'outil calculée sur la base de l'orientation du frame, une alarme est émise.

Remarque

Combiner la correction fine d'outil et les fonctionnalités de la correction de longueur d'outil n'est pas autorisé pour les organes porte-outils orientables. Si vous tentez d'appeler les deux fonctions en même temps, un message d'erreur vous sera donné.

Remarque

Avec TOFRAME (Page 339), il est possible de définir un frame en fonction de la direction dans laquelle est orienté l'organe porte-outil sélectionné.

Remarque

Quand une transformation d'orientation est active (transformation 3, 4, 5 axes), vous pouvez appeler un organe porte-outil ayant une orientation s'écartant de la position zéro, sans déclencher une alarme.

Remarque

Avec un frame actif sans rotation, l'orientation avec TCOFRX/TCOFRY/TCOFRZ mène à une solution triviale, car l'orientation de l'outil pointe déjà dans la direction X/Y/Z.

Orientation dans la direction Z (TOFRAME)

Avec l'instruction TOFRAME du groupe G 53 (Rotation de frame par rapport à l'outil), il est possible de définir un frame dont la direction Z est orientée parallèlement à la direction d'orientation de l'organe porte-outil sélectionné.

Si aucun organe porte-outil n'est actif ou si un organe porte-outil sans modification de l'orientation est actif, la direction Z est comme suit dans le nouveau frame :

- Pour G17, similaire à l'ancienne direction Z.
- Pour G18, similaire à l'ancienne direction Y.
- Pour G19, similaire à l'ancienne direction X.

Paramètres de transfert des cycles standard et cycles de mesure

Pour les paramètres de transfert des cycles standard et cycles de mesure, on applique les plages de valeurs bien définies.

Les plages de valeurs suivantes sont définies pour les valeurs angulaires :

Rotation autour...	Plage de valeurs
1er axe géométrique	-180° ≤ Angle ≤ +180°
2e axe géométrique	-90° ≤ Angle ≤ +90°
3e axe géométrique	-180° ≤ Angle ≤ +180°

Remarque

Lors du transfert de valeurs angulaires à un cycle standard ou de mesure, il convient

Les valeurs inférieures à l'unité de calcul de la CN doivent être arrondies à zéro !

L'unité de calcul de la CN pour les positions angulaires est définie dans le paramètre machine :

PM10210 $MN_INT_INCR_PER_DEG

3.13.10 Modification des organes porte-outil orientables après mesurage de la machine (CORRTC)

La fonction `CORRTC` permet d'écrire les éléments de chaîne cinématique mesurés d'un porte-outil dans des éléments de correction spéciaux.

Remarque

Les valeurs de correction écrites avec la fonction CORRTC ne sont pas immédiatement effectives dans le porte-outil. Les valeurs de correction prennent seulement effet après désactivation du porte-outil, NEWCONF et activation du porte-outil.

Syntaxe

```
<_Corr_Status> = CORRTC(<_Corr_Vect>, <_Corr_Index>, <_Corr_Mode>,
[ <_No_Alarm>])
```

Signification

CORRTC:	Appel de fonction		
<_Corr_Status> :	Valeur de retour de la fonction		
	Type de données :	INT	
	Valeurs :	0	La fonction a été exécutée sans erreur.
		1	Aucun porte-outil n'est actif.
		2	Le porte-outil actif n'a pas été défini avec des chaînes cinématiques.
		10	Le paramètre d'appel <_Corr_Index> est négatif.
		11	Le paramètre d'appel <_Corr_Mode> est négatif.
		12	Renvoi non valide sur la section d'une chaîne partielle (_CORR_INDEX).
		13	Aucun élément de correction n'est défini dans la section référencée par le paramètre _CORR_INDEX ($TC_CARR_CORR_ELEM).
		20	La position des centaines de _CORR_MODE n'est pas valide. Seules les valeurs 0 et 1 sont autorisées.
		21	La position des milliers de _CORR_MODE n'est pas valide. Seules les valeurs 0 et 1 sont autorisées.
		30	Lors de la correction d'un vecteur d'offset, l'écart dans au moins une coordonnée par rapport à la valeur actuelle est supérieur à la valeur maximale spécifiée par la donnée de réglage SD41612 $SN_CORR_TOCARR_LIN_MAX.
		31	La tentative d'écrire une variable système a été refusée pour cause de droits d'écriture insuffisants.
<_Corr_Vect> :	Vecteur de correction Le contenu du vecteur de correction est défini par les paramètres <Corr_Index> et <_Corr_Mode> suivants. Si <_Corr_Status> = 30, le contenu du vecteur est écrasé (voir ci-dessus).		
	Type de données :	REAL	
<_Corr_Index> :	Désigne la section pour laquelle le vecteur de direction de l'élément de correction doit être corrigé.		
	Type de données :	INT	

<_Corr_Mode> :	Mode de correction		
	Type de données :	INT	
	Le paramètre <Corr_Mode> est codé en décimal (position des unités jusqu'à la position des milliers) :		
	Position des unités :	Réservé	
	Position des dizaines :	Définit la façon dont l'élément de correction référencé par le contenu de <_Corr_Index> doit être modifié.	
		xx**0**x	Le vecteur de correction est directement écrit dans l'élément de correction. Cette variante peut être utilisée pour écrire directement l'élément de correction sans que l'indice <n> de la donnée système ($NK_OFF_DIR[<n>, ...]) soit nécessairement connu.
		xx**1**x	Comme 0, avec la différence que la valeur de correction transmise est interprétée en coordonnées mondiales. Une différence entre les variantes 0 et 1 peut apparaître lorsque la chaîne cinématique contient d'autres rotations en position de base (position de tous les axes rotatifs égale à 0).
		xx**2**x	Comme 1, avec la différence que la valeur de correction s'applique sur la totalité de la section. Autrement dit, la valeur renseignée dans l'élément de correction est telle que la totalité de la section atteint la longueur définie par la valeur de correction.
	Position des centaines :	Définit la façon dont le contenu du paramètre <_Corr_Vect> doit être interprété.	
		x**0**xx	Le vecteur de correction transmis <_Corr_Vect> contient la totalité de la nouvelle longueur de l'élément de correction ou de la section référencée par <_Corr_Index> en association avec la position des dizaines de <_Corr_Mode> (correction absolue).
		x**1**xx	Le vecteur de correction transmis <_Corr_Vect> ne contient que la différence par rapport à la longueur actuelle de l'élément de correction ou de la section référencée par <_Corr_Index> en association avec la position des dizaines de <_Corr_Mode> (correction incrémentale).
	Position des milliers :	Définit si la correction maximale admissible doit être limitée ou non par la donnée de réglage $SN_ CORR_TOCARR_LIN_MAX.	
		0xxx	La surveillance de seuils est active.
		1xxx	La surveillance de seuils est inhibée.
<_No_Alarm> :	Comportement en cas d'erreur (valeur retournée > 0) (**facultatif**)		
	Type de données :	BOOL	
	Valeur :	FALSE (Default)	En cas d'erreur, l'exécution du programme est arrêtée et une alarme est émise.
		TRUE	En cas d'erreur, l'exécution du programme n'est pas arrêtée et aucune alarme n'est affichée. Cas d'application : réaction personnalisée en fonction de la valeur de retour

Informations complémentaires sur CORRTC

La structure cinématique d'un porte-outil est décrite par une (type T et type P) ou deux (type M) chaînes cinématiques (chaînes partielles) qui partent du point de référence correspondant (point de référence de la machine ou point de référence de l'organe porte-outil). L'une des deux chaînes, la chaîne outil, se termine au point de référence de l'outil, l'autre, la chaîne pièce dans l'origine du système de coordonnées de base.

La fonction CORRTC écrit les directions d'axe des machines équipées de porte-outils dans des éléments de correction spéciaux. Une chaîne cinématique est décrite entre autres par des éléments de type OFFSET définis avec $NK_TYPE.

CORRTC utilise des sections

Les deux chaînes partielles peuvent être divisées chacune en quatre sections au maximum :

- La section 1 commence au point de départ de la chaîne et se termine au premier axe rotatif.
- La section 2 est la section comprise entre l'axe rotatif 1 et l'axe rotatif 2.
- La section 3 est la section comprise entre l'axe rotatif 2 et la fin de la chaîne.

Le graphique suivant présente un organe porte-outil orientable avec 2 axes rotatifs.

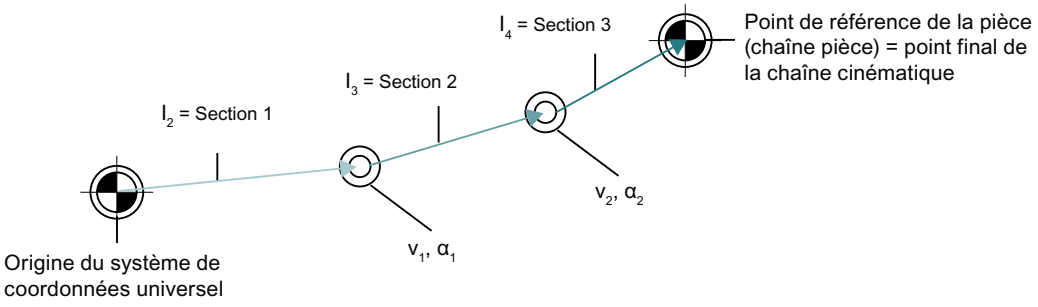


Figure 3-11 Exemple de CORRTC

Les sections sont définies de manière univoque : Lorsqu'on parcourt la chaîne cinématique partielle depuis son point de départ jusqu'à la fin, la première section comporte l'indice 1, la suivante l'indice 2, etc.

Éléments de correction

Dans chacune de ces sections, la variable système $TC_CARR_CORR_ELEM [, 0 ... 3] permet de référencer un élément constant de chaîne cinématique (du type $NK_TYPE[<n>] = "OFFSET"). Les valeurs de correction déterminées lors du mesurage de la machine sont écrites avec la fonction CORRTC dans ces éléments ainsi désignés.

L'ordre des références dans $TC_CARR_CORR_ELEM[m, 0..3] doit correspondre aux sections susmentionnées. Autrement dit, $TC_CARR_CORR_ELEM[m, 0] ne peut contenir qu'un élément de chaîne appartenant au vecteur d'offset **l**1, etc.

La valeur de référence est toujours la valeur correspondante qui est effective lors de l'appel de l'organe porte-outil actif de CORRTC. Les contenus des données de cinématique modifiés dans la gestion des données après l'activation de l'organe porte-outil n'ont aucune incidence sur les effets de la fonction CORRTC.

3.13.11 Correction en ligne de la longueur d'outil (TOFFON, TOFFOF)

La variable système $AA_TOFF[<n>] permet de corriger en temps réel et en trois dimensions les longueurs d'outil effectives en fonction des trois directions de l'outil.

Les trois descripteurs d'axes géométriques servent d'indices <n>. De cette façon, le nombre des directions de correction activées est déterminé par les axes géométriques qui sont actifs au même moment.

Toutes les corrections peuvent être actives simultanément.

La fonction de correction des longueurs d'outil en ligne est applicable dans :

- Transformation de l'orientation TRAORI
- Organes porte-outil orientables TCARR

Remarque

La correction en ligne de longueur d'outil est une **option** qui doit être validée auparavant. Cette fonction n'a de sens qu'en liaison avec la transformation active de l'orientation ou avec un organe porte-outil orientable à l'état activé.

Syntaxe

```
TRAORI
TOFFON(<direction de correction>[,<valeur de l'off-
set>])
WHEN TRUE DO $AA_TOFF[<sens de correction>]             ; Dans des actions synchro-
                                                        nes.
...
TOFFOF(<sens de correction>)
```

Signification

TOFFON :	**Activation** de la correction en ligne de la longueur d'outil	
	<sens de correction> :	Direction de l'outil (X, Y, Z) dans laquelle la correction en ligne de la longueur d'outil doit prendre effet.
	<valeur de l'offset> :	Dès que la fonction est activée, une valeur offset peut être indiquée pour la direction dans laquelle doit s'effectuer la correction et cette valeur fait immédiatement l'objet d'un mouvement de correction.
TOFFOF :	**Remise à zéro** de la correction en ligne de la longueur d'outil Les valeurs de correction dans le sens de correction spécifié sont remises à zéro et un arrêt du prétraitement des blocs est déclenché.	

Exemples

Exemple 1 : activation de la correction de longueur d'outil

Code de programme	Commentaire
PM21190 $MC_TOFF_MODE =1	; Accostage des valeurs absolues.
PM21194 $MC_TOFF_VELO[0] =1000	
PM21196 $MC_TOFF_VELO[1] =1000	
PM21194 $MC_TOFF_VELO[2] =1000	
PM21196 $MC_TOFF_ACCEL[0] =1	
PM21196 $MC_TOFF_ACCEL[1] =1	
PM21196 $MC_TOFF_ACCEL[2] =1	
N5 DEF REAL XOFFSET	
N10 TRAORI(1)	; Activation de la transformation.
N20 TOFFON(Z)	; Activation de la correction en ligne de la longueur d'outil pour la direction Z de l'outil.
N30 WHEN TRUE DO $AA_TOFF[Z]=10 G4 F5	; Interpolation d'une valeur de correction de longueur d'outil égale à 10 pour la direction Z de l'outil.
...	
N100 XOFFSET=$AA_TOFF_VAL[X]	; Affectation de la correction actuelle dans la direction X.
N120 TOFFON(X,-XOFFSET) G4 F5	; Remise à 0 de la correction de longueur d'outil dans la direction X de l'outil.

Exemple 2 : Désactivation de la correction de longueur d'outil

Code de programme	Commentaire
N10 TRAORI(1)	; Activation de la transformation.
N20 TOFFON(X)	; Activation de la correction en ligne de la longueur d'outil pour la direction X de l'outil.
N30 WHEN TRUE DO $AA_TOFF[X]=10 G4 F5	; Interpolation d'une valeur de correction de longueur d'outil égale à 10 pour la direction X de l'outil.
...	
N80 TOFFOF(X)	; L'offset de position de la direction d'outil X est supprimé : ...$AA_TOFF[X]=0 Aucun axe n'est déplacé. L'offset de position est ajouté à la position actuelle dans le SCP en fonction de l'orientation actuelle.

Informations complémentaires

Préparation des blocs

Lors de la préparation des blocs au cours de leur prétraitement, l'offset de longueur d'outil qui agit dans l'exécution de blocs est pris en considération. Afin de pouvoir exploiter au mieux les vitesses d'axes maximales autorisées, il est nécessaire d'arrêter la préparation des blocs avec un arrêt du prétraitement des blocs STOPRE pendant la réalisation d'un offset d'outil.

Cet offset d'outil est connu au moment du prétraitement des blocs si les corrections de longueur d'outil n'ont pas été modifiées après le lancement du programme ou - quand il y a eu modification des corrections de longueur d'outil - si le nombre de blocs exécutés est supérieur au nombre de blocs que peut recevoir le tampon IPO entre le prétraitement des blocs et leur exécution.

Variable $AA_TOFF_PREP_DIFF

Le degré de différence entre la correction actuellement effective dans l'interpolateur et la correction effective au moment de la préparation des blocs peut être interrogée dans la variable $AA_TOFF_PREP_DIFF[<n>].

Réglage des paramètres machine et des données de réglage

Les données système suivantes sont disponibles pour la correction en ligne de la longueur d'outil :

- MD20610 $MC_ADD_MOVE_ACCEL_RESERVE (réserve d'accélération pour déplacement forcé)
- MD21190 $MC_TOFF_MODE
  Le contenu de la variable système $AA_TOFF[<n>] fait l'objet d'un déplacement en absolu ou d'une intégration forcée.
- MD21194 $MC_TOFF_VELO (vitesse de la correction en ligne de la longueur d'outil)
- MD21196 $MC_TOFF_ACCEL (accélération de la correction en ligne de la longueur d'outil)
- Donnée de réglage pour la spécification de valeurs limites :
  SD42970 $SC_TOFF_LIMIT (limite supérieure de la valeur de correction de la longueur d'outil)

3.13.12 Modification des données de correction pour outils orientables

3.13.12.1 Calcul de l'orientation (ORISOLH)

La fonction ORISOLH prédéfinie assiste l'utilisateur au réglage des positions d'axe rotatif d'une machine, de telle sorte qu'un outil de tournage soit orienté dans une position définie, relative à la pièce et indépendante de la cinématique. Une transformation 6 axes, paramétrée avec des chaînes cinématiques, doit être préalablement active.

Deux fonctions de base sont disponibles :

- Orienter l'outil
  Les deux angles β et γ sont spécifiés. La fonction calcule alors les angles des trois axes d'orientation nécessaires.
- Orienter directement l'outil
  Les angles du deuxième et troisième axe d'orientation sont spécifiés. La fonction calcule alors les angles correspondants β et γ ainsi que l'angle du premier axe d'orientation encore manquant.

Remarque

Ordre des axes d'orientation

En parcourant la chaîne cinématique qui décrit la structure de la machine en allant de la pièce jusqu'à l'outil, les règles suivantes s'appliquent pour l'ordre des trois axes d'orientation d'une transformation 6 axes :

- L'axe d'orientation le plus proche de la **pièce** est le **premier** axe d'orientation.
- L'axe d'orientation le plus proche de l'**outil** est le **troisième** axe d'orientation.

Le premier axe d'orientation est habituellement une broche, et la rotation correspondante est réalisée dans ces cas par un frame rotatif.

Syntaxe

```
<RetVal> = ORISOLH(<Cntrl>,<W1>,<W2>)
```

Signification

ORISOLH :	Appel de fonction		
<RetVal> :	Valeur de retour de la fonction		
	Type de données :	INT	
	Plage de valeurs :	0, -2, -3, ..., -17	
	Valeurs :	0	La fonction a été terminée sans erreur.
		-2	Aucune transformation valide (transformation d'orientation 6 axes) n'est active.
		-3	Le premier paramètre (<Cntrl>) est négatif.
		-4	La position des unités du premier paramètre (<Cntrl>) n'est pas valide. Seules les valeurs 0 et 1 sont autorisées.
		-5	La position des dizaines du premier paramètre (<Cntrl>) n'est pas valide. Seules les valeurs 0 et 3 sont autorisées.
		-6	La position des centaines du premier paramètre (<Cntrl>) n'est pas valide. Seules les valeurs 0 et 1 sont autorisées.
		-7	La position des milliers du premier paramètre (<Cntrl>) n'est pas valide. Seules les valeurs 0 et 3 sont autorisées.
		-8	Dans la fonction "Orienter directement l'outil", l'angle γ est trop grand.
		-9	Au moins une des positions d'axe spécifiées dépasse une limite d'axe dans la fonction "Orienter directement l'outil".
		-10	Aucun outil n'est actif.
		-11	L'orientation demandée ne peut être réglée.
		-12	L'adaptation des angles d'axe libres pour denture Hirth est impossible pour la première et seule solution.
		-13	L'adaptation des angles d'axe libres pour denture Hirth est impossible pour la deuxième solution.
		-14	L'adaptation des angles d'axe libres pour denture Hirth est impossible pour aucune des deux solutions.
		-15	Le premier axe d'orientation est paramétré comme axe Hirth.
		-16	Le deuxième et le troisième axe rotatif sont paramétrés comme axes Hirth. Un des deux axes au maximum doit être un axe Hirth.
		-17	Pour la fonction "Pivotement direct", au moins une des positions d'axes spécifiées n'est pas compatible avec la denture Hirth correspondante.

<Cntrl> :	Commande le comportement de la fonction		
	Type de données :	INT	
	Le paramètre <Cntrl> est codé en décimal (position des unités jusqu'à la position des milliers) :		
	Position des unités :	La position des unités commande le comportement en cas d'erreur.	
		xxx**0**	En cas d'erreur (valeur retournée < 0), l'exécution du programme est annulée et l'alarme 14106 est émise. **Remarque :** Indépendamment de la valeur de la position des unités, l'alarme est également émise lorsque le paramètre <Cntrl> est négatif.
		xxx**1**	En cas d'erreur (valeur retournée < 0), **aucune** alarme n'est émise. L'utilisateur a la possibilité de programmer sa propre réaction.
	Position des dizaines :	Commande de comportement lors de la présence d'un axe d'orientation avec denture Hirth. **Remarque :** Ce paramètre n'est évalué que dans le contexte de la fonction "Orienter l'outil" (c.-à-d. lorsque la valeur de la position des centaines est "0").	
		xx**0**x	La position de l'axe est arrondie à la position la plus proche.
		xx**1**x	Les positions d'axe sont arrondies de telle sorte que l'écart de l'angle β par rapport à sa valeur programmée devienne minimal.
		xx**2**x	Les positions d'axe sont arrondies de telle sorte que l'angle β prenne la valeur maximale inférieure à la valeur programmée (β arrondi vers le bas).
		xx**3**x	Les positions d'axe sont arrondies de telle sorte que l'angle β prenne la valeur minimale supérieure à la valeur programmée (β arrondi vers le haut).
	Position des centaines :	Spécifie la fonction qui doit être exécutée ou la signification des deux paramètres suivants <W1> et <W2>.	
		x**0**xx	Fonction "Orienter l'outil" Les paramètres <W1> et <W2> ont la signification suivante : • <W1> = β • <W2> = γ Les angles correspondants des axes d'orientation sont calculés.
		x**1**xx	Fonction "Orienter directement l'outil" <W1> est la position spécifiée pour le deuxième axe d'orientation, <W2> est la position spécifiée pour le troisième axe d'orientation d'une transformation 6 axes. La fonction détermine la position du premier axe d'orientation ainsi que les angles β et γ, compatibles avec les deux spécifications de position. En l'absence d'erreur, la variable système $P_ORI_POS[<n>, <m>] fournit toujours deux solutions. Le premier indice <n> (0 ou 1) renvoie à la solution et le deuxième indice <m> (0 ... 2) à l'axe d'orientation : • $P_ORI_POS[0/1, **0**] : Position du premier axe d'orientation

			• $P_ORI_POS[0/1, **1**] : Angle β • $P_ORI_POS[0/1, **2**] : Angle γ La fonction vérifie si les positions spécifiées <W1> et <W2> sont compatibles avec des dentures Hirth ou limites logicielles éventuellement actives. Si ce n'est pas le cas, un numéro d'erreur correspondant est retourné (voir paramètre <RetVal>). Lors d'une sélection quelconque des deux angles <W1> et <W2>, le tranchant de l'outil ne se trouve généralement pas dans le plan d'usinage. L'angle γ formant l'écart entre le tranchant et le plan d'usinage ne doit pas dépasser la valeur limite définie par la donnée de réglage SD42999 $SC_ORISOLH_INCLINE_TOL.
	Position des milliers :	Spécifie comment les positions des solutions sont éventuellement modifiées lorsque la valeur de la position des centaines est "0", c.-à-d. pour la fonction "Orienter l'outil".	
		0xxx	Les positions d'axe calculées doivent être les plus proches possibles des positions d'axe machine.
		1xxx	Dans le cas des axes à valeur modulo, les positions d'axe calculées doivent être les plus proches possibles du milieu de la plage modulo, dans le cas des autres axes les plus proches possibles de 0. Cela signifie pour les axes non modulo que les positions d'axe sont limitées à la plage -180° ... +180°.
		2xxx	Les positions d'axe calculées doivent être réduites à la plage -180° ... +180° indépendamment du type d'axe.
<W1> :	Premier angle La signification est déterminée par la position des centaines du paramètre <Cntrl>.		
	Type de données :	REAL	
<W1> :	Deuxième angle La signification est déterminée par la position des centaines du paramètre <Cntrl>.		
	Type de données :	REAL	

Remarque

Les paramètres non programmés ont par défaut la valeur "0".

Informations complémentaires

Les variables système suivantes permettent de lire le nombre de solutions trouvées ainsi que les informations d'état relatives à l'exécution de la fonction ORISOLH :

Variable système	Signification		
$P_ORI_POS [<n>, <m>]	Fournit l'angle des axes d'orientation résultant de la programmation de l'orientation.		
	<n> :	Indice de la solution	
		Plage de valeurs :	0, 1
	<m> :	Indice de l'axe d'orientation	
		Plage de valeurs :	0 ... 2
			L'ordre des axes d'orientation (1 ... 3) se rapporte à la définition des axes dans $NT_ROT_AX_NAME.
	Lors de l'appel de la fonction ORISOLH en mode "Orienter directement l'outil", les variables $P_ORI_POS[0/1, 1] et P_ORI_POS[0/1, 2] contiennent les valeurs des deux angles β et γ des deux solutions. La première solution enregistrée dans $P_ORI_POS[<n>, <m>], portant l'indice <n> = 0, est toujours la solution sélectionnée par la commande lorsque l'orientation demandée est approchée directement. Le deuxième indice <m> se rapporte à l'axe d'orientation, c.-à-d. $NT_ROT_AX_NAME. Les positions d'axe enregistrées dans $P_ORI_POS[<n>, <m>] tiennent compte des offsets renseignés dans $NK_OFF et $NK_OFF_FINE, c.-à-d. ces angles d'axe peuvent être utilisés sans aucune modification dans les blocs suivants pour régler l'orientation souhaitée. Si un axe rotatif est un axe Hirth, les positions de la solution sont arrondies à la position indexée la plus proche de la denture Hirth. Dans le cas des axes rotatifs avec denture Hirth, la variable système $P_ORI_DIFF permet de lire les différences des positions d'axe par rapport à la solution exacte ainsi que les solutions adaptées à l'indexage Hirth.		
$P_ORI_DIFF [<n>, <m>]	Fournit la différence entre les positions exactes des axes d'orientation et celles fournies par $P_ORI_POS résultant de la programmation de l'orientation.		
	<n> :	Indice de la solution	
		Plage de valeurs :	0, 1
	<m> :	Indice de l'axe d'orientation	
		Plage de valeurs :	0 ... 2
			L'ordre des axes d'orientation (1 ... 3) se rapporte à la définition des axes dans $NT_ROT_AX_NAME.
	Le contenu ne peut être différent de zéro que si les positions sont indexées (denture Hirth), c.-à-d. lorsque la donnée système $NT_HIRTH_INCR de l'axe concerné est différente de zéro et qu'il s'agit d'un axe rotatif manuel.		

Variable système	Signification		
$P_ORI_SOL	Lorsque, pour une transformation d'orientation avec plusieurs axes d'orientation, les angles d'axe sont calculés pour obtenir une orientation spécifiée, il existe généralement plusieurs solutions. La variable système $P_ORI_SOL fournit le nombre de solutions valides ainsi que des informations d'état supplémentaires. Le contenu de $P_ORI_SOL est codé de la manière suivante :		
	Valeurs < 0	États d'erreur généraux	
		-1	Aucune solution n'a encore été calculée pour la transformation active (appel d'ORISOLH manquant).
		-2	Aucune transformation n'est active ou la transformation active n'est pas une transformation d'orientation (transformation 6 axes) pouvant fournir des positions pour une programmation spécifiée de l'orientation.
		-4	L'orientation souhaitée n'est pas réalisable avec la cinématique donnée.
		-5	Aucune solution n'a été trouvée lors de l'appel de la fonction ORISOLH en mode "Orienter directement l'outil".
		-6	Lors de l'appel de la fonction ORISOLH en mode "Orienter directement l'outil", l'angle γ est trop grand.
		-7	Lors de l'appel de la fonction ORISOLH en mode "Orienter directement l'outil", un angle a été spécifié qui ne peut pas être réglé en raison de la denture Hirth.
		-8	Le premier axe d'orientation (axe frame) ne doit pas être paramétré comme axe Hirth.
		-9	Le deuxième et le troisième axe rotatif sont paramétrés comme axes Hirth. Un des deux axes au maximum doit être un axe Hirth.
		-10	Aucune adaptation de la / des solution(s) à l'axe Hirth n'a été trouvée.
	Valeurs > 0 Position des unités	Nombre de solutions mathématiquement possibles sans tenir compte des limites d'axe et d'éventuelles conditions d'erreur.	
		0	Il n'existe aucune solution, c.-à-d. l'orientation demandée ne peut être réglée. Ce cas de figure peut avoir trois causes différentes : • En raison de la cinématique de la machine (positionnement non orthogonal des axes d'orientation), l'orientation n'est par principe pas atteignable indépendamment de la plage de déplacement des axes d'orientation. Dans ce cas, les positions des dizaines et des centaines de $P_ORI_SOL sont égales à zéro et les variables d'état $P_ORI_STAT affectées aux axes d'orientation contiennent la valeur "-4". • Les solutions calculées ne peuvent pas être atteintes car elles dépasseraient les limitations d'axe. Les positions des axes d'orientation qui résulteraient sans limitation d'axe sont fournies dans $P_ORI_POS. • Lors de l'appel de la fonction ORISOLH en mode "Orienter directement l'outil", les positions d'axe ont été spécifiées de telle sorte que le vecteur d'orientation ou le vecteur normal d'orientation de l'outil est parallèle au premier axe d'orien-

Variable système	Signification		
			tation dont la position doit être calculée. La position de cet axe n'est pas définie dans ces cas.
		1	Il n'existe pas de solution. Ce cas de figure peut avoir trois causes différentes : • En raison de l'orientation spécifiée et de la cinématique de la machine, il n'existe qu'une seule solution, même sans tenir compte des limitations d'axe (du point de vue mathématique, il s'agit de deux solutions identiques). Cette situation se produit, dans le cas de cinématiques non orthogonales, à la limite de la plage d'orientation. $P_ORI_POS contient les deux solutions (identiques). • Il n'existe qu'une seule solution, car la deuxième solution n'est pas valide en raison d'un dépassement des limites d'axe. La solution valide est alors toujours la première solution dans $P_ORI_POS. La deuxième solution, calculée sans prise en compte des limites d'axe, peut également être lue dans $P_ORI_POS. • C'est le cas normal lors de l'appel de la fonction ORISOLH en mode "Orienter directement l'outil". Lorsque les positions d'axe sont spécifiées pour deux axes d'orientation, il n'existe généralement qu'une seule position valide pour les axes d'orientation manquants à calculer.
		2	Il existe deux solutions.
		8	Il existe un nombre infini de solutions ; autrement dit, la position d'un axe d'orientation (axe polaire) est indifférente. Concernant le ou les autre(s) axe(s), l'une des positions possibles est exclue en raison d'un dépassement des limites d'axe.
		9	Il existe un nombre infini de solutions ; autrement dit, la position d'un axe d'orientation (axe polaire) est indéfinie. La position des centaines de la variable système $P_ORI_STAT identifie l'axe indéfini.
	Valeurs > 0 Position des dizaines	Indication codée sur bits des limites d'axe dépassées. La variable système $P_ORI_STAT permet de déterminer la cause exacte de l'erreur.	
		Bit 0 (valeur 10) :	Au moins une limite d'axe du 1er axe d'orientation a été dépassée dans l'une des solutions.
		Bit 1 (valeur 20) :	Au moins une limite d'axe du 2e axe d'orientation a été dépassée dans l'une des solutions.
		Bit 2 (valeur 40) :	Au moins une limite d'axe du 3e axe d'orientation a été dépassée dans l'une des solutions.
	Valeurs > 0 Position des centaines	Indication codée sur bits des positions d'axe non définies (uniquement possible s'il existe un nombre infini de solutions, c.-à-d. si la position des unités est égale à "9").	
		Bit 0 (valeur 100) :	La position du 1er axe d'orientation n'est pas définie.
		Bit 1 (valeur 200) :	La position du 2e axe d'orientation n'est pas définie.
		Bit 2 (valeur 400) :	La position du 3e axe d'orientation n'est pas définie.

Variable système	Signification
	Les désignations "1er, 2e et 3e axe d'orientation" se rapportent aux définitions des axes dans $NT_ROT_AX_NAME.

Variable système	Signification			
$P_ORI_STAT [<n>]	Fournit l'état pour chacun des 3 axes d'orientation (max.) après l'appel d'ORISOLH.			
	<n> :	Indice de l'axe d'orientation (correspond à l'indice de l'axe d'orientation concerné dans $NT_ROT_AX_NAME)		
		Plage de valeurs :	0 ... 2	
			L'ordre des axes d'orientation (1 ... 3) se rapporte à la définition des axes dans $NT_ROT_AX_NAME.	
	Le contenu de $P_ORI_STAT est codé de la manière suivante :			
	Valeurs < 0	États d'erreur généraux		
		-1	L'état n'est pas défini (appel manquant d'ORISOLH).	
		-2	Aucune transformation n'est active ou la transformation active n'est pas une transformation d'orientation (transformation 6 axes) pouvant fournir des positions pour une programmation spécifiée de l'orientation.	
		-3	L'axe ne fait pas partie de la transformation active.	
		-4	Impossible de calculer la position de l'axe, car l'orientation souhaitée n'est pas réalisable avec la cinématique donnée, même en supposant une plage de déplacement quelconque.	
		-5	Lors de l'appel de la fonction ORISOLH en mode "Orienter directement l'outil", les positions d'axe ont été spécifiées de telle sorte que le vecteur d'orientation ou le vecteur normal d'orientation de l'outil est parallèle au premier axe d'orientation dont la position doit être calculée. La position de cet axe n'est pas définie dans ces cas.	
		-6	Lors de l'appel de la fonction ORISOLH en mode "Orienter directement l'outil", l'angle γ est trop grand.	
		-7	Lors de l'appel de la fonction ORISOLH en mode "Orienter directement l'outil", un angle a été spécifié qui ne peut pas être réglé en raison de la denture Hirth.	
		-8	Le premier axe d'orientation (axe frame) ne doit pas être paramétré comme axe Hirth.	
		-9	Le deuxième et le troisième axe rotatif sont paramétrés comme axes Hirth. Un des deux axes au maximum doit être un axe Hirth.	
		-10	Aucune adaptation de la / des solution(s) à l'axe Hirth n'a été trouvée.	
	Valeurs > 0 Position des unités	Indication codée sur bits des limites d'axe dépassées de la première solution.		
		Bit 0 (valeur 1) :	La première solution dépasse la limite d'axe inférieure.	
		Bit 1 (valeur 2) :	La première solution dépasse la limite d'axe supérieure.	
	Valeurs > 0 Position des dizaines	Indication codée sur bits des limites d'axe dépassées de la deuxième solution.		
		Bit 0 (valeur 10) :	La deuxième solution dépasse la limite d'axe inférieure.	
		Bit 1 (valeur 20) :	La deuxième solution dépasse la limite d'axe supérieure.	

Variable système	Signification		
	Valeurs > 0 Position des centaines	Indication d'une position d'axe non définie.	
		Bit 0 (valeur 100) :	La position de l'axe d'orientation n'est pas définie, c.-à-d. l'orientation demandée peut être réalisée avec tous les réglages possibles de l'axe rotatif (position polaire). Cette information est également présente dans la variable système $P_ORI_SOL.
	Plusieurs des numéros d'erreur indiquant un dépassement des limites d'axe peuvent apparaître simultanément. Lors d'un dépassement d'une limite d'axe, la fonction essaie d'atteindre une position à l'intérieur des limites d'axe autorisées en additionnant ou soustrayant des multiples de 360°. Si cela est impossible, le résultat ne permet pas de déterminer de manière univoque si la limite supérieure ou inférieure a été dépassée. Si aucune solution existe pour l'orientation demandée ($P_ORI_SOL = 0), l'état des axes d'orientation contenus dans la transformation est égal à "0".		

Remarque

$NT_ROT_AX_NAME

Cette variable système identifie un maximum de 3 axes utilisés pour le réglage de l'orientation. Elle contient les noms des éléments de chaîne ($NK_NAME) définissant les axes machine (axes rotatifs) qui doivent exécuter les déplacements d'orientation résultant d'une transformation cinématique. L'ordre des trois axes rotatifs (maximum) contenus dans la variable système est sans importance pour la cinématique de la machine, car il est dérivé de la structure des chaînes cinématiques. Toutefois, comme cet ordre définit l'ordre d'accès des autres variables aux axes rotatifs, l'ordre des axes d'orientation dans $NT_ROT_AX_NAME doit correspondre à la description de la cinématique.

Remarque

Informations d'état

Les informations d'état, signalant par ex. qu'une orientation n'est pas réalisable ou que sa réalisation entraînerait un dépassement des limites d'axe en question, ne génèrent pas d'alarme CN. Il relève de la responsabilité de l'utilisateur de réagir de manière appropriée aux conditions signalées.

3.13.12.2 Activation de la modification des données de correction pour outils orientables (CUTMOD, CUTMODK)

La modification des données de correction pour outils orientables est activée dans le programme CN par l'instruction CUTMOD (en association avec les organes porte-outil orientables) ou CUTMODK (pour les transformations d'orientation définies à l'aide de chaînes cinématiques).

Remarque

Comme les organes porte-outil orientables et les transformations d'orientation définies à l'aide de chaînes cinématiques ne peuvent pas être actifs simultanément, aucun conflit n'apparaît entre les deux variantes.

Syntaxe

```
CUTMOD = <Value>
```

ou

```
CUTMODK = <Command>
```

Signification

CUTMOD :	Appel de fonction dans le contexte des organes porte-outil orientables		
<Value> :	Valeur affectée		
	Type de données :	INT	
	Valeur :	0	La fonction est désactivée. Les valeurs fournies par les variables système $P_AD... sont égales aux paramètres d'outil correspondants.
		> 0	La fonction est activée si un organe porte-outil orientable portant le numéro indiqué est actif, ce qui signifie que l'activation est liée à un organe porte-outil orientable défini. Par rapport aux paramètres d'outil correspondants, les valeurs fournies par les variables système $P_AD... sont modifiées le cas échéant en fonction de la rotation active. La désactivation de l'organe porte-outil orientable désigné désactive la fonction temporairement et l'activation d'un autre organe porte-outil orientable la désactive durablement. Dans le premier cas, la fonction est donc de nouveau active si le même organe porte-outil orientable est réactivé. Dans le deuxième cas, il est nécessaire de l'activer à nouveau même si l'organe porte-outil orientable portant le numéro indiqué est réactivé ultérieurement. La fonction n'est pas influencée par un reset.
		-1	La fonction est toujours activée lorsqu'un organe porte-outil orientable est actif. Lors d'un changement de l'organe porte-outil ou lorsque celui-ci est désactivé, puis réactivé ultérieurement, il n'est pas nécessaire de réactiver CUTMOD.
		-2	La fonction est toujours activée lorsqu'un organe porte-outil orientable dont le numéro est identique à l'organe porte-outil orientable actif à l'instant est actif. Si aucun organe porte-outil orientable n'est actif, cela équivaut à CUTMOD=0. Si un organe porte-outil orientable est actif, cela équivaut à l'indication directe du numéro de l'organe porte-outil courant.
		< -2	Les valeurs inférieures à 2 sont ignorées, ce qui signifie que ce cas est traité comme si CUTMOD n'était pas programmé. **Remarque :** L'utilisation de cette plage de valeurs est déconseillée, car elle est réservée pour d'éventuelles extensions ultérieures.
CUTMODK :	Appel de fonction dans le contexte des transformations d'orientation définies à l'aide de chaînes cinématiques		

<Command> :	Ordre affecté		
	Type de données :	STRING	
	Valeur :	"NEW"	Seuls les états pertinents pour la "Modification des données de correction" d'une transformation active définie à l'aide de chaînes cinématiques, de même que le nom de la transformation et le frame de contour sont enregistrés. **Remarque :** Cet ordre n'est autorisé que si une transformation appropriée (TRAORI_DYN, TRAORI_STAT ou TRAANG_K) est active.
		"OFF"	Désactive la "Modification des données de correction" active. Les données précédemment enregistrées avec "NEW" sont conservées. **Remarque :** Cet ordre n'est autorisé que si CUTMODK n'est pas active. Il reste alors sans effet. Un jeu de paramètres éventuellement existant pour la "Modification des données de correction" est conservé.
		"ON"	Avec cet ordre, la "Modification des données de correction" est de nouveau activée avec un jeu de paramètres enregistré avec l'ordre "NEW". Lors de l'exécution de cet ordre, si une transformation du même nom que le jeu de paramètre enregistré est active, la "Modification des données de correction" est immédiatement effective. Sinon l'activation est retardée jusqu'à ce qu'une transformation active soit activée.
		"CLEAR"	Désactive, tout comme l'ordre "OFF", la "Modification des données de correction" et supprime en outre le jeu de paramètres enregistré. **Remarque :** Cet ordre n'est autorisé que si CUTMODK n'est pas active.

Remarque

SD42984 $SC_CUTDIRMOD

La fonction qui est activable avec l'instruction CUTMOD ou CUTMODK remplace la fonction activable au moyen de la donnée de réglage SD42984 $SC_CUTDIRMOD. Cependant la disponibilité de cette fonction reste inchangée. Toutefois, l'utilisation parallèle des deux fonctions étant déconseillée, elle ne peut être activée que lorsque CUTMOD est égal à zéro et CUTMODK égal à la chaîne vide.

Informations complémentaires

Lecture des données de correction modifiées

Les données de correction modifiées sont disponibles dans les variables système et variables OPI suivantes :

Signification	Variable système	Variable BTSS
Position du tranchant	$P_AD[2]	cuttEdgeParam2
Angle du porte-outil	$P_AD[10]	cuttEdgeParam10
Sens de coupe	$P_AD[11]	cuttEdgeParam11
Angle de dépouille	$P_AD[24]	cuttEdgeParam24

Les données sont toujours modifiées par rapport aux paramètres d'outil correspondants ($TC_DP2[..., ...], etc.) lorsque la fonction "Modification des données de correction des outils orientables" a été activée avec l'instruction `CUTMOD` ou `CUTMODK` et qu'un outil a été orienté par un organe porte-outil orientable ou une transformation d'orientation.

Autres variables système pertinentes pour la fonction

Variable système	Signification
$P_CUTMOD_ANG / $AC_CUTMOD_ANG	Fournit l'angle de rotation d'un outil dans le plan d'usinage actif, utilisé par les fonctions CUTMOD ou CUTMODK pour la détermination des données de tranchant modifiées.
$P_CUTMOD / $AC_CUTMOD	Lecture de la valeur courante valide, qui a été programmée en dernier avec l'instruction CUTMOD (numéro de l'organe porte-outil pour lequel la modification des données de correction doit être activée). Si la dernière valeur programmée était `CUTMOD = -2` (activation avec l'organe porte-outil orientable actif à l'instant), la valeur de retour de la variable système n'est pas "-2", mais le numéro de l'organe porte-outil orientable qui était actif au moment de la programmation.
$P_CUTMODK / $AC_CUTMODK	Lit le nom de la transformation utilisé pour créer le jeu de paramètres "Modification des données de correction" actuellement valide.

Variable système	Signification	
$P_CUT_INV / $AC_CUT_INV	Livre la valeur TRUE lorsque l'outil est tourné de sorte que le sens de rotation de la broche doit être inversé. A cet effet, les quatre conditions suivantes doivent être remplies dans le bloc auquel se rapporte l'opération de lecture correspondante : 1. Un outil de tournage ou de rectification est actif (types d'outil 400 à 599 et/ou SD42950 $SC_TOOL_LENGTH_TYPE = 2). 2. La modification des données de correction a été activée à l'aide de l'instruction `CUTMOD` ou `CUTMODK`. 3. Un organe porte-outil orientable ou une transformation d'orientation définie à l'aide de chaînes cinématiques est actif/active, sélectionné(e) par l'instruction CUTMOD ou CUTMODK. 4. L'outil est orienté par l'organe porte-outil orientable ou par la transformation d'orientation cinématique de sorte que la normale résultante du tranchant de l'outil est tournée de plus de 90° (normalement 180°) par rapport à la position initiale. Si au moins une des quatre conditions mentionnées n'est pas remplie, la variable contient la valeur FALSE. Pour les outils dont la position du tranchant n'est pas définie, la valeur des variables est toujours FALSE.	
$P_CUTMOD_ERR	État d'erreur après le dernier appel de la fonction CUTMOD La fonction CUTMOD peut également être appelée de manière implicite lors d'un changement d'outil. La variable est réinitialisée à zéro lors d'un Reset. À chaque changement d'outil, elle est d'abord réinitialisée puis, le cas échéant, réécrite. La variable est codée par bits. Les bits ont les significations suivantes :	
	Bit 0 :	Aucun sens de coupe valide n'est défini pour l'outil actif.
	Bit 1 :	Les angles de coupe (angle de dépouille et angle de porte-outil) de l'outil actif ont la valeur zéro.
	Bit 2 :	La valeur de l'angle de dépouille de l'outil actif n'est pas autorisée (< 0° ou > 180°).
	Bit 3 :	La valeur de l'angle de porte-outil de l'outil actif n'est pas autorisée (< 0° ou > 90°).
	Bit 4 :	La valeur de l'angle de la plaquette de l'outil actif n'est pas autorisée (< 0° ou > 90°).
	Bit 5 :	La combinaison position du tranchant - angle du porte-outil de l'outil actif n'est pas autorisée (pour les positions du tranchant 1 à 4, l'angle du porte-outil doit être ≤ 90°, pour les positions du tranchant 5 bis 8 ≥ 90°).
	Bit 6 :	Rotation non autorisée de l'outil actif. L'outil a été tourné de ± 90° (avec une tolérance d'environ 1°) par rapport au plan d'usinage actif. Par conséquent, la position du tranchant dans le plan d'usinage n'est plus définie.
	Bit 7 :	La plaquette ne se trouve pas dans le plan d'usinage et l'angle entre plaquette et plan d'usinage dépasse la limite supérieure spécifiée dans la donnée de réglage SD42998 $SC_CUTMOD_PLANE_TOL.
	Bit 8 :	La plaquette ne se trouve pas dans le plan d'usinage. La valeur absolue de l'angle α est supérieure à 1°. L'angle α est l'angle de rotation autour de l'axe de coordonnée perpendiculaire à la fois à l'axe de rotation de l'angle β et à l'axe de rotation de l'angle γ (pour G18, c'est l'axe X).

$P_... : Variables de prétraitement

$AC_... : Variables d'exécution

Toutes les variables d'exécution peuvent être lues dans des actions synchrones. Un accès en lecture depuis le prétraitement entraîne un arrêt du prétraitement.

Changement de plan

Pour la détermination de la position modifiée du tranchant, du sens de coupe et de l'angle du porte-outil ou de l'angle de dépouille, le tranchant doit être considéré dans le plan actif correspondant (G17 - G19).

Si la donnée de réglage SD42940 $SC_TOOL_LENGTH_CONST (modification des composantes de longueur d'outil en cas de changement de plan) contient cependant une valeur valide différente de zéro (plus ou moins 17, 18 ou 19), son contenu détermine le plan dans lequel les grandeurs déterminantes sont considérées.

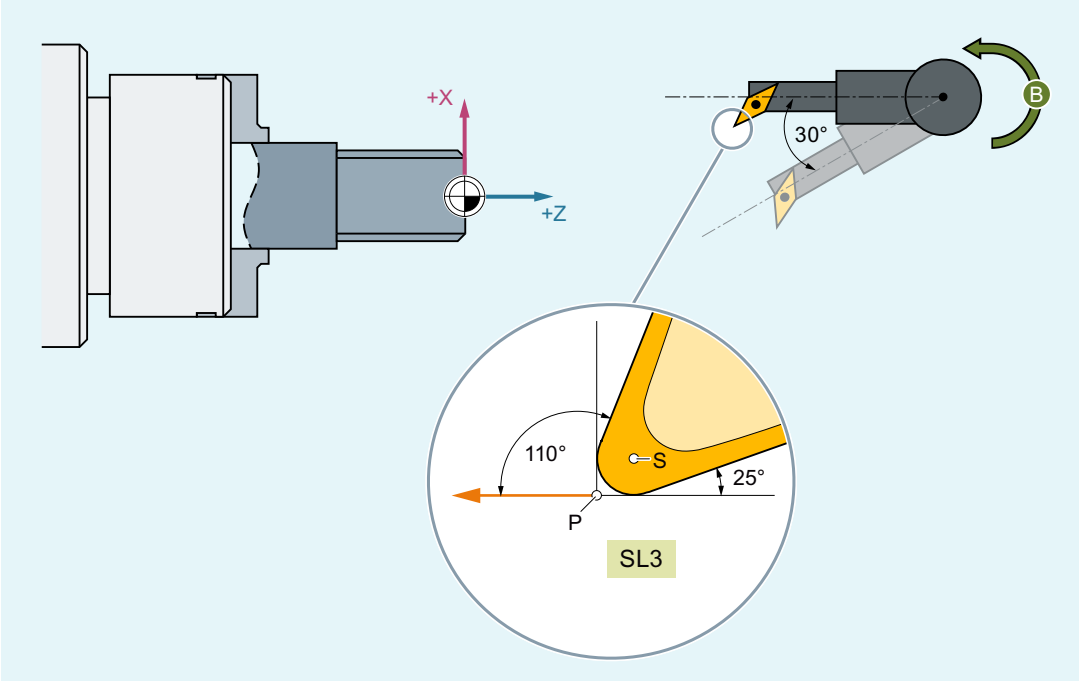
Cette règle de priorité des données de réglage par rapport au code G peut être invalidée en activant le bit 18 du paramètre machine $MC_TOOL_PARAMETER_DEF_MASK. Autrement dit, lorsque ce bit est à 1, c'est toujours le plan défini avec l'instruction G du groupe 6 qui est valide.

Prise d'effet des données de tranchant modifiées

La position modifiée du tranchant et le point de référence modifié du tranchant prennent effet immédiatement même s'ils sont programmés pour un outil déjà actif. A cet effet, il n'est pas nécessaire de réactiver l'outil.

Exemple

Pour un outil avec la position de tranchant 3 et un organe porte-outil orientable pouvant tourner l'outil autour de l'axe B, la position du tranchant doit être modifiée après une rotation d'outil à l'aide de l'instruction CUTMOD.



S : Centre du tranchant

P : Point de référence du tranchant

SL : Position du tranchant

Code de programme	Commentaire
N10 $TC_DP1[1,1]=500	
N20 $TC_DP2[1,1]=3	; Position du tranchant
N30 $TC_DP3[1,1]=12	
N40 $TC_DP4[1,1]=1	
N50 $TC_DP6[1,1]=6	
N60 $TC_DP10[1,1]=110	; Angle du porte-outil
N70 $TC_DP11[1,1]=3	; Sens de coupe
N80 $TC_DP24[1,1]=25	; Angle de dépouille
N90 $TC_CARR7[2]=0 $TC_CARR8[2]=1 $TC_CARR9[2]=0	; Axe B
N100 $TC_CARR10[2]=0 $TC_CARR11[2]=0 $TC_CARR12[2]=1	; Axe C
N110 $TC_CARR13[2]=0	
N120 $TC_CARR14[2]=0	
N130 $TC_CARR21[2]=X	
N140 $TC_CARR22[2]=X	
N150 $TC_CARR23[2]="M"	

Code de programme	Commentaire
N160 TCOABS CUTMOD=0	
N170 G18 T1 D1 TCARR=2	; X Y Z
N180 X0 Y0 Z0 F10000	; 12.000 0.000 1.000
N190 $TC_CARR13[2]=30	
N200 TCARR=2	
N210 X0 Y0 Z0	; 10.892 0.000 -5.134
N220 G42 Z-10	; 8.696 0.000 -17.330
N230 Z-20	; 8.696 0.000 -21.330
N240 X10	; 12.696 0.000 -21.330
N250 G40 X20 Z0	; 30.892 0.000 -5.134
N260 CUTMOD=2 X0 Y0 Z0	; 8.696 0.000 -7.330
N270 G42 Z-10	; 8.696 0.000 -17.330
N280 Z-20	; 8.696 0.000 -21.330
N290 X10	; 12.696 0.000 -21.330
N300 G40 X20 Z0	; 28.696 0.000 -7.330
N310 M30	

Les valeurs numériques des commentaires indiquent les positions de fin de bloc en coordonnées machine (SCM), dans l'ordre X → Y → Z.

Explications

Dans le bloc N180, l'outil est d'abord activé avec CUTMOD=0 et organe porte-outil orientable non pivoté. Tous les vecteurs d'offset de l'organe porte-outil étant 0, la position accostée correspond aux longueurs d'outil indiquées dans $TC_DP3[1,1] et $TC_DP4[1,1].

Dans le bloc N200, l'organe porte-outil orientable est activé avec une rotation de 30° autour de l'axe B. La position du tranchant n'étant pas modifiée compte tenu de CUTMOD=0, l'ancien point de référence du tranchant reste déterminant. C'est pourquoi la position qui garde l'ancien point de référence du tranchant à l'origine (ce qui signifie que le vecteur (1, 12) tourne de 30° dans le plan Z/X) est accostée dans le bloc N210.

Contrairement au bloc N200, CUTMOD=2 est actif dans le bloc N260. En raison de la rotation de l'organe porte-outil orientable, la position modifiée du tranchant est 8. Il en résulte également des positions d'axe différentes.

Dans les blocs N220 et N270, la correction du rayon d'outil (CRO) est activée. La position du tranchant, qui diffère dans les deux sections de programme, n'influence pas les positions finales des blocs dans lesquels la correction du rayon d'outil (CRO) est active. Les positions correspondantes sont donc identiques. Ce n'est que dans les blocs de désactivation N260 et N300 que les différentes positions du tranchant ont de nouveau un effet.

3.13.13 Utilisation des environnements d'outils

Vue d'ensemble des fonctions

- Enregistrement de l'environnement d'outil (TOOLENV) (Page 832)
- Suppression de l'environnement d'outil (DELTOOLENV) (Page 835)
- Lecture des numéros T, D et DL (GETTENV) (Page 836)
- Lecture de longueurs d'outil ou de composantes de longueur d'outil (GETTCOR) (Page 837)
- Modification des composantes d'outil (SETTCOR) (Page 844)

Vue d'ensemble des variables système

- Lecture des informations relatives aux environnements d'outil archivés ($P_TOOLENVN, ($P_TOOLENV) (Page 837)

3.13.13.1 Enregistrement de l'environnement d'outil (TOOLENV)

La fonction TOOLENV sert à enregistrer tous les états actuels significatifs pour l'exploitation des données d'outil stockées dans la mémoire.

Dans le détail, il s'agit des données suivantes :

- L'instruction G active du groupe :
  - 6 (G17, G18, G19)
  - 56 (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)
- L'axe transversal actif :
- Les paramètres machine :
  - PM18112 $MN_MM_KIND_OF_SUMCORR (caractéristiques des corrections totales dans la zone TO)
  - PM20360 $MC_TOOL_PARAMETER_DEF_MASK (définition des paramètres d'outil)

- Les données de réglage :
  - SD42900 $SC_MIRROR_TOOL_LENGTH (inversion du signe de la longueur d'outil pour la fonction miroir)
  - SD42910 $SC_MIRROR_TOOL_WEAR (inversion du signe de l'usure d'outil pour la fonction miroir)
  - SD42920 $SC_WEAR_SIGN_CUTPOS (signe de l'usure pour les outils avec dispositifs de coupe)
  - SD42930 $SC_WEAR_SIGN (signe de l'usure)
  - SD42935 $SC_WEAR_TRANSFORM (transformations pour composantes d'outil)
  - SD42940 $SC_LENGTH_CONST (changement des composantes de longueur d'outil lors d'un changement de plan)
  - SD42942 $SC_TOOL_LENGTH_CONST_T (Changement des composantes de longueur d'outil pour outils de tournage lors d'un changement de plan)
  - SD42950 $SC_TOOL_LENGTH_TYPE (affectation des composantes de longueur d'outil indépendamment du type d'outil)
  - SD42954 $SC_TOOL_ORI_CONST_M (Changement des composantes d'orientation de l'outil pour outils de fraisage lors d'un changement de plan)
  - SD42956 $SC_TOOL_ORI_CONST_T (Changement des composantes d'orientation de l'outil pour outils de tournage lors d'un changement de plan)
- La composante d'orientation du frame global actuel (rotations et fonctions miroir, pas de décalages d'origine ni d'homothéties)
- La composante d'orientation et la longueur résultante de l'organe porte-outil orientable actif
- La composante d'orientation et la longueur résultante d'une transformation active

Outre les données citées qui décrivent l'environnement de l'outil, les numéros T, D et DL de l'outil actif sont également enregistrés afin de pouvoir accéder ultérieurement à cet outil dans le même environnement comme lors de l'appel de TOOLENV, sans qu'il soit nécessaire de désigner à nouveau l'outil.

Syntaxe

```
<Status> = TOOLENV(<Name>)
```

Signification

`TOOLENV(...)` :	Fonction prédéfinie pour l'enregistrement d'un environnement d'outil	
	Seule dans le bloc :	oui

<Status> :	Valeur de retour de la fonction. Les valeurs négatives indiquent des états d'erreur.		
	Type de données :		INT
	Valeur :	0	Fonction OK
		-1	Pas d'espace mémoire réservé pour les environnements d'outil : PM18116 $MN_MM_NUM_TOOL_ENV = 0 C'est-à-dire que la fonctionnalité "Environnements d'outil" n'est pas disponible.
		-2	Plus aucun espace mémoire libre disponible pour les environnements d'outil.
		-3	Une chaîne vide en tant que nom d'environnement d'outil est non admissible.
		-4	Aucun paramètre (<Name>) spécifié.
Paramètres			
1 <Name> :	Nom sous lequel le jeu de paramètres actuel doit être enregistré Si un jeu de paramètres du même nom existe déjà, celui-ci est écrasé. Dans ce cas, le statut est "0".		
	Type de données :		STRING

Informations complémentaires

Cote de base / cote d'adaptation - Correction de longueur d'outil

Si la gestion de magasin d'outils est active (seulement disponible avec l'option "Gestion des outils"), c'est la valeur du paramètre machine qui décide si oui ou non la longueur de l'adaptateur ou la cote de base (paramètre spécifique au tranchant $TC_DP21, $TC_DP22 et $TC_DP23) doit entrer dans le calcul de la longueur d'outil :

PM18104 $MN_MM_NUM_TOOL_ADAPTER (adaptateur d'outils dans la zone TO)

Compte tenu qu'une modification de ce paramètre machine peut seulement devenir effective au démarrage de la commande, il n'est pas enregistré dans l'environnement d'outil.

Longueur résultante des porte-outils orientables et des transformations

Remarque

Tant pour les porte-outils orientables que pour les transformations, il existe des variables système ou des paramètres machine qui agissent comme des composantes supplémentaires de longueur d'outil et qui sont assujettis en tout ou partie aux rotations exécutables par celles-ci. Les composantes supplémentaires de longueur d'outil qui en résultent doivent être enregistrées lors de l'appel de TOOLENV compte tenu qu'elles constituent une partie de l'environnement dans lequel l'outil est utilisé.

Transformation d'adaptateur

La transformation d'adaptateur est une caractéristique de l'adaptateur d'outil et de ce fait de l'outil complet. C'est pourquoi elle ne fait pas partie intégrante d'un environnement d'outil qui peut être appliqué à un autre outil.

L'enregistrement de l'ensemble des données requises pour la détermination de la longueur totale de l'outil permet de calculer à un moment ultérieur la longueur effective d'un outil, même si celui-ci n'est plus actif à ce moment-là ou si les conditions ambiantes (par ex. instructions G ou données de réglage) ont changé. De même, la longueur effective d'un autre outil peut être calculée en supposant que celui-ci a été utilisé dans les mêmes conditions que l'outil pour lequel l'état a été enregistré.

Nombre maximal de jeux de paramètres pour les environnements d'outil

Le paramètre machine PM18116 $MN_MM_NUM_TOOL_ENV permet de déterminer le nombre maximal de jeux de paramètres pouvant être stockés pour la description d'environnements d'outil. Les données se trouvent dans la zone TOA. Elles sont conservées même après la mise hors tension de la commande.

La sauvegarde des données est impossible. En d'autres termes, ces données ne peuvent pas être transmises entre différentes commandes.

3.13.13.2 Suppression de l'environnement d'outil (DELTOOLENV)

La fonction DELTOOLENV permet de supprimer des jeux de paramètres pour la description d'environnements d'outil La suppression signifie que le jeu de paramètres enregistré sous un nom défini n'est plus accessible (toute tentative d'accès génère une alarme).

Remarque

Les jeux de paramètres peuvent seulement être supprimés avec la fonction DELTOOLENV, via un download INITIAL.INI ou un démarrage à froid (démarrage de la CN avec paramètres machine standard). Il n'existe aucune autre procédure de suppression automatique.

Syntaxe

```
<Status> = DELTOOLENV(<Name>)
<Status> = DELTOOLENV()
```

Signification

`DELTOOLENV(...)` :	Fonction prédéfinie pour la suppression d'un environnement d'outil		
	Seule dans le bloc :		oui
`<Status>` :	Valeur de retour de la fonction. Les valeurs négatives indiquent des états d'erreur.		
	Type de données :		INT
	Valeur :	0	Fonction OK
		-1	Pas d'espace mémoire réservé pour les environnements d'outil : PM18116 $MN_MM_NUM_TOOL_ENV = 0 C'est-à-dire que la fonctionnalité "Environnements d'outil" n'est pas disponible.
		-2	Il n'existe pas d'environnement d'outil avec le nom spécifié.

Paramètres			
1	`<Name>` :	Nom du jeu de paramètres à supprimer	
		Type de données :	STRING
`DELTOOLENV()` :		`DELTOOLENV()` sans spécification de nom supprime tous les jeux de paramètres relatifs à la description des environnements d'outil	

3.13.13.3 Lecture des numéros T, D et DL (GETTENV)

La fonction GETTENV sert à lire les numéros T, D et DL enregistrés dans un environnement d'outil

Syntaxe

```
<Status> = GETTENV(<Name>, <TDDL>)
```

Signification

`GETTENV(...)` :	Fonction prédéfinie pour la lecture de numéros T, D et DL dans un jeu de paramètres relatif à la description d'un environnement d'outil		
	Seule dans le bloc :		oui
`<Status>` :	Valeur de retour de la fonction. Les valeurs négatives indiquent des états d'erreur.		
	Type de données :		INT
	Valeur :	0	Fonction OK
		-1	Pas d'espace mémoire réservé pour les environnements d'outil : PM18116 $MN_MM_NUM_TOOL_ENV = 0 C'est-à-dire que la fonctionnalité "Environnements d'outil" n'est pas disponible.
		-2	Il n'existe pas d'environnement d'outil avec le nom spécifié.

Paramètres			
1	`<Name>` :	Nom du jeu de paramètres à partir duquel doivent être lus les numéros T, D et DL	
		Type de données :	STRING
2	`<TDDL>` :	Le champ de ce paramètre de résultat contient les numéros T, D et DL de l'outil dont l'environnement est enregistré dans le jeu de paramètres spécifié : • <TDDL> [0] : Numéro T • <TDDL> [1] : Numéro D • <TDDL> [2] : numéro DL	
		Type de données :	INT[3]
`GETTENV(,<TDDL>)`, `GETTENV("",<TDDL>)` :		Lors de l'appel de la fonction GETTENV, l'omission du premier paramètre ou le transfert de la chaîne vide en tant que premier paramètre est autorisé(e). Dans ces deux cas particuliers, les numéros T, D et DL de l'outil **actif** sont retournés dans <TDDL>.	

3.13.13.4 Lecture des informations relatives aux environnements d'outil archivés ($P_TOOLENVN, ($P_TOOLENV)

Les informations relatives aux environnements d'outil archivés peuvent être lues dans les variables système suivantes :

`$P_TOOLENVN` :	Fournit le nombre de jeux de paramètres définis (et pas encore supprimés) à l'aide de TOOLENV pour la description d'environnements d'outil			
	Syntaxe :	`<n> = $P_TOOLENVN`		
	Signification :	`<n>` :	Nombre de jeux de paramètres définis.	
			Type de données :	INT
			Plage de valeurs :	0 ... PM18116 $MN_MM_NUM_TOOL_ENV
	L'accès à cette variable système est également autorisé lorsqu'aucun environnement d'outil n'est possible (PM18116 = 0). Dans ce cas, la valeur de retour est "0".			
`$P_TOOLENV` :	Fournit le nom du <i>-ième jeu de paramètres pour la description d'un environnement d'outil			
	Syntaxe :	`<Name> = $P_TOOLENV[<i>]`		
	Signification :	`<Name>` :	Nom du jeu de paramètres avec le numéro <i>	
			Type de données :	STRING
		`<i>` :	Numéro du jeu de paramètres	
			Type de données :	INT
			Plage de valeurs :	1 ... $P_TOOLENVN
	L'affectation des numéros aux jeux de paramètres n'est pas fixe mais peut au contraire être modifiée en supprimant ou en créant de nouveaux jeux de paramètres. La numérotation des jeux de paramètres s'effectue en interne. Si <i> renvoie à un jeu de paramètres non défini, la chaîne vide est retournée. Si l'indice <i> est non valide, c'est-à-dire que si <i> est inférieur à 1 ou supérieur au nombre maximal de jeux de paramètres pour les environnements d'outil (PM18116 $MN_MM_NUM_TOOLENV), l'alarme suivante est émise : Alarme 17020 "Indice1 de tableau non autorisé"			

3.13.13.5 Lecture de longueurs d'outil ou de composantes de longueur d'outil (GETTCOR)

La fonction GETTCOR sert à lire les longueurs d'outil ou les composantes de longueur d'outil.

Le paramétrage permet de spécifier les composantes à prendre en compte et les conditions d'utilisation dans lesquelles l'outil doit être pris en considération.

Syntaxe

```
<Status> = GETTCOR(<Len>[, <Comp>, <Stat>, <T>, <D>, <DL>])
```

Signification

`GETTCOR(...)` :	Fonction prédéfinie pour la lecture de longueurs d'outil ou de composantes de longueur d'outil		
	Seule dans le bloc :		oui
`<Status>` :	Valeur de retour de la fonction. Les valeurs négatives indiquent des états d'erreur.		
	Type de données :		INT
	Valeur :	0	Fonction OK
		-1	Pas d'espace mémoire réservé pour les environnements d'outil : PM18116 $MN_MM_NUM_TOOL_ENV = 0 C'est-à-dire que la fonctionnalité "Environnements d'outil" n'est pas disponible.
		-2	Il n'existe pas d'environnement d'outil avec le nom indiqué sous <Stat>.
		-3	Chaîne non valide dans le paramètre <Comp>. Les causes de cette erreur peuvent être des caractères non valides ou programmés en double.
		-4	Numéro T non valide
		-5	Numéro D non valide
		-6	Numéro DL non valide
		-7	Tentative d'accès à un module mémoire non disponible.
		-8	Tentative d'accès à une option non disponible (par ex. orientation d'outil programmable, gestion d'outil).
		-9	La chaîne <Comp> contient un deux-points (identificateur pour les spécifications d'un système de coordonnées), toutefois aucun caractère valide ne suit pour la désignation du système de coordonnées requis.
Paramètres			

1	<Len> :	Vecteur résultat	
		Type de données :	REAL[11]
		Les composantes du vecteur sont classées dans l'ordre suivant : • <Len> [0] : Type d'outil • <Len> [1] : Position du tranchant • <Len> [2] : Abscisse • <Len> [3] : Ordonnée • <Len> [4] : Cote • <Len> [5] : Rayon de l'outil Le système de coordonnées défini dans <Comp> et <Stat> est considéré comme système de coordonnées de référence pour les composantes de longueur. Si aucun système de coordonnées n'est défini dans <Comp>, les longueurs d'outil sont représentées dans le système de coordonnées machine. L'affectation de l'abscisse, de l'ordonnée et de la cote relatives aux axes géométriques dépend du niveau actif dans l'environnement d'outil utilisé. C'est-à-dire que pour G17 l'abscisse est parallèle à X, pour G18 elle est parallèle à Z, etc. Les composantes <Len>[6] à <Len>[10] contiennent les paramètres supplémentaires pouvant être spécifiés pour la description géométrique d'un outil (par ex. $TC_DP7 à $TC_DP11 pour la géométrie ou les composantes correspondantes pour usure ou corrections de somme et de paramétrage). Ces 5 éléments supplémentaires et le rayon d'outil sont seulement définis pour les composantes E, G, S et W. Leur évaluation ne dépend pas de <Stat>. C'est pourquoi les valeurs correspondantes dans <Len>[6] à <Len>[10] ne peuvent être différentes de zéro que si au moins une des quatre composantes citées entre dans le calcul de la longueur d'outil. Les autres composantes n'ont aucune incidence sur le résultat. Les indications de cotes se rapportent au système de base de la commande (anglo-saxon ou métrique).	

2	<Comp> :	Composantes de longueur d'outil (facultatif)			
		Type de données :	STRING		
		La chaîne de caractères est constituée de deux chaînes partielles séparées l'une de l'autre par un deux-points. Forme gén. : "<SubStr_1> [: <SubStr_2]"			
		<SubStr_1> :	La **première chaîne partielle** désigne les composantes de longueur d'outil qui doivent être prises en compte lors du calcul de longueur d'outil. L'ordre des caractères dans la chaîne partielle ainsi que leur casse (majuscules ou minuscules) est quelconque. Autant d'espaces ou tabulations (white spaces) que nécessaire peuvent être insérés entre les caractères. **Remarque :** Les caractères de la chaîne partielle **ne doivent pas** être programmés en double !		
			Symbole :	–	Signe moins (seulement admissible en tant que premier caractère !) La longueur totale d'outil est calculée, après soustraction des composantes spécifiées dans la chaîne suivante.
				C	Cote d'adaptation ou cote de base (celle des deux variantes de composantes qui est active pour l'outil utilisé)
				E	Corrections de réglage
				G	Géométrie
				K	Transformation cinématique (seulement exploitée en cas de transformation générique à 3, 4 et 5 axes !)
				S	Corrections totales
				T	Porte-outil orientable
				W	Usure
			Si la première chaîne partielle (à l'exception des espaces) est vide, cela signifie que la longueur totale d'outil doit être calculée en tenant compte de toutes les composantes. Il en va de même lorsque le paramètre <Comp> n'est pas spécifié.		
		<Substr_2> :	La **deuxième chaîne partielle** facultative désigne le système de coordonnées dans lequel la longueur d'outil doit être délivrée. La deuxième chaîne partielle est constituée d'un seul et unique caractère pertinent.		
			Symbole :	A	Système de coordonnées réglable (SCR)
				B	Système de coordonnées de base (SCB)
				K	Système de coordonnées de l'outil de la transformation cinématique (SCT)
				M	Système de coordonnées machine (SCM)
				T	Système de coordonnées de l'outil (SCO)
				W	Système de coordonnées pièce (SCP)
			Si aucun système de coordonnées n'est spécifié, l'évaluation a lieu dans le système de coordonnées machine (SCM). Les rota-		

			tions à prendre en compte le cas échéant sont déterminées par l'environnement d'outil défini dans <Stat>.
3	<_Stat> :	Nom du jeu de paramètres pour la description d'un environnement d'outil (facultatif)	
		Type de données :	STRING
		Si la valeur de ce paramètre est la chaîne vide ("") ou si elle n'est pas spécifiée, l'état actuel est utilisé. Si aucun outil n'est spécifié, l'outil actuel est utilisé.	
4	<T> :	Numéro T interne de l'outil (facultatif)	
		Type de données :	INT
		Si ce paramètre n'est pas spécifié ou si sa valeur est "0", l'outil mémorisé dans <Stat> est utilisé. Si la valeur de ce paramètre est "-1", le numéro T de l'outil actif est utilisé. La spécification explicite du numéro de l'outil actif est également admise. **Remarque :** Si <Stat> n'est pas spécifié, l'état actuel est utilisé comme environnement d'outil. Compte tenu qu'avec <T> = 0, le renvoi a lieu au numéro T enregistré dans l'environnement d'outil, c'est l'outil actif qui est utilisé dans celui-ci, c'est-à-dire que les indications <T> = 0 et <T> = -1 ont la même signification dans ce cas particulier.	
5	<D> :	Numéro de tranchant de l'outil (facultatif)	
		Type de données :	INT
		Si ce paramètre n'est pas spécifié ou si sa valeur est "0", le numéro D utilisé dépend de la source du numéro T. Si le numéro T est utilisé à partir de l'environnement d'outil, le numéro D de l'environnement d'outil est lu également, autrement c'est celui de l'outil actuellement actif.	
6	<DL> :	Numéro de la correction en fonction du lieu (facultatif)	
		Type de données :	INT
		Si ce paramètre n'est pas spécifié, le numéro DL utilisé dépend de la source du numéro T. Si le numéro T est utilisé à partir de l'environnement d'outil, le numéro D de l'environnement d'outil est lu également, autrement c'est celui de l'outil actuellement actif.	

Exemples

`GETTCOR(_LEN)`	Calcule la longueur de l'outil actuellement actif dans le système de coordonnées machine en tenant compte de toutes les composantes.
`GETTCOR(_LEN,"CGW:W")`	La longueur d'outil constituée de la cote d'adaptation ou de base, de la géométrie et de l'usure est calculée pour l'outil actif. D'autres constituantes telles que par ex. le porte-outil orientable ou les transformations cinématiques ne sont pas prises en compte. La valeur est toujours donnée dans le système de coordonnées pièce.

`GETTCOR (_LEN,"-K:B")`	La longueur totale de l'outil actif sans prise en compte des composantes de longueur d'une transformation cinématique éventuellement active est calculée. La valeur est donnée dans le système de coordonnées de base.
`GETTCOR (_LEN,":M","Testenv1",,3)`	La longueur totale de l'outil archivé pour l'environnement d'outil sous le nom "Testenv1" est calculée dans le système de coordonnées machine. Le calcul s'effectue toutefois indépendamment du numéro de tranchant mémorisé pour le numéro de tranchant D3.

plus d'informations

Transformation d'adaptateur / porte-outil orientable / transformation cinématique

Les rotations et inversions de composantes qui ont été réalisées le cas échéant par la transformation d'adaptateur, le porte-outil orientable et une transformation cinématique, font partie intégrante de l'environnement d'outil. C'est pourquoi elles sont toujours effectuées, même si la composante de longueur correspondante ne doit pas être prise en compte. Si celles-ci ne sont pas souhaitées, des environnements d'outil pour lesquels les transformations correspondantes ne sont pas actives, doivent être définis. Dans de nombreux cas (notamment lorsqu'aucune transformation ni porte-outil orientable n'est utilisé sur une machine), les jeux de paramètres archivés pour les environnements d'outil remplissent ces conditions automatiquement. Ainsi l'utilisateur n'a pas besoin de les prendre spécialement en compte.

Outils de tournage et de rectification : Calcul de la longueur d'outil en fonction de PM20360 $MC_TOOL_PARAMETER_DEF_MASK

La façon dont l'usure ou la longueur d'outil sur les outils de tournage et de rectification doit être évaluée dans un axe diamétral éventuellement présent est déterminée dans le paramètre machine :

PM20360 $MC_TOOL_PARAMETER_DEF_MASK (définition des paramètres d'outil)

Bit	Valeur	
0	Sur les outils de tournage et de rectification, le **paramètre d'usure** de l'axe transversal est pris en compte en tant que valeur de diamètre :	
	= 0 (par défaut)	non
	= 1	oui
1	Sur les outils de tournage et de rectification, la **composante de longueur d'outil** de l'axe transversal est prise en compte en tant que valeur de diamètre :	
	= 0 (par défaut)	non
	= 1	oui

Si les bits concernés sont mis à 1, l'entrée associée est évaluée avec le facteur 0,5. Cette évaluation influence également le calcul de la longueur d'outil fournie par GETTCOR.

Exemple :

PM20360 $MC_TOOL_PARAMETER_DEF_MASK = 3

PM20100 $MC_DIAMETER_AX_DEF (axe géométrique avec fonction d'axe transversal) = "X"

X est l'axe diamétral (configuration de tour standard)

```
Code de programme                          Commentaire
N30 $TC_DP1[1,1]=500
N40 $TC_DP2[1,1]=2
N50 $TC_DP3[1,1]=3,0                       ; Géométrie L1
N60 $TC_DP4[1,1]=4,0
N70 $TC_DP5[1,1]=5,0
N80 $TC_DP12[1,1]=12,0                     ; Usure L1
N90 $TC_DP13[1,1]=13,0
N100 $TC_DP14[1,1]=14,0
N110 T1 D1 G18
N120 R1=GETTCOR(_LEN,"GW")
N130 R3=_LEN[2]                            ; 17,0 (= 4,0 + 13,0)
N140 R4=_LEN[3]                            ; 7,5 (= 0,5 * 3,0 + 0,5 * 12,0)
N150 R5=_LEN[4]                            ; 19,0 (= 5,0 + 14,0)
N160 M30
```

Composantes de longueur de la transformation cinématique et du porte-outil orientable

Si un **porte-outil orientable** est pris en compte pour le calcul de la longueur d'outil, les vecteurs suivants entrent dans le calcul de la longueur d'outil :

Type	Vecteurs
M	l1 et l2
T	l1, l2 et l3
P	La longueur d'outil n'est pas influencée par le porte-outil orientable.

Dans le cas de la **transformation 5 axes** générique, les paramètres machine suivants entrent dans le calcul de la longueur d'outil pour les types de transformation 24 et 56 :

Type de transformation	Paramètres machine
24	PM24550/24650 $MC_TRAFO5_BASE_TOOL_1/2 PM24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 PM24558/24658 $MC_TRAFO5_PART_OFFSET_1/2
56	PM24550/24650 $MC_TRAFO5_BASE_TOOL_1/2 PM24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2

Le type de transformation 56 (outil en mouvement et pièce en mouvement) correspond au type M pour le porte-outil orientable.

Pour cette transformation 5 axes, le vecteur PM24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 (vecteur du décalage cinématique de la 1e/2e transformation 5 axes dans le canal) correspond, dans les versions logicielles antérieures, à la somme des deux vecteurs l_1 et l_3 pour le porte-outil orientable de type M.

Dans les deux cas, seules la somme est pertinente pour la transformation. Le type de composition issu des deux composantes individuelles est sans importance. Lors du calcul de la longueur d'outil il importe toutefois de connaître la part à affecter à l'outil et la part à affecter à

la table de travail. C'est pourquoi le paramètre machine PM24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 (vecteur du décalage cinématique dans la table) a été introduit. Il correspond au vecteur l3. Le paramètre machine PM24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 ne correspond alors plus à la somme de l1 et l3, mais seulement au vecteur l1. Si le paramètre machine PM24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 est égal à zéro, le comportement est identique au comportement précédent.

Compatibilité

Avec les fonctions TOOLENV et SETTCOR entre autres, la fonction GETTCOR sert à remplacer des parties de la fonctionnalité qui étaient réalisées jusqu'ici de manière externe dans les cycles de mesure.

Dans ces derniers, seule une partie des paramètres déterminant en fin de compte la longueur effective de l'outil a été analysée. Les fonctions citées peuvent être paramétrées de manière que le comportement des cycles de mesure concernant le calcul de la longueur d'outil soit reproductible.

3.13.13.6 Modification des composantes d'outil (SETTCOR)

La fonction SETTCOR sert à modifier les composantes d'outil en tenant compte de toutes les autres conditions influant sur l'évaluation des différentes composantes.

Remarque

Concernant la terminologie : Dans les chapitres suivants, lorsqu'il est question de composantes d'outil en rapport avec la longueur d'outil, il s'agit des composantes prises en compte sous la forme vectorielle, qui composent la longueur totale de l'outil (par ex. géométrie ou usure). C'est pourquoi untel telle composante est constituée de trois valeurs individuelles (L1, L2, L3) désignées ci-après comme valeurs de coordonnées.

La composante d'outil "Géométrie" est constituée par ex. par conséquent de trois valeurs de coordonnées $TC_DP3 à $TC_DP5.

Syntaxe

```
<Status> = SETTCOR(<CorVal>, <Comp>, [<CorComp>, <CorMode>, <GeoAx>,
<Stat>, <T>, <D>, <DL>])
```

Signification

`SETTCOR(...)` :	Fonction prédéfinie pour la modification de composantes d'outil	
	Seule dans le bloc :	oui

<Status> :	Valeur de retour de la fonction. Les valeurs négatives indiquent des états d'erreur.		
	Type de données :		INT
	Valeur :	0	Fonction OK
		-1	Pas d'espace mémoire réservé pour les environnements d'outil : PM18116 $MN_MM_NUM_TOOL_ENV = 0 C'est-à-dire que la fonctionnalité "Environnements d'outil" n'est pas disponible.
		-2	Il n'existe pas d'environnement d'outil avec le nom indiqué sous <Stat>.
		-3	Chaîne non valide dans le paramètre <Comp>. Les causes de cette erreur peuvent être des caractères non valides ou programmés en double.
		-4	Numéro T non valide.
		-5	Numéro D non valide.
		-6	Numéro DL non valide.
		-7	Tentative d'accès à un module mémoire non disponible.
		-8	Tentative d'accès à une option non disponible (par ex. orientation d'outil programmable, gestion d'outil).
		-9	Valeur chiffrée non admissible pour le paramètre <CorComp**>**.
		-10	Valeur chiffrée non admissible pour le paramètre <CorMode>.
		-11	Les contenus des paramètres <Comp> et <CorComp> sont contradictoires.
		-12	Les contenus des paramètres <Comp> et <CorMode> sont contradictoires.
		-13	Le contenu du paramètre <GeoAx> ne désigne aucun axe géométrique.
		-14	Tentative d'écriture sur une correction de réglage non disponible.
Paramètres			
1	<CorVal> :	Vecteur de correction L'affectation suivante est valable dans le système de coordonnées pièce (SCP) défini par <Stat> : • <CorVal> [0] : abscisse • <CorVal> [1] : ordonnée • <CorVal> [2] : cote Si une seule composante d'outil doit être corrigée (c'est-à-dire aucune correction vectorielle, voir paramètre <CorMode>), la valeur de correction se trouve toujours dans <CorVal>[0], quel que soit l'axe sur lequel elle agit. Le contenu des deux autres composantes n'est alors pas évalué. Si <CorVal> ou une composante de <CorVal> se rapporte à l'axe transversal, l'indication est évaluée en tant que **rayon**. Cela signifie qu'un outil par ex. "s'allonge" de la cote indiquée, ce qui entraîne par conséquent une modification deux fois plus grande du diamètre de la pièce. Les indications de cotes se rapportent au **système de base** (anglo-saxon ou métrique) de la commande.	
		Type de données :	REAL[3]

2	<Comp> :	Composante(s) d'outil			
		Type de données :	STRING		
		La chaîne de caractères est constituée de deux chaînes partielles séparées l'une de l'autre par un deux-points. Forme gén. : "<SubStr_1> [: <SubStr_2]"			
		<SubStr_1> :	La **première chaîne partielle** doit toujours être disponible et peut être constituée d'un ou de deux caractères. À cet effet, le premier ou l'unique caractère est présent pour la 1re composante (Val_1) et le deuxième caractère pour la 2e composante (Val_2), qui sont éditées en fonction des paramètres <CorComp> et <CorMode> suivants.		
			Symbole :	C	Cote d'adaptation ou cote de base (celle des deux variantes de composantes qui est active pour l'outil utilisé)
				E	Corrections de réglage
				G	Géométrie
				S	Corrections totales
				W	Usure
		<Substr_2> :	La **deuxième chaîne partielle** est facultative. Elle peut être constituée alternativement de la lettre (unique) "W" ou "T".		
			Symbole :	W	Si la deuxième chaîne partielle est vide ou contient la lettre "W", les valeurs de correction sont calculées comme si elles étaient mesurées dans le système de coordonnées **pièce** (SCP).
				T	Si la deuxième chaîne partielle contient la lettre "T", les valeurs de correction sont calculées comme si elles étaient mesurées dans le systèmes de coordonnées **outil** (**T**ool Coordinate System, TCS).
		La casse des caractères dans la chaîne (majuscules ou minuscules) est sans importance. Autant d'espaces ou tabulations (white spaces) que nécessaire peuvent être insérés entre les caractères.			

3	<CorComp> :	Spécifie la ou les composant(e)s du jeu de paramètres d'outil qui doivent être décrites (facultatif)		
		Type de données :		INT
		Valeur :	0	La valeur de correction <CorVal>[0] se rapporte à l'axe géométrique transmis dans le paramètre <GeoAx> dans le système de coordonnées pièce ou dans le système de coordonnées outil (voir à ce sujet la description du paramètre <Comp>). C'est-à-dire que la valeur de correction doit entrer dans le calcul des composantes d'outil désignées de manière que, en tenant compte de tous les paramètres qui peuvent avoir une influence sur le calcul de la longueur d'outil, il en résulte une modification de la longueur totale de l'outil dans le sens de l'axe donné et avec la valeur prescrite. Cette modification doit être obtenue en corrigeant la composante indiquée sous <Comp> et la règle opératoire symbolique spécifiée sous <CorMode> (voir paramètre suivant). C'est pourquoi la correction qui en résulte peut avoir un effet sur les trois composantes d'axe.
			1	Comme "0" mais vectorielle. Le contenu du vecteur <CorVal> se rapporte aux abscisses, aux ordonnées et aux cotes dans le système de coordonnées pièce ou outil (voir à ce sujet la description du paramètre <Comp>). Le paramètre suivant <GeoAx> n'est pas évalué.
			2	Les corrections vectorielles, c'est-à-dire L1, L2 et L3 peuvent être modifiées simultanément. Contrairement aux variantes "0" et "1", les valeurs de correction contenues dans <CorVal> se rapportent toutefois aux coordonnées de la composante Val_1 (voir le paramètre suivant <CorMode>) de l'outil. Une inclinaison éventuellement disponible de l'outil par rapport au système de coordonnées pièce n'a aucune influence sur la correction.
			3 - 5	Correction de la longueur d'outil L1 à L3 ($TC_DP3 à $TC_DP5) ou des valeurs correspondantes en cas d'usure, de corrections de réglage ou de somme. La valeur de correction est contenue dans <CorVal>[0]. Elle est mesurée dans les coordonnées de la composante Val_1 (voir le paramètre suivant <CorMode>) de l'outil. Une inclinaison éventuellement disponible de l'outil par rapport au système de coordonnées pièce n'a aucune influence sur la correction.
			6	Correction du rayon d'outil ($TC_DP6) ou des valeurs correspondantes en cas d'usure, de corrections de réglage ou de somme. Les bits 10 et 11 (évaluation de la spécification de diamètre ou d'usure de diamètre en tant qu'indication de rayon ou de diamètre) du paramètre machine PM20360 $MC_TOOL_PARAMETER_DEF_MASK sont pris en compte.
			7 – 11	Correction de $TC_DP7 à $TC_DP11 ou des valeurs correspondantes en cas d'usure, de corrections de réglage ou de somme. Ces paramètres sont traités comme le rayon d'outil.
		Si ce paramètre n'est pas spécifié, sa valeur est "0".		

<table>
<tr><td>4</td><td><CorMode> :</td><td colspan="3">Spécifie le type d'opération d'écriture à exécuter (facultatif)</td></tr>
<tr><td></td><td></td><td colspan="2">Type de données :</td><td>INT</td></tr>
<tr><td></td><td></td><td rowspan="4">Valeur :</td><td>0</td><td>Val_{1nouv} = <CorVal></td></tr>
<tr><td></td><td></td><td>1</td><td>Val_{1nouv} = Val_{1anc} + <CorVal></td></tr>
<tr><td></td><td></td><td>2</td><td>Val_{1nouv} = <CorVal>
Val_{2nouv} = 0</td></tr>
<tr><td></td><td></td><td>3</td><td>Val_{1nouv} = Val_{1anc} + Val_{2anc} + <CorVal>
Val_{2nouv} = 0</td></tr>
<tr><td></td><td></td><td colspan="3">La syntaxe Val_{1anc} + Val_{2anc} est à interpréter de manière symbolique. Si les deux composantes sont évaluées avec des valeurs différentes (en raison de l'état de <_Stat>), c'est-à-dire qu'une rotation est effective entre les deux composantes, Val_{2anc} est transformée avant l'addition de sorte que la longueur d'outil qui en résulte après la suppression de Val_{2nouv} et avant l'addition de <CorVal> reste inchangée.
<CorVal> se rapporte toujours à Val_1. <CorVal> est une valeur mesurée en fonction de la deuxième partie du paramètre <Comp> dans le système de coordonnées pièce (SCP) ou dans le système de coordonnées outil (SCO). C'est pourquoi il est déjà transformé le cas échéant par rapport à la composante d'outil dans laquelle il doit être pris en compte. Par conséquent, il ne peut pas être immédiatement calculé avec la valeur enregistrée mais doit être retransformé avant d'être additionné à Val_1 ou Val_2. Il peut en résulter une action de la correction dans un axe autre que celui défini par <CorComp> ou une action sur plusieurs axes.
Dans le cas où <CorComp> = 0, c'est-à-dire si <CorVal> ne contient pas de vecteur mais seulement une valeur individuelle, les opérations décrites sont exécutées dans les coordonnées dans lesquelles <CorVal> a été mesuré (SCP/SCO). Cela vaut notamment aussi pour la mise à zéro de Val_{2nouv} dans les variantes 2 et 3. Ce résultat est alors retransformé dans les coordonnées de l'outil. Cela peut avoir pour conséquence qu'aucune des valeurs de coordonnées à remettre à zéro (L1, L2, L3) n'est nulle ou que les valeurs de coordonnées qui étaient nulles auparavant sont à présent différentes de zéro. Toutefois, l'exécution successive des opérations correspondantes pour les trois axes géométriques garantit la nullité pour les trois valeurs de coordonnées de la composante à supprimer. Si l'outil n'est pas pivoté par rapport au système de coordonnées pièce, ou bien pivoté de telle sorte que toutes les composantes d'outil restent parallèles aux axes de coordonnées (permutations d'axes), la modification d'une seule coordonnée d'outil est garantie.
L'exécution successive de la même opération (<CorMode>) avec <CorComp> = 0 pour la totalité des trois axes de coordonnées dans un ordre quelconque est identique à l'exécution unique de la même opération avec <CorComp>=1.
Pour les valeurs de paramètre "0" et "1", le paramètre <Comp> doit contenir un caractère et deux caractères pour les valeurs de paramètre "2" et "3".
Exemple :
<Comp> contient la chaîne "ES", <CorMode> la valeur "2"
⇒ Correction de $réglage_{nouv}$ = <CorVal>, correction de $somme_{nouv}$ = 0
Si le paramètre <CorMode> n'est pas spécifié, sa valeur est "0".</td></tr>
</table>

5	<GeoAx> :	Spécifie l'indice de l'axe géométrique dans lequel la valeur de correction <CorVal>[0] a été mesurée (facultatif)	
		Type de données :	INT
		Plage de valeurs :	0 ... 2
		Les indices 0 à 2 se rapportent aux abscisses, aux ordonnées et aux cotes dans le niveau actif (G17/G18/G19) de l'environnement d'outil actuel. Le contenu de ce paramètre n'est évalué que lorsque le paramètre <CorComp> a la valeur "0".	
6	<Stat> :	Nom du jeu de paramètres pour la description d'un environnement d'outil (facultatif)	
		Type de données :	STRING
		Si la valeur de ce paramètre est la chaîne vide ("") ou si elle n'est pas spécifiée, l'état actuel est utilisé. Si aucun outil n'est spécifié, l'outil actuel est utilisé.	
7	<T> :	Numéro T interne de l'outil (facultatif)	
		Type de données :	INT
		Si ce paramètre n'est pas spécifié ou si sa valeur est "0", l'outil mémorisé dans <Stat> est utilisé. Si la valeur de ce paramètre est "-1", le numéro T de l'outil actif est utilisé. La spécification explicite du numéro de l'outil actif est également admise. **Remarque :** Si <Stat> n'est pas spécifié, l'état actuel est utilisé comme environnement d'outil. Compte tenu qu'avec <T> = 0, le renvoi a lieu au numéro T enregistré dans l'environnement d'outil, c'est l'outil actif qui est utilisé dans celui-ci, c'est-à-dire que les indications <T> = 0 et <T> = -1 ont la même signification dans ce cas particulier.	
8	<D> :	Numéro de tranchant de l'outil (facultatif)	
		Type de données :	INT
		Si ce paramètre n'est pas spécifié ou si sa valeur est "0", le numéro D utilisé dépend de la source du numéro T. Si le numéro T est utilisé à partir de l'environnement d'outil, le numéro D de l'environnement d'outil est lu également, autrement c'est celui de l'outil actuellement actif.	
9	<TL> :	Numéro de la correction en fonction du lieu (facultatif)	
		Type de données :	INT
		Si ce paramètre n'est pas spécifié, le numéro DL utilisé dépend de la source du numéro T. Si le numéro T est utilisé à partir de l'environnement d'outil, le numéro D de l'environnement d'outil est lu également, autrement c'est celui de l'outil actuellement actif. Si T, D et DL spécifient un outil sans corrections en fonction du lieu, aucune correction de somme ou de réglage ne doit être spécifiée dans le paramètre <Comp> (code d'erreur dans <Status>).	

Remarque

Les combinaisons possibles des trois paramètres <Comp>, <CorComp> et <CorMode> ne sont pas toutes pertinentes. Par exemple, la règle opératoire 3 dans <CorComp> exige l'indication de deux caractères dans <Comp>. Si une combinaison non admissible de paramètres est spécifiée, un code d'erreur correspondant est retourné dans <Status>.

Exemples

Exemple 1

Code de programme	Commentaire
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fraise
N30 $TC_DP3[1,1]=10,0	; Géométrie L1
N40 $TC_DP12[1,1]=1,0	; Usure L1
N50 _CORVAL[0]=0,333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"G",0,0,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> Position SCM X0.000 Y0.000 **Z1.333**
N90 M30	

<CorComp> est "0", c'est pourquoi la valeur de coordonnées effective dans le sens Z de la composante géométrique est remplacée par la valeur de correction 0,333.

La longueur totale d'outil qui en résulte est par conséquent : L1 = 0,333 + 1,000 = 1,333

Exemple 2

Code de programme	Commentaire
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fraise
N30 $TC_DP3[1,1]=10,0	; Géométrie L1
N40 $TC_DP12[1,1]=1,0	; Usure L1
N50 _CORVAL[0]=0,333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"W",0,1,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> Position SCM X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> est "1", c'est pourquoi la valeur de correction de 0,333 effective dans le sens Z est additionnée à la valeur d'usure de 1,0.

La longueur totale d'outil qui en résulte est par conséquent : L1 = 10,0 + 1,333 = 11,333

Exemple 3

Code de programme	Commentaire
N10 DEF REAL _CORVAL[3]	

Code de programme	Commentaire
N20 $TC_DP1[1,1]=120	; Fraise
N30 $TC_DP3[1,1]=10,0	; Géométrie L1
N40 $TC_DP12[1,1]=1,0	; Usure L1
N50 _CORVAL[0]=0,333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"GW",0,2,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> Position SCM X0.000 Y0.000 Z0.333
N90 M30	

<CorComp> est "2", c'est pourquoi la correction effective dans le sens Z est enregistrée dans la composante géométrique (l'ancienne valeur est écrasée), et la valeur d'usure est supprimée.

La longueur totale d'outil qui en résulte est par conséquent : L1 = 0,333 + 0,0 = 0,333

Exemple 4

Code de programme	Commentaire
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fraise
N30 $TC_DP3[1,1]=10,0	; Géométrie L1
N40 $TC_DP12[1,1]=1,0	; Usure L1
N50 _CORVAL[0]=0,333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"GW",0,3,2)**	
N80 T1 D1 X0 Y0 Z0	; ==> Position SCM X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> est "3", c'est pourquoi la valeur d'usure et la valeur de correction sont additionnées à la composante géométrique et la composante d'usure est supprimée.

La longueur totale d'outil qui en résulte est par conséquent : L1 = 11,333 + 0,0 = 11,333.

Exemple 5

Code de programme	Commentaire
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fraise
N30 $TC_DP3[1,1]=10,0	; Géométrie L1
N40 $TC_DP12[1,1]=1,0	; Usure L1
N50 _CORVAL[0]=0,333	
N60 T1 D1 G17 G0	
N70 R1=**SETTCOR(_CORVAL,"GW",0,3,0)**	
N80 T1 D1 X0 Y0 Z0	; ==> Position SCM X0.333 Y0.000 Z11.000
N90 M30	

<CorComp> est "3" comme dans l'exemple précédent ; la correction agit toutefois à présent sur l'axe géométrique avec l'indice "0" (axe X) qui est affecté dans le cas d'une fraise à la composante d'outil L3 du fait de G17. C'est pourquoi l'appel de SETTCOR n'a aucune influence sur les paramètres d'outil $TC_DP3 et $TC_DP12. À la place, la valeur de correction est enregistrée dans $TC_DP5.

Exemple 6

Code de programme	Commentaire
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; Outil de tournage
N30 $TC_DP3[1,1]=10,0	; Géométrie L1
N40 $TC_DP4[1,1]=15,0	; Géométrie L2
N50 $TC_DP12[1,1]=10,0	; Usure L1
N60 $TC_DP13[1,1]=0,0	; Usure L2
N70 _CORVAL[0]=5,0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=**SETTCOR(_CORVAL,"GW",0,3,1)**	
N110 T1 D1 X0 Y0 Z0	; ==> Position SCM X24.330 Y0.000 Z17.500
N120 M30	

L'outil est un outil de tournage. Une rotation du frame est activée dans le bloc N80 de manière que le système de coordonnées de base (SCB) soit pivoté par rapport au système de coordonnées pièce (SCP). La valeur de correction (N70) agit dans le SCP sur l'axe géométrique avec l'indice "1", c'est-à-dire, vu que G18 est actif, sur l'axe X. Compte tenu que <CorMode> = 3 s'applique, l'usure de l'outil dans le sens de l'axe X du SCP doit être nulle après l'exécution de N100.

C'est pourquoi le contenu des paramètres d'outil pertinents est le suivant à la fin du programme :

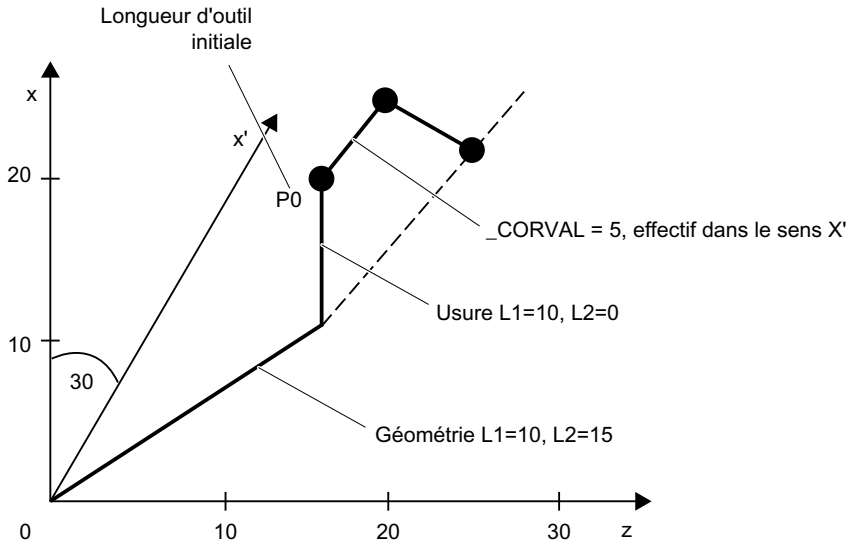
$TC_DP3[1,1] : 21,830 ; Géométrie L1

$TC_DP4[1,1] : 21,830 ; Géométrie L2

$TC_DP12[1,1] : 2,500 ; Usure L1

$TC_DP13[1,1] : -4,330 ; Usure L2

La figure ci-dessous illustre les conditions géométriques. L'usure totale incluant _CORVAL est projetée sur le sens X' dans le SCP. Il en résulte le point P2. Les coordonnées de ce point (mesurées en coordonnées X-Y) sont enregistrées dans la composante géométrique de l'outil. Le vecteur de différence P_2 - P_1 subsiste dans l'usure. De ce fait, l'usure n'a plus aucune composante dans le sens de _CORVAL.



Si l'on poursuit l'exemple de programme après N110 avec les instructions suivantes, l'usure résiduelle est totalement prise en compte dans la géométrie, compte tenu que la correction agit à présent dans l'axe Z' (paramètre <GeoAx> = 0) :

```
N120 _CORVAL[0]=0,0
N130 R1=SETTCOR(_CORVAL,"GW",0,3,0)
N140 T1 D1 X0 Y0 Z0 ; ==> Position SCM X24.330 Y0.000 Z17.500
```

Compte tenu que la nouvelle valeur de correction est "0", la longueur totale de l'outil et, de ce fait également, la position accostée dans N140 ne doit pas être modifiée. Si _CORVAL était différent de "0" dans N120, il en résulterait une nouvelle longueur totale de l'outil et, par voie de conséquence, une position modifiée dans N140. La composante d'usure de la longueur d'outil serait toutefois nulle dans ce cas, c'est-à-dire que la longueur totale de l'outil est ensuite toujours contenue dans la composante géométrique de l'outil.

Le même résultat qu'avec l'appel répété de la fonction SETTCOR avec le paramètre <CorComp> = 0 est obtenu également par un appel unique avec <CorComp> = 1 (correction vectorielle) :

Code de programme	**Commentaire**
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; Outil de tournage
N30 $TC_DP3[1,1]=10,0	; Géométrie L1
N40 $TC_DP4[1,1]=15,0	; Géométrie L2
N50 $TC_DP12[1,1]=10,0	; Usure L1
N60 $TC_DP13[1,1]=0,0	; Usure L2
N70 _CORVAL[0]=0,0	
N71 _CORVAL[1]=5,0	
N72 _CORVAL[2]=0,0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=**SETTCOR(_CORVAL,"GW",1,3,1)**	
N110 T1 D1 X0 Y0 Z0	; ==> Position SCM X24.330 Y0.000 Z17.500
N120 M30	

Dans ce cas, toutes les composantes d'usure de l'outil sont égales à zéro immédiatement après le premier appel de SETTCOR dans N100.

Exemple 7

Code de programme	Commentaire
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; Outil de tournage
N30 $TC_DP3[1,1]=10,0	; Géométrie L1
N40 $TC_DP4[1,1]=15,0	; Géométrie L2
N50 $TC_DP12[1,1]=10,0	; Usure L1
N60 $TC_DP13[1,1]=0,0	; Usure L2
N70 _CORVAL[0]=5,0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=**SETTCOR(_CORVAL,"GW",3,3)**	
N110 T1 D1 X0 Y0 Z0	; ==> Position SCM X25.000 Y0.000 Z15.000
N120 M30	

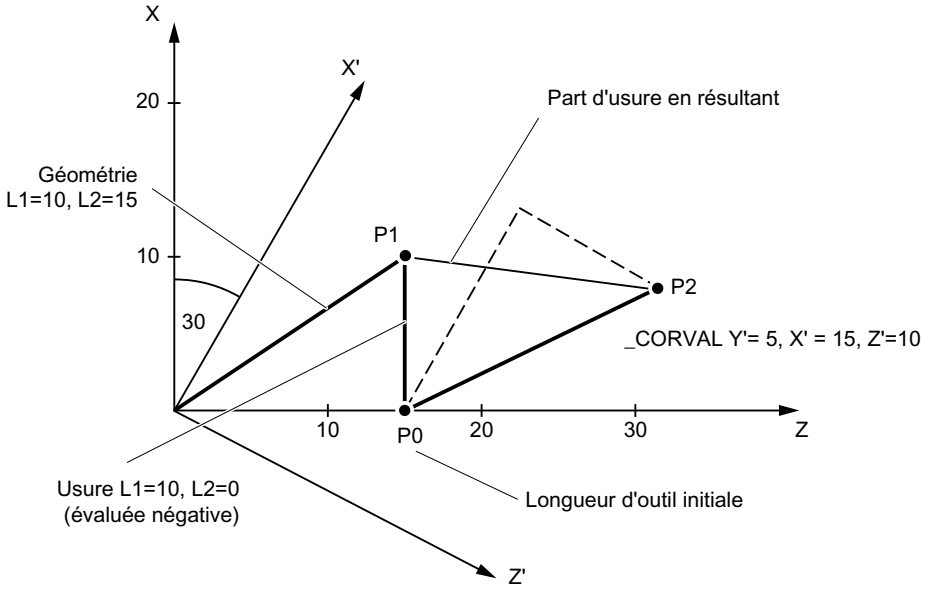
Par rapport à l'exemple 6, le paramètre est ici <CorComp> = 3, c'est pourquoi le paramètre <GeoAx> peut être omis. La valeur contenue dans _CORVAL[0] agit à présent immédiatement sur la composante d'outil L1, la rotation dans N80 n'a aucune influence sur le résultat, les composantes d'usure dans $TC_DP12 sont prises en compte avec _CORVAL[0] dans la composante géométrique. Ainsi la longueur totale de l'outil se trouve, du fait de $TC_DP13, dans la composante géométrique de l'outil dès le premier appel de SETTCOR dans N100.

Exemple 8

Code de programme	Commentaire
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; Outil de tournage
N30 $TC_DP3[1,1]=10,0	; Géométrie L1
N40 $TC_DP4[1,1]=15,0	; Géométrie L2
N50 $TC_DP5[1,1]=20,0	; Géométrie L3
N60 $TC_DP12[1,1]=10,0	; Usure L1
N70 $TC_DP13[1,1]=0,0	; Usure L2
N80 $TC_DP14[1,1]=0,0	; Usure L3
N90 $SC_WEAR_SIGN=TRUE	
N100 _CORVAL[0]=10,0	
N110 _CORVAL[1]=15,0	
N120 _CORVAL[2]=5,0	
N130 ROT Y-30	
N140 T1 D1 G18 G0	
N150 R1=**SETTCOR(_CORVAL,"W",1,1)**	
N160 T1 D1 X0 Y0 Z0	; ==> Position SCM X7.990 Y25.000 Z31.160
N170 M30	

La donnée de réglage SD42930 $SC_WEAR_SIGN est mise à 1 dans N90, c'est-à-dire que l'usure doit être évaluée avec un signe négatif. La correction est vectorielle (<CorComp> = 1) et le

vecteur de correction doit être ajouté à l'usure (<CorMode> = 1). La figure ci-dessous illustre les conditions géométriques dans le plan Z-X :



Étant donné que <CorMode> = 1, la composante géométrique de l'outil reste inchangée. Le vecteur de correction défini dans le SCP (rotation autour de l'axe Y) doit être pris en compte dans la composante d'usure, de sorte que la longueur totale de l'outil renvoie au point P_2 dans la figure 3. C'est pourquoi la composante d'usure de l'outil qui en résulte est donnée par la distance des deux points P_1 et P_2.
Mais compte tenu de l'évaluation négative de l'usure en raison de la donnée de réglage SD42930 $SC_WEAR_SIGN, la correction ainsi calculée doit être saisie dans la mémoire de correcteurs avec un signe négatif. Ainsi, le contenu des paramètres d'outil pertinents est le suivant à la fin du programme :

$TC_DP3[1,1] : 10,000 ; Géométrie L1 (inchangée)

$TC_DP4[1,1] : 15,000 ; Géométrie L2 (inchangée)

$TC_DP5[1,1] : 10,000 ; Géométrie L3 (inchangée)

$TC_DP12[1,1] : 2,010 ; Usure L1 (= 10 - 15 * cos(30) + 10 * sin(30))

$TC_DP13[1,1] : -16,160 ; Usure L2 (= -15 * sin(30) - 10 * cos(30))

$TC_DP14[1,1] : -5,000 ; Usure L3

Au niveau de la composante L3 dans le sens Y, on constate l'effet de la donnée de réglage SD42930 $SC_WEAR_SIGN sans la complication supplémentaire due à la rotation du frame.

Informations complémentaires

Outils de tournage et de rectification : Calcul de la longueur d'outil en fonction de PM20360 $MC_TOOL_PARAMETER_DEF_MASK

La façon dont l'usure ou la longueur d'outil des outils de tournage et de rectification doit être évaluée dans un axe diamétral éventuellement présent est déterminée dans le paramètre machine :

MD20360 $MC_TOOL_PARAMETER_DEF_MASK.<Bit> = <Valeur>

<Bit>	<Valeur>	Signification
0	0	Sur les outils de tournage et de rectification, le **paramètre d'usure** de l'axe transversal est pris en compte dans la **valeur de rayon** :
	1	Sur les outils de tournage et de rectification, le **paramètre d'usure** de l'axe transversal est pris en compte en tant que **valeur de diamètre** :
1	0	Sur les outils de tournage et de rectification, la **composante de longueur d'outil** de l'axe transversal est prise en compte en tant que **valeur de rayon** :
	1	Sur les outils de tournage et de rectification, la **composante de longueur d'outil** de l'axe transversal est prise en compte en tant que **valeur de diamètre** :

Si les bits concernés sont mis à 1, l'entrée associée est évaluée avec le facteur 0,5. La correction au moyen de SETTCOR est réalisée de sorte que la modification effective de la longueur totale de l'outil soit identique à celle transmise dans <CorVal>. Si une longueur est évaluée avec le facteur 0,5 sur la base du paramètre machine PM20360 $MC_TOOL_PARAMETER_DEF_MASK pour le calcul de longueur, la correction de cette composante doit donc s'effectuer avec la valeur double transférée.

Exemple

PM20360 $MC_TOOL_PARAMETER_DEF_MASK = 2 (la longueur de l'outil doit être évaluée dans l'axe diamétral avec le facteur 0,5)

L'axe X et l'axe de diamétral.

```
Code de programme                          Commentaire
N10 DEF REAL _LEN[11]
N20 DEF REAL _CORVAL[3]
N30 $TC_DP1[1,1]=500                       ; Type d'outil
N40 $TC_DP2[1,1]=2                         ; Position du tranchant
N50 $TC_DP3[1,1]=3.                        ; Géométrie - longueur 1
N60 $TC_DP4[1,1]=4.                        ; Géométrie - longueur 2
N70 $TC_DP5[1,1]=5.                        ; Géométrie - longueur 3
N80 _CORVAL[0]=1.
N90 _CORVAL[1]=1.
N100 _CORVAL[2]=1.
N110 T1 D1 G18 G0 X0 Y0 Z0                 ; ==> Position SCM X1.5 Y5 Z4
N120 R1=SETTCOR(_CORVAL,"G",1,1)
N130 T1 D1 X0 Y0 Z0                        ; ==> Position SCM X2.5 Y6 Z5
N140 R3=$TC_DP3[1,1]                       ; = 5. = (3,000 + 2.*1,000)
N150 R4=$TC_DP4[1,1]                       ; = 5. = (4,000 + 1,000)
N160 R5=$TC_DP5[1,1]                       ; = 6. = (5,000 + 1,000)
N170 M30
```

La correction de la longueur d'outil doit être de 1 mm dans chaque axe (N80 à N100). C'est pourquoi 1 mm est ajouté à la longueur initiale dans les longueurs L2 et L3. En revanche, la valeur de correction double (2 mm) est ajoutée à la longueur d'outil initiale dans L1, afin de modifier de 1 mm la longueur totale de l'outil comme exigé. Par rapport aux positions accostées dans les blocs N110 et N130, on constate que chaque position d'axe a changé de 1 mm.

3.13.14 Lecture de l'affectation des longueurs d'outil L1, L2, L3 par rapport aux axes de coordonnées (LENTOAX)

La fonction LENTOAX fournit des informations sur l'affectation des longueurs L1, L2 et L3 de l'outil **actif** par rapport aux abscisses, aux ordonnées et aux cotes. L'affectation des abscisses, ordonnées et cotes aux axes géométriques est influencée par les frames et le plan actif (G17 - G19).

Seule la composante géométrique d'un outil ($TC_DP3[<t>,<d>] à $TC_DP5[<t>,<d>]) est prise en compte, c'est-à-dire qu'une affectation d'axe d'autres composantes qui diffèrerait de celle-ci (par ex. usure) n'a aucune incidence sur le résultat.

Syntaxe

```
<Status> = LENTOAX(<AxInd>, <Matrix>[, <Coord>])
```

Principe

`LENTOAX(...)` :	Fonction prédéfinie pour la lecture de l'affectation des longueurs L1, L2 et L3 de l'outil actif aux axes de coordonnées.		
	Seule dans le bloc :		oui
`<Status>` :	Valeur de retour de la fonction. Les valeurs négatives indiquent des états d'erreur.		
	Type de données :		INT
	Valeur :	0	Fonction OK L'information qui figure dans <AxInd> est suffisante pour la description (toutes les composantes de longueur d'outil sont parallèles aux axes géométriques).
		1	La fonction est OK mais pour une description correcte le contenu de <Matrix> doit être analysé (les composantes de longueur d'outil ne sont pas parallèles aux axes géométriques).
		-1	Chaîne non valide dans le paramètre <Coord>.
		-2	Aucun outil actif.
Paramètres			

1	<AxInd> :	Si les composantes de longueur d'outil sont parallèles aux axes géométriques, les indices d'axe assignés aux composantes de longueur L1 à L3 sont retournés dans le champ **<**AxInd> : • <AxInd> [0] : abscisse • <AxInd> [2] : ordonnée • <AxInd> [2] : cote		
		Type de données :		INT[3]
		Valeur :	0	Aucune affectation disponible (l'axe n'existe pas)
			1 ... 3 ou -1 ... -3	Numéro de la longueur qui agit dans l'axe de coordonnées correspondant. Le signe est négatif lorsque la composante de longueur d'outil est orientée dans le sens négatif des coordonnées.
		Si les composantes de longueur ne sont pas toutes parallèles ou antiparallèles aux axes géométriques, l'indice est retourné dans <AxInd> à l'axe qui contient la plus grande part d'une composante de longueur d'outil. Dans ce cas (si la fonction ne génère pas une erreur pour une autre raison), la valeur retournée est <Status> = 1. La représentation des composantes de longueur d'outil L1 à L3 sur les axes géométriques 1 à 3 est alors entièrement décrite par le contenu du 2e paramètre <Matrix>.		
2	<Matrix> :	Matrice qui constitue le vecteur des longueurs d'outil (L1=1, L2=1, L3=1) dans le vecteur des axes de coordonnées (abscisse, ordonnée, cote), c'est-à-dire que les composantes de longueur d'outil sont assignés aux colonnes dans l'ordre L1, L2, L3, et les axes sont affectés aux lignes dans l'ordre abscisse, ordonnée, cote.		
		Type de données :		REAL
		Dans la matrice, tous les éléments sont toujours valides, même si l'axe géométrique faisant partie d'un axe de coordonnées n'est pas disponible, c'est-à-dire si l'entrée correspondante dans <AxInd> est égale à zéro.		
3	<Coord> :	Système de coordonnées pour lequel l'affectation s'applique (facultatif)		
		Type de données :		STRING
		Symbole :	MCS M	Représentation de la longueur d'outil dans le système de coordonnées machine
			BCS B	Représentation de la longueur d'outil dans le système de coordonnées de base
			WCS W	Représentation de la longueur d'outil dans le système de coordonnées pièce (par défaut)
			KCS K	Représentation de la longueur d'outil dans le système de coordonnées outil de la transformation cinématique
			TCS T	Représentation de la longueur d'outil dans le système de coordonnées outil
		La casse des caractères dans la chaîne (majuscules ou minuscules) est sans importance. Si le paramètre <Coord> est non spécifié, le SCP est utilisé (par défaut).		

Remarque

Dans le TCS, toutes les composantes de longueur d'outil sont toujours parallèles ou antiparallèles aux axes.

Les composantes ne peuvent être antiparallèles que si la fonction miroir est active et que la donnée de réglage suivante est définie :

SD42900 $SC_MIRROR_TOOL_LENGTH (inversion du signe de la longueur d'outil pour la fonction miroir)

Exemple

Cas standard d'outil de fraisage avec G17.

L1 agit dans Z (cote), L2 agit dans Y (ordonnée), L3 agit dans X (abscisse).

Appel de la fonction sous la forme :
`<Status>=LENTOAX(<AxInd>,<Matrix>,"WCS")`

Le paramètre de résultat <AxInd> contient alors les valeurs :

<AxInd>[0] = 3

<AxInd>[1] = 2

<AxInd>[2] = 1

ou en abrégé : (3, 2, 1)

La matrice associée (<Matrix>) est dans ce cas :

$$\text{<Matrix>} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

Le passage de G17 vers G18 ou G19 ne modifie en rien le résultat car l'affectation des composantes de longueur aux axes géométriques varie de la même façon que l'affectation des abscisses, ordonnées et cotes.

Si G17 est actif, une rotation du frame de 60 degrés autour de Z est programmée, par ex. :
`ROT Z60`

Le sens de la cote (sens Z) reste inchangé, la part principale de L2 se situe à présent dans le sens du nouvel axe X, la part principale de L1 dans le sens de l'axe Y négatif. La valeur de retour (<Status>) est donc "1", <AxInd> contient les valeurs (2, -3, 1).

La matrice associée (<Matrix>) est dans ce cas :

$$\text{<Matrix>} = \begin{pmatrix} 0 & \sin 60° & \cos 60° \\ 0 & \cos 60° & -\sin 60° \\ 1 & 0 & 0 \end{pmatrix}$$

3.14 Modes de déplacement

3.14.1 Variation de l'avance (FNORM, FLIN, FCUB, FPO)

En vue d'une définition plus flexible de l'avance, la programmation de l'avance selon DIN 66025 est étendue aux variations linéaires et cubiques.

Les variations cubiques peuvent être programmées directement ou sous la forme de splines interpolés. Par ce moyen, il est possible de programmer des variations continues et lissées, dépendant de la courbure de la pièce à usiner.

Ces variations de la vitesse permettent des variations d'accélération sans à-coup et, de ce fait, la réalisation d'états de surface plus réguliers.

Syntaxe

```
F... FNORM
F... FLIN
F... FCUB
F=FPO(...,...,...)
```

Signification

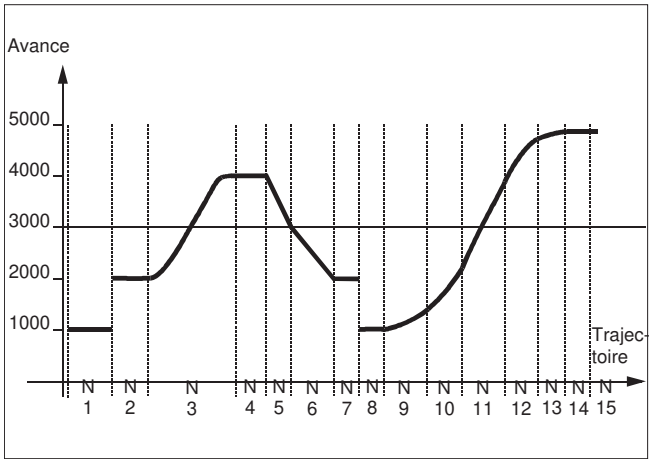
`FNORM` :	Préréglage. La valeur de l'avance est définie sur la longueur de la trajectoire du bloc et vaut alors comme valeur modale.
`FLIN` :	Variation de l'avance tangentielle **linéaire :** La valeur de l'avance varie linéairement de la valeur en début de bloc à la valeur en fin de bloc et vaut alors comme valeur modale. Ce comportement peut être combiné avec G93 et G94.
`FCUB` :	Variation de l'avance tangentielle **cubique :** Les valeurs F programmées dans les blocs - rapportées au point final de bloc - sont raccordées par un spline. Le spline débute et se termine de façon tangentielle aux valeurs d'avance programmées dans le bloc précédent, voire suivant, et agit avec G93 et G94. Si l'adresse F manque dans un bloc, c'est la dernière valeur F programmée qui sera utilisée.
`F=FPO...` :	Variation de l'avance tangentielle **définie par un polynôme :** l'adresse F décrit la variation de l'avance selon un polynôme, de la valeur courante jusqu'à la valeur en fin de bloc. La valeur finale vaut alors comme valeur modale.

Optimisation de l'avance sur des sections de trajectoire courbes

Le polynôme d'avance `F=FPO` et le spline d'avance `FCUB` doivent toujours être exécutés à vitesse de coupe constante `CFC` . De ce fait, une variation de consigne d'avance à accélération continue est générée.

Exemple : Différentes courbes d'avance

Vous trouverez, dans cet exemple, la programmation et la représentation graphique de différentes courbes d'avance.

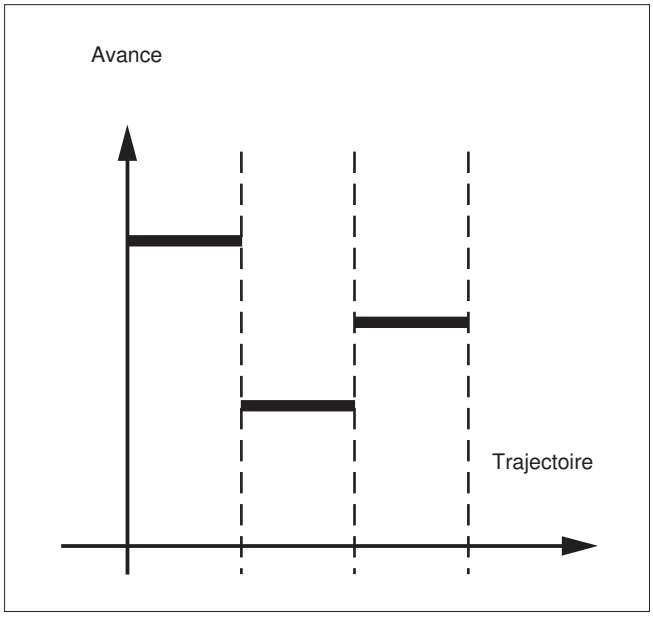


Code de programme	**Commentaire**
N1 F1000 FNORM G1 X8 G91 G64	; Courbe d'avance constante, indication en cotes relatives
N2 F2000 X7	; Variation brusque de la vitesse de consigne
N3 F=FPO(4000, 6000, -4000)	; Courbe d'avance par polynôme avec avance 4000 en fin de bloc
N4 X6	; Avance polynomiale 4000 prise comme valeur modale.
N5 F3000 FLIN X5	; Courbe d'avance linéaire
N6 F2000 X8	; Courbe d'avance linéaire
N7 X5	; Avance linéaire prise comme valeur modale
N8 F1000 FNORM X5	; Courbe d'avance constante avec variation d'accélération en échelon.
N9 F1400 FCUB X8	; Toutes les valeurs de F programmées dans les blocs suivants sont reliées par des splines.
N10 F2200 X6	
N11 F3900 X7	
N12 F4600 X7	
N13 F4900 X5	; Désactiver la courbe spline.
N14 FNORM X5	
N15 X20	

Informations complémentaires

FNORM

L'adresse d'avance F désigne l'avance tangentielle en tant que valeur constante selon DIN 66025.

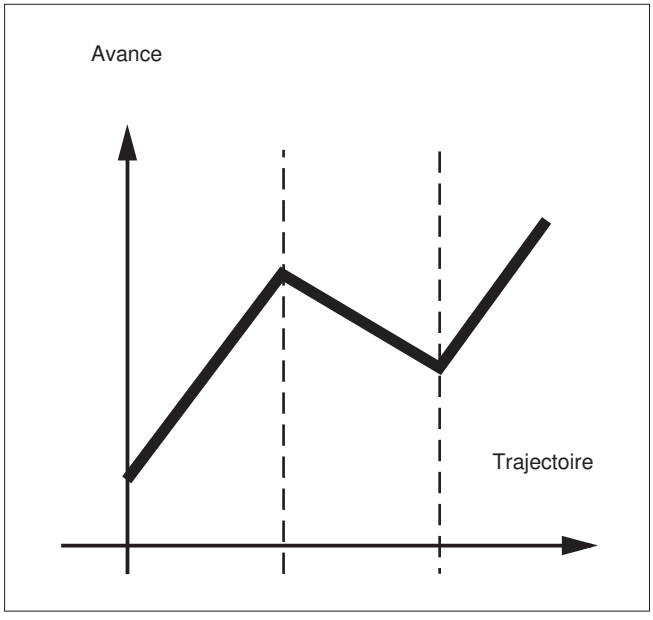


FLIN

L'avance varie de façon linéaire jusqu'à la fin du bloc, depuis la valeur d'avance courante jusqu'à la valeur F programmée.

Exemple :

```
N30 F1400 FLIN X50
```



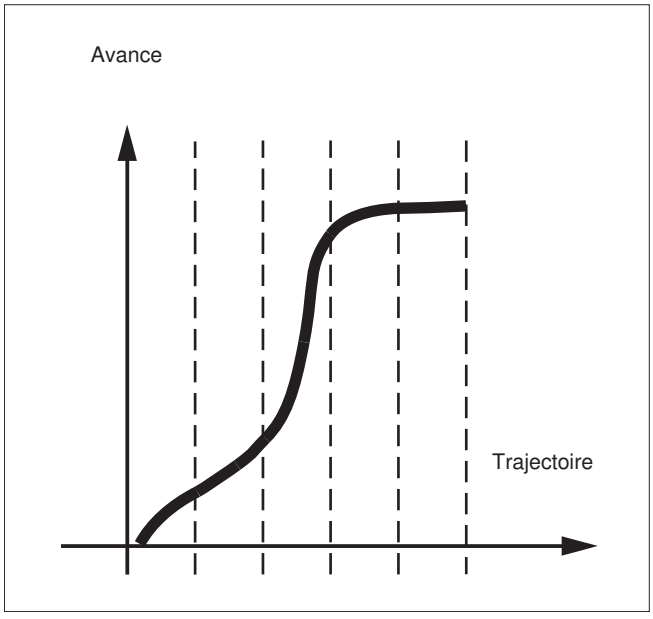
FCUB

L'avance varie de façon cubique jusqu'à la fin du bloc, depuis la valeur courante jusqu'à la valeur F programmée. La commande relie toutes les valeurs programmées dans les blocs avec un FCUB

actif par des splines. Les valeurs d'avance servent dans ce cas de points intermédiaires pour le calcul de l'interpolation spline.

Exemple :

```
N50 F1400 FCUB X50
N60 F2000 X47
N70 F3800 X52
```



F=FPO(...,...,...)

La variation de l'avance est directement programmée par le biais d'un polynôme. L'introduction des coefficients du polynôme est la même que pour l'interpolation polynomiale.

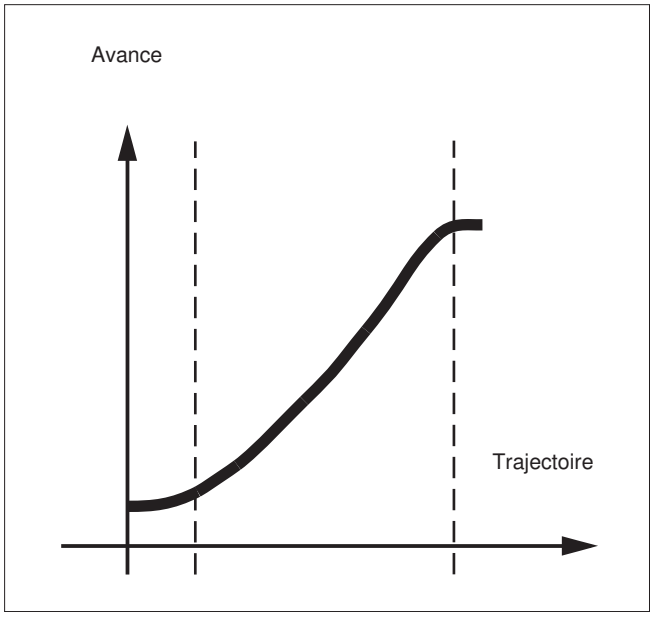
Exemple :

```
F=FPO(endfeed, quadf, cubf)
```

endfeed, quadf et cubf sont des variables précédemment définies.

endfeed :	Avance en fin de bloc
quadf :	Coefficient de polynôme de degré 2
cubf :	Coefficient de polynôme de degré 3

Quand FCUB est activée, le spline se raccorde tangentiellement en début et en fin de bloc à l'avance définie via FPO.



Autres conditions à prendre en compte

- Les fonctions programmées pour le mode de déplacement restent valables, indépendamment de la variation de l'avance.
- La variation programmable de l'avance est toujours valable de façon absolue, indépendamment de G90 ou G91.
- La variation de l'avance FLIN et FCUB fonctionne avec G93 et G94, **sauf** avec G95, G96/G961 et G97/G971.
- Lorsque le compacteur COMP... est activé, les règles ci-après sont appliquées quand plusieurs blocs sont regroupés en un segment de spline :

`FNORM` :	Pour le segment de spline, le mot F du dernier bloc qui en fait partie est valide.
`FLIN` :	Pour le segment de spline, le mot F du dernier bloc qui en fait partie est valide. La valeur F programmée est valide en fin du segment et est alors accostée de façon linéaire.
`FCUB` :	Le spline d'avance généré s'écarte au maximum des points finaux programmés d'une valeur égale à la valeur définie dans le paramètre machine PM20172 $MC_COMPRESS_VELO_TOL.
`F=FPO(...,...,...)` :	Ces blocs ne sont pas compactés.

3.14.2 Comportement à l'accélération

3.14.2.1 Mode d'accélération (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA)

Pour programmer le mode d'accélération, on dispose des instructions de programme pièce suivantes :

- "BRISK, BRISKA"
  Le déplacement des axes individuels et des axes à interpolation s'opère avec une accélération maximale jusqu'à ce que la vitesse d'avance programmée soit atteinte (**accélération sans limitation des à-coups**).
- "SOFT, SOFTA"
  Le déplacement des axes individuels et des axes à interpolation s'opère avec une accélération constante jusqu'à ce que la vitesse d'avance programmée soit atteinte (**accélération avec limitation des à-coups**).
- "DRIVE, DRIVEA"
  Le déplacement des axes individuels et des axes à interpolation s'opère avec une accélération maximale jusqu'à une limite de vitesse programmée (réglage du MD). L'accélération est ensuite réduite (réglage du MD) jusqu'à ce que la vitesse d'avance programmée soit atteinte.

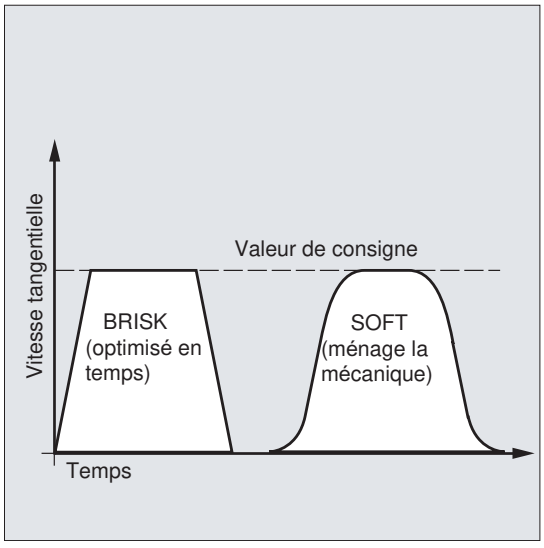


Figure 3-12 Courbes de vitesse tangentielle avec `BRISK` et `SOFT`

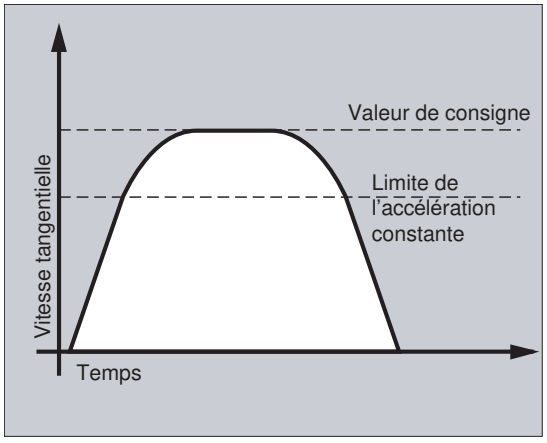


Figure 3-13 Courbe de vitesse tangentielle avec DRIVE

Syntaxe

```
BRISK
BRISKA(<axe1>,<axe2>,…)
SOFT
SOFTA(<axe1>,<axe2>,…)
DRIVE
DRIVEA(<axe1>,<axe2>,…)
```

Signification

BRISK :	Instruction d'activation de l'accélération sans limitation des à-coups pour les axes à interpolation
BRISKA :	Instruction d'activation de l'accélération sans limitation des à-coups pour les déplacements d'axes individuels (JOG, JOG/INC, axe de positionnement, axe d'oscillation, etc.)
SOFT :	Instruction d'activation de l'accélération avec limitation des à-coups pour les axes à interpolation
SOFTA :	Instruction d'activation de l'accélération avec limitation des à-coups pour les déplacements d'axes individuels (JOG, JOG/INC, axe de positionnement, axe d'oscillation, etc.)
DRIVE :	Instruction d'activation de l'accélération réduite au-dessus d'une limite de vitesse configurée (MD35220 $MA_ACCEL_REDUCTION_SPEED_POINT) pour les axes à interpolation
DRIVEA :	Instruction d'activation de l'accélération réduite au-dessus d'une limite de vitesse configurée (MD35220 $MA_ACCEL_REDUCTION_SPEED_POINT) pour les déplacements d'axes individuels (JOG, JOG/INC, axe de positionnement, axe d'oscillation, etc.)
(<axe1>,<axe2>,…) :	Axes individuels auxquels le mode d'accélération activé soit s'appliquer

Autres conditions à prendre en compte

Changement de mode d'accélération en cours d'usinage

Un changement de mode d'accélération en cours d'usinage dans un programme pièce (BRISK ↔ SOFT) entraîne un changement de bloc au niveau de la transition avec arrêt précis en fin de bloc même en contournage.

Exemples

Exemple 1 : SOFT et BRISKA

Code de programme
N10 G1 X… Y… F900 SOFT
N20 BRISKA(AX5,AX6)
...

Exemple 2 : DRIVE et DRIVEA

Code de programme
N05 DRIVE
N10 G1 X… Y… F1000
N20 DRIVEA (AX4, AX6)
...

3.14.2.2 Influence de l'accélération dans le cas des axes asservis (VELOLIMA, ACCLIMA, JERKLIMA)

Pour les couplages d'axes (Page 922) (asservissement tangentiel, déplacements conjugués, couplage par valeur pilote, réducteur électronique), le déplacement des axes/broches asservis s'effectue en fonction d'un ou de plusieurs axes/broches pilotes.

Il est possible d'influencer les limitations dynamiques des axes/broches asservis avec les fonctions VELOLIMA, ACCLIMA et JERKLIMA du programme pièce ou au moyen d'actions synchrones, même lorsque le couplage d'axes est actif.

Remarque

La fonction JERKLIMA n'est pas disponible pour tous les types de couplage.

- ACCLIMA et VELOLIMA agissent sur le déplacement global de la broche en mode broche, mais également en cas d'absence de couplage. Une limitation est affichée par l'identifiant 22 dans les variables $AC_SMAXACC_INFO ou $AC_SMAXVELO_INFO.
- En mode broche, VELOLIMA est limité à 100 pour cent.
- Ce qui suit s'applique aux couplages de broches en ce qui concerne la vitesse autorisée : Lorsque le déplacement du couplage atteint déjà la vitesse maximale et donc que le mouvement de base programmé s'arrête, l'alarme 22015 "La broche asservie a une dynamique insuffisante pour le déplacement additionnel" s'affiche.

Syntaxe

```
VELOLIMA(<axe>)=<valeur>
ACCLIMA(<axe>)=<valeur>
JERKLIMA(<axe>)=<valeur>
```

Signification

VELOLIMA :	Instruction pour la correction de la **vitesse** maximale programmée
ACCLIMA :	Instruction pour la correction de l'**accélération** maximale programmée
JERKLIMA :	Instruction pour la correction de l'**à-coup** maximum programmé
<axe> :	Axe asservi dont il fait corriger les limitations dynamiques
<valeur> :	Valeur de correction en pourcentage

Exemples

Exemple 1 : correction des limitations dynamiques pour un axe asservi (AX4)

```
Code de programme          Commentaire
...
VELOLIMA[AX4]=75           ; Correction de limitation à 75 % de la vitesse axiale
                           maximale inscrite dans le paramètre machine.
ACCLIMA[AX4]=50            ; Correction de limitation à 50 % de l'accélération axiale
                           maximale inscrite dans le paramètre machine.
JERKLIMA[AX4]=50           ; Correction de limitation à 50 % de l'à-coup axial maximum
                           inscrit dans le paramètre machine pour un déplacement avec
                           interpolation.
...
```

Exemple 2 : réducteur électronique

L'axe 4 est couplé à l'axe X par un couplage "Réducteur électronique". La capacité d'accélération de l'axe asservi est limitée à 70 % de l'accélération maximale. La vitesse maximale admise est limitée à 50 % de la vitesse maximale. Lorsque le couplage a été réalisé, la vitesse maximale admise est ramenée à 100 %.

```
Code de programme                    Commentaire
...
N120 ACCLIMA[AX4]=70                 ; Accélération maximale après réduction.
N130 VELOLIMA[AX4]=50                ; Vitesse maximale après réduction.
...
N150 EGON(AX4,"FINE",X,1,2)          ; Réalisation du couplage par réducteur élec-
                                     tronique.
...
N200 VELOLIMA[AX4]=100               ; Vitesse maximale restaurée.
...
```

Exemple 3 : influencer le couplage par valeur pilote via une action synchrone statique

L'axe 4 est couplé à l'axe X par valeur pilote. L'action synchrone statique 2 limite l'accélération à 80 % à partir de la position 100.

Code de programme	**Commentaire**
...	
N120 IDS=2 WHENEVER $AA_IM[AX4] > 100 DO ACCLIMA[AX4]=80	; Action synchrone
N130 LEADON(AX4, X, 2)	; Couplage par valeur pilote activé
...	

3.14.2.3 Activation de valeurs dynamiques spécifiques à la technologie (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH, DYNPREC)

Avec les instructions du groupe G 59 "Mode dynamique pour interpolation de trajectoire", il est possible d'activer une dynamique adaptée pour des pas d'usinage technologique distincts.

Les valeurs dynamiques et les instructions G sont configurables et dépendent donc des réglages des paramètres machine.

Informations complémentaires : Description fonctionnelle Fonctions de base

Syntaxe

Activation de valeurs dynamiques :
DYNNORM/DYNPOS/DYNROUGH/DYNSEMIFIN/DYNFINISH/DYNPREC

Remarque

Les valeurs dynamiques sont déjà actives dans le bloc dans lequel l'instruction G correspondante est programmée. Il n'y a pas d'arrêt d'usinage.

Lecture ou écriture d'un élément de tableau déterminé :
R<m>=$MA...[n,X]
$MA...[n,X]=<valeur>

Signification

DYNNORM :	Activation de la **dynamique normale**
DYNPOS :	Activation de la dynamique pour **mode positionnement, taraudage**
DYNROUGH :	Activation de la dynamique pour **ébauche**
DYNSEMIFIN :	Activation de la dynamique pour **préfinition**
DYNFINISH :	Activation de la dynamique pour **finition**
DYNPREC :	Activation de la dynamique pour **superfinition**
R<m> :	Paramètre de calcul de numéro <m>

$MA...[n,X] :	Paramètre machine avec élément de tableau déterminant la dynamique	
<n> :	Indice de tableau	
	Plage de valeurs :	0 ... 5
	0	Dynamique normale (DYNNORM)
	1	Dynamique pour le mode positionnement (DYNPOS)
	2	Dynamique pour l'ébauche (DYNROUGH)
	3	Dynamique pour la préfinition (DYNSEMIFIN)
	4	Dynamique pour la finition (DYNFINISH)
	5	Dynamique pour la superfinition (DYNPREC)
<X> :	Adresse d'axe	
<valeur> :	Valeur dynamique	

Exemples

Exemple 1 : activation de valeurs dynamiques

Code de programme	Commentaire
DYNNORM G1 X10	; Initialisation
DYNPOS G1 X10 Y20 Z30 F...	; Mode de positionnement, taraudage
DYNROUGH G1 X10 Y20 Z30 F10000	; Ébauche
DYNSEMIFIN G1 X10 Y20 Z30 F2000	; Préfinition
DYNFINISH G1 X10 Y20 Z30 F1000	; Finition
DYNPREC G1 X10 Y20 Z30 F600	; Superfinition

Exemple 2 : lecture ou écriture d'un élément de tableau déterminé

Accélération maximale pour l'ébauche, axe X.

Code de programme	Commentaire
R1=$MA_MAX_AX_ACCEL[2, X]	; Lecture
$MA_MAX_AX_ACCEL[2, X]=5	; Écriture

3.14.3 Déplacement avec commande anticipatrice (FFWON, FFWOF)

La commande anticipatrice ramène à zéro la distance de poursuite qui dépend de la vitesse dans un déplacement avec interpolation. La commande anticipatrice permet d'accroître la précision et donc, d'améliorer la qualité de l'usinage.

Syntaxe

```
FFWON
FFWOF
```

Signification

FFWON :	Instruction d'**activation** de la commande anticipatrice
FFWOF	Instruction de **désactivation** de la commande anticipatrice

Remarque

Dans les paramètres machine, on définit le type de commande anticipatrice et les axes à interpolation dont le déplacement devra être effectué avec la commande anticipatrice.

Par défaut : Commande anticipatrice dépendant de la vitesse

Option : Commande anticipatrice dépendant de l'accélération

Exemple

```
Code de programme
N10 FFWON
N20 G1 X… Y… F900 SOFT
```

3.14.4 Précision de contour programmable (CPRECON, CPRECOF)

La fonction "Précision de contour programmable" permet de réduire les défauts de trajectoire sur les contours incurvés grâce à l'adaptation automatique de la vitesse.

La précision de contour à respecter est spécifiée soit en fonction de la configuration de la machine (PM20470 $MC_MC_CPREC_WITH_FFW ; voir les spécifications du constructeur de machine), soit par la donnée de réglage $SC_CONTPREC, soit par la tolérance de contour programmée CTOL. Plus la valeur et le facteur K_V des axes géométriques sont petits, plus l'avance tangentielle est réduite au niveau des contours incurvés.

Les instructions de programme CN CPRECON et CPRECOF permettent d'activer et de désactiver la fonction "Précision de contour programmable".

Syntaxe

```
CPRECON
...
CPRECOF
```

Signification

CPRECON :	Appel d'instruction G : **activer** la fonction "Précision de contour programmable".	
	Prise d'effet :	modale
CPRECOF :	Appel d'instruction G : **désactiver** la fonction "Précision de contour programmable".	
	Prise d'effet :	modale

CPRECON et CPRECOF forment toutes deux le groupe de fonctions G 39 (Précision de contour programmable).

Remarque

Le paramètre de réglage $SC_MINFEED (avance tangentielle minimale pour CPRECON) permet à l'utilisateur de spécifier une vitesse minimale pour l'avance tangentielle.

L'avance ne peut descendre en deçà de cette valeur que si une valeur F plus faible a été programmée ou que les limitations dynamiques des axes nécessitent une vitesse tangentielle inférieure.

Remarque

La fonction "Précision de contour programmable" ne prend en considération que les axes géométriques de la trajectoire. Elle n'a aucune incidence sur les vitesses des axes de positionnement.

Exemple

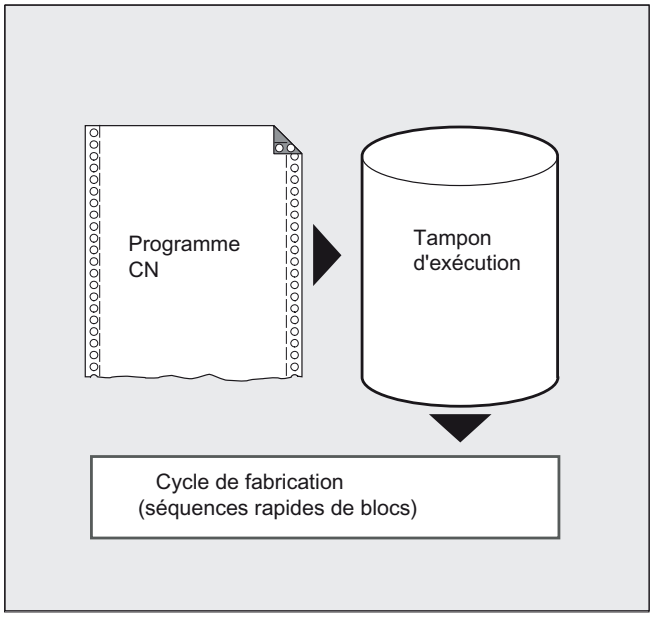
Code de programme	**Commentaire**
N10 G0 X0 Y0	
N20 CPRECON	; Activation de la "précision de contour programmable".
N30 G1 G64 X100 F10000	; Usinage à 10 m/min en contournage.
N40 G3 Y20 J10	; Limitation automatique de l'avance dans le bloc à interpolation circulaire.
N50 G1 X0	; Avance de nouveau sans limitation (10 m/min).
...	
N100 CPRECOF	; Désactivation de la "précision de contour programmable".
N110 G0 ...	

Voir aussi

Programmation d'une tolérance de contour/orientation (CTOL, OTOL, ATOL) (Page 894)

3.14.5 Exécution du programme avec une mémoire tampon (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE)

Selon la configuration, la commande dispose d'une mémoire tampon d'une certaine capacité, capable de stocker les blocs prétraités afin de pouvoir ensuite les transmettre à un rythme rapide lors du déroulement de l'usinage. Ainsi, on peut effectuer des trajets courts à grande vitesse. La mémoire tampon est remplie au fur et à mesure que le permet la commande.



Repérage de la phase d'exécution

La section d'exécution à mémoriser temporairement dans la mémoire tampon est repérée au début par "STOPFIFO" et à la fin par "STARTFIFO". L'exécution des blocs prétraités et mémorisés dans la mémoire tampon débute seulement après l'instruction "STARTFIFO" ou quand la mémoire tampon est pleine.

Contrôle automatique de la mémoire tampon

L'instruction "FIFOCTRL" permet d'appeler le contrôle automatique de la mémoire tampon. "FIFOCTRL" agit d'abord comme "STOPFIFO". À chaque programmation, le traitement de la mémoire tampon a lieu seulement lorsqu'elle celle-ci est pleine. Il n'en est pas de même dans le cas de la marche à vide de la mémoire tampon : à partir d'un niveau de remplissage égal à 2/3, la vitesse tangentielle est progressivement réduite avec "FIFOCTRL", afin d'éviter une marche à vide complète et une décélération jusqu'à l'arrêt.

Arrêt du prétraitement des blocs

Le traitement et la mise en mémoire tampon des blocs s'arrêtent lorsque l'instruction "STOPRE" est programmée dans le bloc. Le bloc suivant n'est exécuté que lorsque tous les blocs prétraités

et mémorisés auparavant ont été exécutés en entier. Un arrêt précis (comme G9) a lieu dans le bloc précédent.

IMPORTANT
Interruption de programme Lorsque la correction d'outil et activée et dans le cas d'interpolations de type spline, il est recommandé de ne pas programmer "STOPRE" afin d'éviter d'interrompre des séquences de blocs contiguës.

Syntaxe

Tableau 3-3 Repérage de la section d'exécution :

```
STOPFIFO
...
STARTFIFO
```

Tableau 3-4 Contrôle automatique de la mémoire tampon :

```
...
FIFOCTRL
...
```

Tableau 3-5 Arrêt du prétraitement des blocs :

```
...
STOPRE
...
```

Remarque

Les instructions "STOPFIFO", "STARTFIFO", "FIFOCTRL" et "STOPRE" doivent être programmées dans un bloc propre.

Signification

STOPFIFO :	"STOPFIFO" repère le début de la section d'exécution à stocker temporairement dans la mémoire tampon. Avec "STOPFIFO", l'exécution est arrêtée et la mémoire tampon remplie jusqu'à ce que : • "STARTFIFO" ou "STOPRE" soit détecté ou • la mémoire tampon soit pleine ou • la fin de programme soit atteinte.
STARTFIFO :	"STARTFIFO" démarre l'exécution rapide de la section de traitement parallèlement au remplissage de la mémoire tampon
FIFOCTRL :	Activation du contrôle automatique de la mémoire tampon
STOPRE :	Arrêter le prétraitement des blocs

Remarque

Le remplissage de la mémoire tampon n'est ni effectué, ni interrompu lorsque la section d'exécution contient des instructions qui nécessitent un fonctionnement sans tampon (accostage du point de référence, fonctions de mesure, ...).

Remarque

Lors de l'accès à des données d'état de la machine ($SA...), la commande déclenche un arrêt du prétraitement des blocs.

Exemple : arrêter le prétraitement des blocs

Code de programme	Commentaire
...	
N30 MEAW=1 G1 F1000 X100 Y100 Z50	; Bloc de mesure avec palpeur de mesure de la première entrée de mesure et interpolation linéaire.
N40 STOPRE	; Arrêt du prétraitement des blocs.
...	

3.14.6 Définition de sections d'arrêt temporisé (DELAYFSTON, DELAYFSTOF)

Les procédures DELAYFSTON et DELAYFSTOF prédéfinies servent à définir une section interruptible sous conditions dans le programme pièce (section d'arrêt temporisé).

Remarque

DELAYFSTON et DELAYFSTOF ne sont **pas** autorisés dans les actions synchrones !

Syntaxe

```
DELAYFSTON
...
DELAYFSTOF
```

Signification

DELAYFSTON :	Définition du début d'une section d'arrêt temporisé	
	Uniquement dans le bloc :	oui
DELAYFSTOF :	Définition de la fin d'une section d'arrêt temporisé	
	Uniquement dans le bloc :	oui

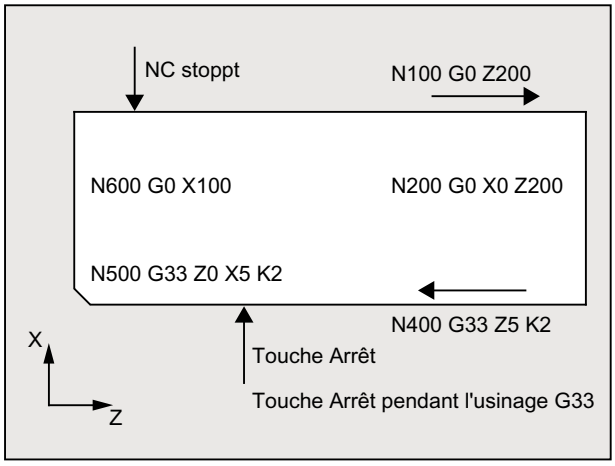
Exemple de programmation

Dans une boucle, le bloc de programme suivant est répété :

Code de programme

```
...
N99 MY_LOOP:
N100 G0 Z200
N200 G0 X0 Z200
N300 DELAYFSTON
N400 G33 Z5 K2 M3 S1000
N500 G33 Z0 X5 K3
N600 G0 X100
N700 DELAYFSTOF
N800 GOTOB MY_LOOP
...
```

On peut voir sur la figure suivante que l'utilisateur appuie sur "Arrêt" dans la section d'arrêt temporisé et que la CN commence l'opération de freinage à l'extérieur de la section d'arrêt temporisé, c'est-à-dire dans le bloc N100. Ainsi, la CN s'arrête au début de N100.



informations complémentaires

Fin de sous-programme

A la fin du sous-programme dans lequel DELAYFSTON a été appelée, l'instruction DELAYFSTOF est activée de manière implicite.

Imbrication

Si le sous-programme 1 appelle le sous-programme 2 dans une section d'arrêt temporisé, la totalité du sous-programme 2 forme une section d'arrêt temporisé. Notamment, l'instruction DELAYFSTOF est sans effet dans le sous-programme 2.

Exemple :

Code de programme	Commentaire
N10010 **DELAYFSTON**	; Blocs avec N10xxx comme niveau de programme 1.
N10020 R1 = R1 + 1	
N10030 G4 F1	; Début de la section d'arrêt temporisé.
...	
N10040 sous-programme2	
...	
...	; Interprétation du sous-programme 2.
N20010 **DELAYFSTON**	; Sans effet, nouveau départ, 2ème niveau.
...	
N20020 **DELAYFSTOF**	; Sans effet, fin dans un autre niveau.
N20030 RET	
N10050 **DELAYFSTOF**	; Fin de la section d'arrêt temporisé dans le même niveau.
...	
N10060 R2 = R2 + 2	
N10070 G4 F1	; Fin de la section d'arrêt temporisé. Les arrêts s'activent directement à partir de maintenant.

Variables système

Les variables système suivantes permettent d'interroger le système pour savoir si l'exécution du programme pièce se trouve à l'instant donné dans une section d'arrêt temporisé.

- dans le programme pièce avec $P_DELAYFST
- dans les actions synchrones avec $AC_DELAYFST

Valeur	Signification
0	Section d'arrêt temporisé inactive
1	Section d'arrêt temporisé active

3.14.7 Inhibition d'une partie de programme pour SERUPRO (IPTRLOCK, IPTRUNLOCK)

Pour certaines situations mécaniques complexes, il est nécessaire d'interdire l'opération de recherche de bloc SERUPRO sur la machine.

Avec un pointeur d'interruption programmable, il est possible via la "recherche de points d'interruption", d'intervenir avant la position interdite à la recherche.

Il est également possible de définir des plages interdites à la recherche dans les plages de programme pièce dans lesquelles la CN ne peut pas reprendre le traitement. Avec l'abandon de programme, la CN note le bloc traité en dernier sur lequel peut être effectuée la recherche via l'interface utilisateur IHM.

Syntaxe

```
IPTRLOCK
IPTRUNLOCK
```

Les instructions se trouvent seules sur une ligne du programme pièce et permettent l'utilisation d'un pointeur d'interruption programmable

Signification

`IPTRLOCK:`	Début de la section de programme interdite à la recherche
`IPTRUNLOCK:`	Fin de la section de programme interdite à la recherche

Ces deux instructions ne sont autorisées que dans les programmes pièce et **non pas** dans les actions synchrones.

Exemple

Imbrication de sections de programme interdites à la recherche dans deux niveaux de programmes avec l'instruction implicite "IPTRUNLOCK". L'instruction implicite "IPTRUNLOCK" du sous-programme 1 termine la plage interdite à la recherche.

Code de programme	Commentaire
`N10010 IPTRLOCK()`	

Code de programme	Commentaire
N10020 R1 = R1 + 1	
N10030 G4 F1	; Bloc d'arrêt qui commence la section de programme interdite à la recherche.
...	
N10040 sous-programme2	
...	; Interprétation du sous-programme 2.
N20010 IPTRLOCK ()	; Sans effet, nouveau départ.
...	
N20020 IPTRUNLOCK ()	; Sans effet, fin dans un autre niveau.
N20030 RET	
...	
N10060 R2 = R2 + 2	
N10070 RET	; Fin de la section de programme interdite à la recherche.
N100 G4 F2	; Le programme principal reprend.

Le pointeur d'interruption fournit alors à nouveau une interruption sur 100.

Informations complémentaires

Saisie et recherche de plages interdites à la recherche

Les sections de programme interdites à la recherche sont identifiées à l'aide des instructions "IPTRLOCK" et "IPTRUNLOCK".

L'instruction "IPTRLOCK" gèle le pointeur d'interruption sur un mode bloc par bloc exécutable (SB1). Ce bloc est désigné dans ce qui suit par "bloc d'arrêt". Si une interruption de programme se produit après "IPTRLOCK", la recherche est possible sur l'interface utilisateur IHM après ce bloc d'arrêt.

Reprise sur le bloc courant

Avec "IPTRUNLOCK", le pointeur d'interruption est placé au point d'interruption sur le bloc courant pour la section de programme suivante.

Après qu'une destination de recherche a été trouvée, il est possible de répéter une nouvelle destination de recherche avec le même bloc d'arrêt.

Un pointeur d'interruption édité par l'utilisateur doit être supprimé à nouveau via l'IHM.

Règles relatives aux imbrications

Les points suivants régissent l'interaction des instructions "IPTRLOCK" et "IPTRUNLOCK" avec les imbrications et la fin de sous-programme :

1. À la fin du sous-programme dans lequel "IPTRLOCK" a été appelée, l'instruction "IPTRUNLOCK" est activée de manière implicite.
2. Dans une plage interdite à la recherche, "IPTRLOCK" reste sans effet.
3. Si le sous-programme1 appelle le sous-programme2 dans une plage interdite à la recherche, le sous-programme2 reste complètement interdit à la recherche. Notamment, l'instruction "IPTRUNLOCK" est sans effet dans le sous-programme2.

Informations complémentaires : Description fonctionnelle Fonctions de base

Variable système

La présence d'une plage interdite à la recherche peut être déterminée à l'aide de l'instruction "$P_IPTRLOCK" dans le programme pièce.

Pointeur d'interruption automatique

La fonction de pointeur d'interruption automatique définit automatiquement comme interdit à la recherche un type de couplage précédemment défini. Le paramètre machine active le pointeur d'interruption automatique pour

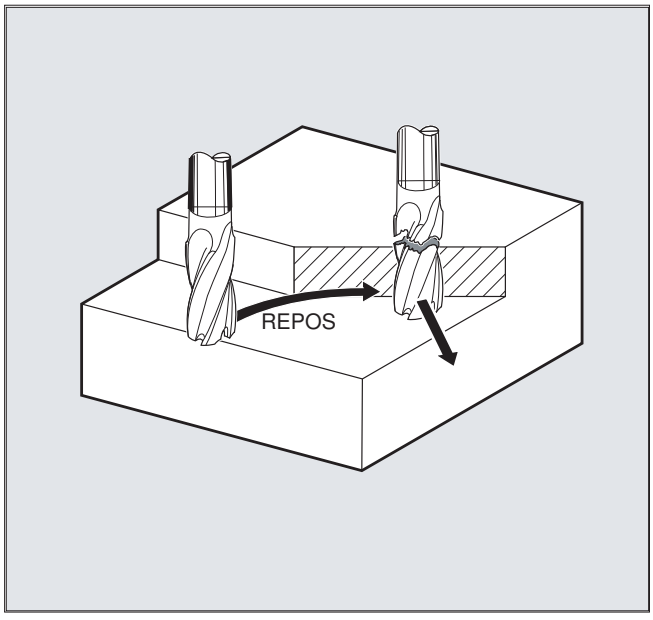
- Réducteur électronique pour "EGON"
- Couplage des axes par valeur pilote pour "LEADON"

Si le pointeur d'interruption programmé et le pointeur d'interruption automatique réglé à l'aide du paramètre machine se chevauchent, la plage interdite à la recherche la plus grande possible est créée.

3.14.8 Réaccostage du contour (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)

Si vous stoppez le programme en cours d'usinage et que vous dégagez l'outil, par exemple à cause d'un bris d'outil ou parce que vous voulez effectuer une mesure, vous pouvez par la suite réaccoster, à l'aide du programme, le contour à un point de votre choix.

La commande `REPOS` a le même effet dans un ASUP qu'un retour de sous-programme (par exemple M17). Les blocs suivants ne sont plus exécutés. Pour de plus amples informations sur l'interruption de l'exécution d'un programme, voir aussi "Routine d'interruption (ASUP) (Page 551)".



Syntaxe

```
REPOSA RMIBL DISPR=…
REPOSA RMBBL
REPOSA RMEBL
REPOSA RMNBL
REPOSL RMIBL DISPR=…
REPOSL RMBBL
REPOSL RMEBL
REPOSL RMNBL
REPOSQ RMIBL DISPR=… DISR=…
REPOSQ RMBBL DISR=…
REPOSQ RMEBL DISR=…
REPOSQA DISR=…
REPOSH RMIBL DISPR=… DISR=…
REPOSH RMBBL DISR=…
REPOSH RMEBL DISR=…
REPOSHA DISR=…
```

Signification

Sélection du trajet d'accostage

`REPOSA` :	Réaccostage du contour avec les axes géométriques sur une droite. Tous les autres axes de canal sont également repositionnés.
`REPOSL` :	Réaccostage du contour avec les axes géométriques sur une droite. Tous les autres axes de canal doivent être programmés explicitement.
`REPOSQ DISR=…` :	Réaccostage du contour avec les axes géométriques sur un quart de cercle avec rayon DISR. Tous les autres axes de canal doivent être programmés explicitement.
`REPOSQA DISR=…` :	Réaccostage du contour avec les axes géométriques sur un quart de cercle avec rayon DISR. Tous les autres axes de canal sont également repositionnés.
`REPOSH DISR=…` :	Réaccostage du contour avec les axes géométriques sur un demi-cercle avec diamètre DISR. Tous les autres axes de canal doivent être programmés explicitement.
`REPOSHA DISR=…` :	Réaccostage du contour avec les axes géométriques sur un demi-cercle avec rayon DISR. Tous les autres axes de canal sont également repositionnés.

Sélection du point de réaccostage

`RMIBL` :	Accostage du point d'interruption
`RMIBL DISPR=…` :	Point d'entrée à une distance DISPR en mm/inch avant le point d'interruption
`RMBBL` :	Accostage du point de début de bloc
`RMEBL` :	Accostage du point de fin de bloc
`RMEBL DISPR=…` :	Accostage du point de fin de bloc à une distance DISPR avant le point final
`RMNBL` :	Accostage au point de contour suivant
`A0 B0 C0` :	Axes dans lesquels le déplacement doit avoir lieu

Remarque

Compatibilité

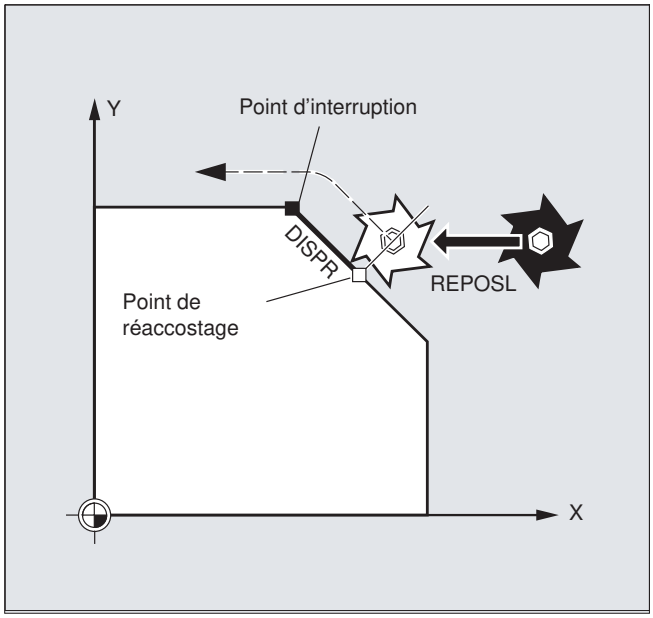
Pour préserver la compatibilité avec des versions logicielles antérieures, le mode d'accostage `REPOS` peut toujours être programmé au moyen des instructions G modales `RMI`, `RMB`, `RME` et `RMN`. Lors d'une utilisation au sein d'un ASUP, celui-ci doit être doté d'un attribut `SAVE` dans l'instruction `PROC`. Dans le cas contraire, le mode d'accostage `REPOS` modal, s'il diffère du préréglage de `RMI`, aurait une incidence sur des actions `REPOS` ultérieures.

Réaccostage du contour sur une droite, REPOSA, REPOSL

L'outil rallie le point de réaccostage sur une ligne droite.

Exemple

```
REPOSL RMIBL DISPR=6 F400
```

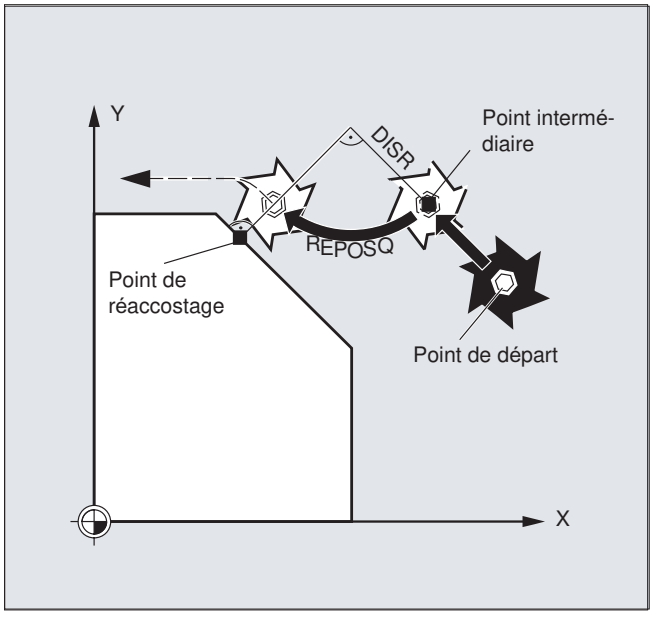


Réaccostage du contour avec quart de cercle, REPOSQ, REPOSQA

L'outil rallie le point de réaccostage sur un quart de cercle de rayon`DISR=....` La commande calcule automatiquement le point intermédiaire nécessaire entre le point de départ et le point de réaccostage.

Exemple

```
REPOSQ RMIBL DISR=10 F400
```

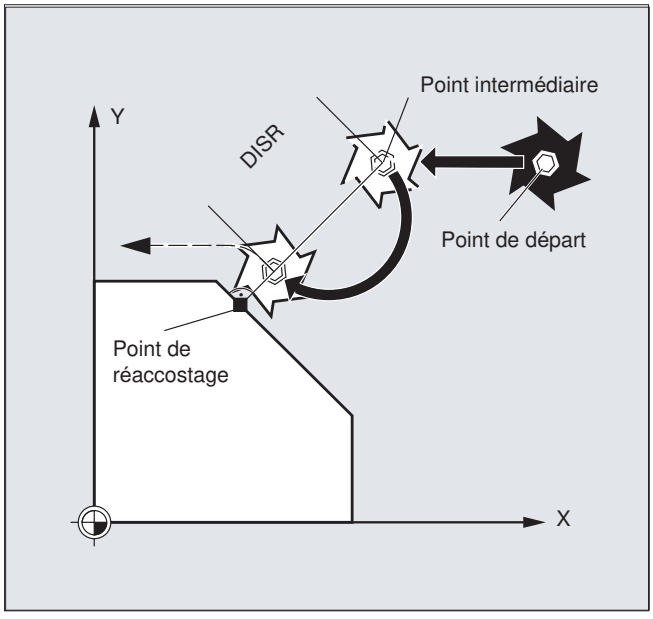


Réaccostage du contour avec demi-cercle, REPOSH, REPOSHA

L'outil rallie le point de réaccostage sur un demi-cercle de diamètre `DISR=...`. La commande calcule automatiquement le point intermédiaire nécessaire entre le point de départ et le point de réaccostage.

Exemple

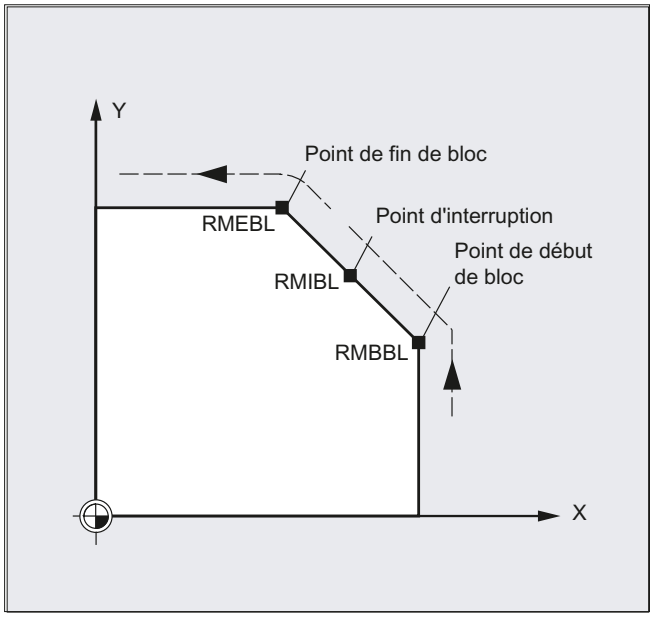
```
REPOSH RMIBL DISR=20 F400
```



Définition d'un point de réaccostage (sauf pour l'accostage SERUPRO sous RMNBL)

En fonction du bloc CN dans lequel l'interruption a eu lieu, vous avez le choix entre trois points de réaccostage :

- RMIBL, point d'interruption
- RMBBL, point de début de bloc ou dernier point final
- RMEBL, point de fin de bloc



Avec RMIBL DISPR=... ou RMEBL DISPR=..., vous pouvez définir un point de réaccostage se trouvant avant le point d'interruption ou le point de fin de bloc.

Avec DISPR=..., vous programmez, en mm/inch, la distance sur le contour entre le point de réaccostage et le point d'interruption ou le point de fin de bloc. Ce point ne peut pas figurer au-delà du point de début de bloc – même pour les plus grandes valeurs.

Si DISPR=... n'est pas programmé, DISPR=0 s'applique, c'est-à-dire que le réaccostage a lieu au point d'interruption (pour RMIBL) ou au point de fin de bloc (pour RMEBL).

Signe de DISPR

Le signe de DISPR est exploité. Si le signe est positif, le comportement est le même que précédemment.

Si le signe est négatif, le réaccostage a lieu derrière le point d'interruption ou, pour RMBBL, derrière le point de départ.

La distance point d'interruption-point de réaccostage est la valeur absolue de DISPR. Ce point ne peut pas se trouver au-delà du point de fin de bloc, même pour les plus grandes valeurs absolues.

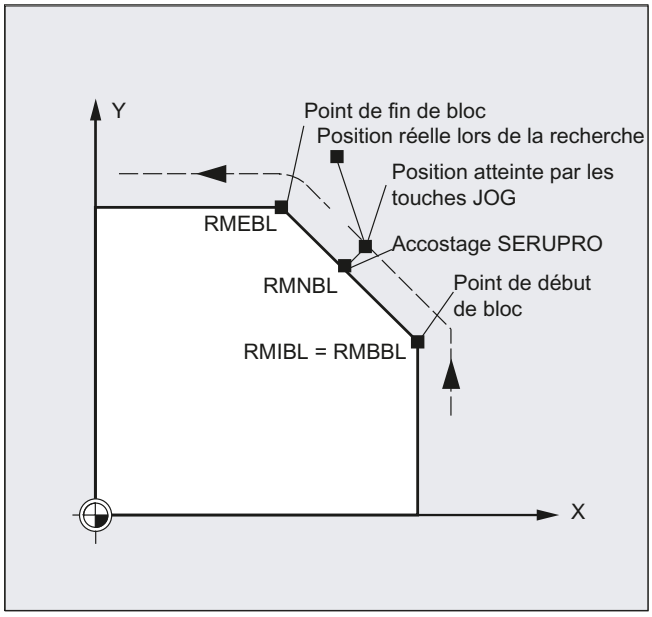
Exemple d'application :

Un capteur surveille les abords d'une griffe de serrage. Une ASUP permettant de contourner la griffe de serrage est déclenchée.

Ensuite, un repositionnement à un point situé derrière la griffe de serrage avec DISPR négatif a lieu et le programme se poursuit.

Accostage SERUPRO sous RMNBL

Si, lors de l'usinage, une interruption a été forcée sur une position quelconque, la distance la plus courte depuis la position d'interruption est parcourue avec l'accostage SERUPRO sous RMNBL afin de n'exécuter ensuite que la distance restant à parcourir. L'utilisateur lance une opération SERUPRO sur le bloc d'interruption et se positionne, à l'aide des touches JOG, avant la position défectueuse du bloc recherché.



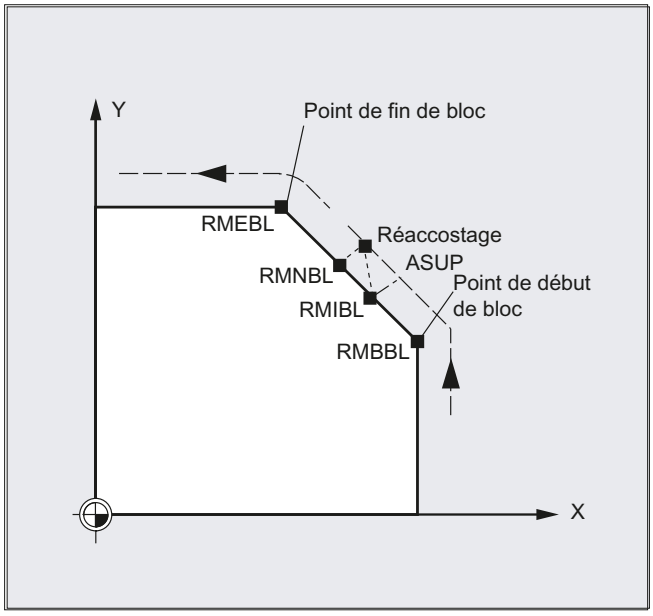
Remarque

SERUPRO

Pour SERUPRO, RMIBL et RMBBL sont identiques. RMNBL n'est pas seulement limité à SERUPRO, il est valide de manière générale.

Accostage au point de contour RMNBL suivant

Au moment de l'interprétation de REPOSA après une interruption, le réaccostage avec RMNBL n'a pas encore complètement commencé, seule la distance restant à parcourir a été traitée. C'est le point de contour suivant du bloc interrompu qui est réaccosté.



Etat du mode REPOS valide

Le mode REPOS valide du bloc interrompu peut être lu via des actions synchrones avec la $AC_REPOS_PATH_MODE :

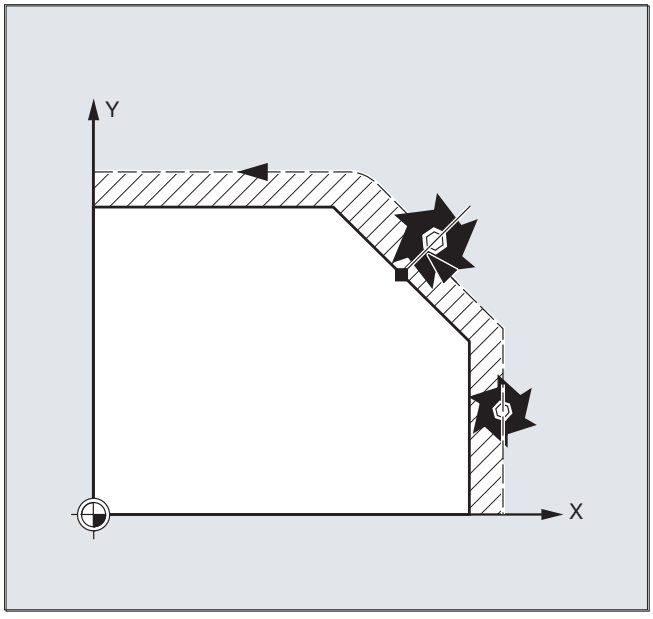
0 Accostage non défini
1 RMBBL : accostage au début
2 RMIBL : accostage au point d'interruption
3 RMEBL : accostage au point de fin de bloc
4 RMNBL : accostage au point de contour suivant du bloc interrompu

accostage avec un nouvel outil

Si vous avez stoppé l'exécution du programme suite à un bris d'outil :

En programmant le nouveau numéro D, vous poursuivez l'exécution du programme à partir du point de réaccostage et avec des valeurs de correction d'outil modifiées.

Suite aux différentes valeurs de correction d'outil, le point d'interruption risque de ne pas pouvoir être réaccosté. Dans ce cas, c'est le point le plus proche du point d'interruption qui devient point de réaccostage du contour (le cas échéant avec une modification DISPR).



Accostage du contour

Le déplacement à effectuer pour réaccoster le contour est programmable. Indiquez zéro pour les adresses des axes à déplacer.

Avec les instructions REPOSA, REPOSQA et REPOSHA, tous les axes sont repositionnés automatiquement. Il n'est pas nécessaire de préciser les axes.

Quand on programme REPOSL, REPOSQ et REPOSH, tous les axes géométriques se déplacent automatiquement, autrement dit sans qu'il soit nécessaire de le préciser dans l'instruction. Tous les autres axes à repositionner sont à préciser dans l'instruction.

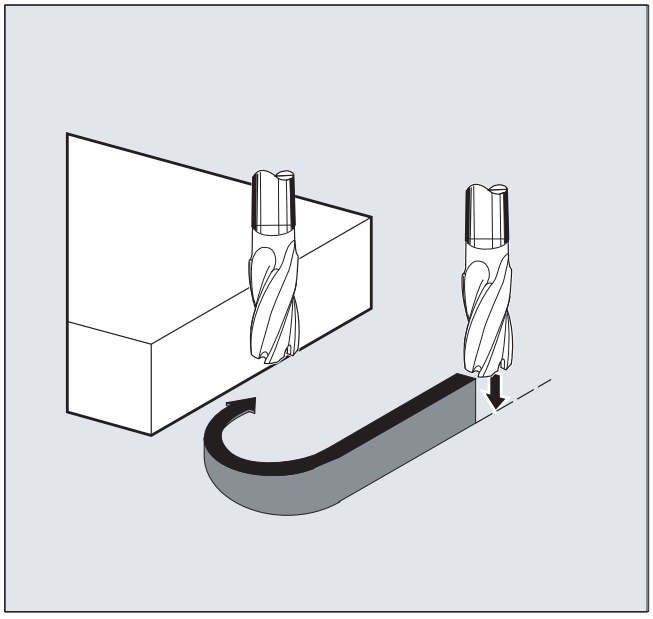
Les points suivants s'appliquent pour les mouvements circulaires REPOSH et REPOSQ :

Le mouvement circulaire est décrit dans le plan de travail G17 à G19 indiqué.

Si vous indiquez le troisième axe géométrique (pénétration) dans le bloc d'accostage et si la position de l'outil et la position programmée dans la direction de pénétration ne concordent pas, le réaccostage du contour se fera suivant un mouvement hélicoïdal.

Dans les cas suivants, il y a commutation automatique sur accostage linéaire REPOSL :

- Vous n'avez indiqué aucune valeur pour DISR.
- Il n'existe aucun sens d'accostage défini (interruption de l'exécution du programme dans un bloc ne contenant aucune information de déplacement).
- Le sens d'accostage est perpendiculaire au plan de travail courant.



3.14.9 Correction du pilotage des déplacements

3.14.9.1 Adaptation de la vitesse d'axe ou de la vitesse de broche maximale (VELOLIM)

L'instruction VELOLIM permet, dans le programme pièce, de réduire la vitesse maximale possible d'un axe ou la vitesse de rotation maximale possible d'une broche en fonction du rapport de transmission réglée au moyen d'un paramètre machine.

Prise d'effet

VELOLIM est effectif :

- dans les modes de fonctionnement AUTOMATIQUE
- sur des axes à interpolation et de positionnement
- sur des broches en mode broche / mode axe.

Syntaxe

```
VELOLIM[<axe>]=<valeur>
```

Signification

VELOLIM :	Adaptation de la valeur limite de vitesse linéaire ou de rotation
<Axe> :	Axe ou broche dont la valeur limite de vitesse linéaire ou de rotation doit être adaptée MD30455 $MA_MISC_FUNCTION_MASK, Bit 6 permet de configurer si "VELOLIM" prend effet indépendamment de son utilisation actuelle en tant que broche ou axe (bit 6 = 1) ou bien s'il doit pouvoir être programmé individuellement pour chaque mode de fonctionnement (bit 6 = 0). Si un effet distinct est configuré, la sélection a lieu lors de la programmation au moyen du descripteur : • Descripteur de broche pour modes broche : S<n> • Descripteur d'axe pour le mode axe, par ex. : C
<valeur> :	Valeur de correction en pourcentage Pour les axes ou les broches en mode axe avec le réglage MD30455 bit 6 = 0, la valeur de correction se rapporte à la vitesse d'axe maximale configurée (MD32000 $MA_MAX_AX_VELO). Pour les broches en mode broche ou axe et le réglage MD30455 bit 6 = 1, la valeur de correction se rapporte à la vitesse de rotation maximale du rapport de transmission activé (MD35130 $MA_GEAR_STEP_MAX_VELO_LIMIT[<n>]).
	Plage de valeurs : 1 ... 100
	La valeur 100 n'entraîne aucune correction de la vitesse ni de la vitesse de rotation.

Remarque

Comportement à la fin du programme pièce et lors d'un Reset de canal

Le comportement de VELOLIM à la fin du programme pièce ou lors d'un Reset de canal dépend du réglage du bit 0 dans le paramètre machine MD32320 $MA_DYN_LIMIT_RESET_MASK.

Remarque

Limitation de la vitesse de rotation en mode broche

Une limitation de vitesse activée par VELOLIM (moins de 100 %) peut être détectée en mode broche au moyen des variables système suivantes :

- $AC_SMAXVELO (vitesse de rotation maximale possible de la broche)
- $AC_SMAXVELO_INFO (identificateur de la cause de la limitation de vitesse)

Exemples

Exemple 1 : limitation de vitesse d'un axe machine

```
Code de programme          Commentaire
...
N70 VELOLIM[X]=80          ; Dans la direction X, le chariot de déplacement axial ne
                           doit se déplacer qu'au maximum à 80 % de la vitesse autori-
                           sée pour l'axe.
...
```

Exemple 2 : Limitation de la vitesse de rotation broche 1 (AX5)

Configuration :

- MD35130 $MA_GEAR_STEP_MAX_VELO_LIMIT[1, AX5] = 1000
  (Vitesse de rotation maximale du rapport de transmission 1 = 1000 tr/min)
- MD30455 $MA_MISC_FUNCTION_MASK[AX5], bit 6 = 1
  (Programmation de VELOLIM effective en même temps pour le mode broche et le mode axe, indépendamment du descripteur programmé)

Programmation :

Code de programme	Commentaire
`N05 VELOLIM[S1]=90`	`; Limitation de la vitesse de rotation maximale de la broche 1 à 90 % de 1000 tr/min.`
`...`	
`N50 VELOLIM[C]=45`	`; Limitation de la vitesse de rotation maximale de la broche 1 à 45 % de 1000 tr/min, C étant le descripteur d'axe de S1.`
`...`	

3.14.9.2 Adaptation de l'à-coup d'axe maximal (JERKLIM)

L'instruction JERKLIM permet, dans un programme pièce, de réduire ou d'augmenter l'à-coup maximal possible d'un axe réglé par un paramètre machine lors de déplacements avec interpolation dans les sections critiques d'un programme.

Condition

Il est nécessaire que le mode d'accélération SOFT soit activé.

Prise d'effet

JERKLIM est effectif :

- dans les modes de fonctionnement AUTOMATIQUE
- uniquement sur des axes à interpolation

Syntaxe

```
JERKLIM[<axe>]=<valeur>
```

Signification

`JERKLIM` :	Adaptation de la valeur limite d'à-coup
`<Axe>` :	Axe machine dont la valeur limite d'à-coup doit être adaptée

<valeur> :	Valeur de correction en pourcentage La valeur de correction se rapporte à l'à-coup maximal configuré de l'axe lors d'un déplacement avec interpolation (MD32431 $MA_MAX_AX_JERK).	
	Plage de valeurs :	1 ... 200
	La valeur 100 n'entraîne aucune correction de l'à-coup.	

Remarque

Comportement à la fin du programme pièce et lors d'un Reset de canal

Le comportement de JERKLIM à la fin du programme pièce ou lors d'un Reset de canal dépend du réglage du bit 0 dans le paramètre machine MD32320 $MA_DYN_LIMIT_RESET_MASK.

Exemple

Code de programme	Commentaire
N1000 G0 X0 Y0 F10000 SOFT G64	
N1100 G1 X20 RNDM=5 ACC[X]=20 ACC[Y]=30	
N1200 G1 Y20 JERKLIM[Y]=200	; Dans la direction Y, le chariot de déplacement axial peut accélérer/décélérer au maximum avec 200 % de l'à-coup autorisé pour l'axe.
N1300 G1 X0 JERKLIM[X]=2	; Dans la direction X, le chariot de déplacement axial doit accélérer/décélérer au maximum avec 2 % de l'à-coup autorisé pour l'axe.
N1400 G1 Y0	
M30	

3.14.9.3 Adaptation de la vitesse tangentielle maximale (FLIM)

Avec la fonction "Adaptation de la vitesse tangentielle maximale", la vitesse du déplacement avec interpolation résultant des valeurs de limitation axiales peut être limitée ou réduite dans les sections critiques d'un programme pièce. La valeur de vitesse à laquelle la vitesse tangentielle maximale doit être limitée est programmée via l'adresse FLIM. La valeur programmée reste effective jusqu'au Reset CN suivant ou jusqu'à la fin du programme pièce.

La correction agit en outre sur l'avance limitée avec FLIM. Le commutateur de correction permet ainsi de dépasser modérément FLIM.

Prise d'effet

La fonction prend effet :

- dans les modes de fonctionnement AUTOMATIQUE
- uniquement sur des axes à interpolation
- uniquement pour G94
- pas en vitesse rapide

Remarque

La fonction "Adaptation de la vitesse tangentielle maximale" est effective uniquement en association avec l'avance linéaire G94. Associée à d'autres types d'avance (G93, G931, G95, G96, G97, G971, G972), la fonction ne peut pas être utilisée.

Syntaxe

```
...
FLIM=<valeur>
...
FLIM=-1
...
```

Signification

FLIM	Adresse permettant d'adapter la vitesse tangentielle maximale	
<valeur>	Valeur de vitesse à laquelle la vitesse tangentielle maximale doit être limitée	
	Type de données :	REAL
	Plage de valeurs :	$1{,}0 * 10^{-6}$ ... $1{,}0 * 10^{38}$
	Unité :	mm/min ou inch/min (selon le système de mesure actif)
FLIM=-1	Annule la limitation programmée par FLIM=<valeur>.	

Exemple

Code de programme	**Commentaire**
...	
N1000 G0 X0 Y0 F10000 G64 G710	
N1100 G1 X20 RNDM=5	
N1200 G1 Y20 **FLIM=5000**	; Le chariot de déplacement axial se déplace dans la direction Y à une vitesse max. de 5 m/min.
N1300 G1 X0 Y40	; Le déplacement avec interpolation du chariot de déplacement axial dans les directions X et Y est réalisé à une vitesse max. de 5 m/min.
N1400 G1 Y0 **FLIM=-1**	; Le chariot de déplacement axial se déplace dans la direction Y à une vitesse de 10 m/min.
M30	

3.14.9.4 Adaptation de l'accélération maximale sur trajectoire (PACCLIM)

Avec la fonction "Adaptation de l'accélération maximale sur trajectoire", l'accélération du déplacement avec interpolation résultant des valeurs de limitation axiales peut être réduite dans les sections critiques d'un programme pièce. La valeur d'accélération jusqu'à laquelle l'accélération maximale sur trajectoire doit être réduite est programmée via l'adresse PACCLIM. La valeur programmée reste effective jusqu'au Reset CN suivant ou jusqu'à la fin du programme pièce.

Conditions

Licences

La fonction est une option soumise à licence ("Limitation de l'accélération sur trajectoire" ; numéro d'article : 6FC5800-0xP26-0YB0) qui doit être attribuée via la gestion de licences du matériel.

Prise d'effet

La fonction prend effet :

- dans les modes de fonctionnement AUTOMATIQUE
- uniquement sur des axes à interpolation

Syntaxe

```
...
PACCLIM=<valeur>
...
PACCLIM=-1
...
```

Signification

PACCLIM	Adresse permettant d'adapter l'accélération maximale sur trajectoire	
<valeur>	Valeur d'accélération jusqu'à laquelle l'accélération maximale sur trajectoire doit être réduite	
	Type de données :	REAL
	Plage de valeurs :	$1{,}0 * 10^{-6}$ ... $1{,}0 * 10^{38}$
	Unité :	m/s^2 ou $inch/s^2$ (selon le système de mesure actif)
PACCLIM=-1	Annule la limitation programmée par PACCLIM=<valeur>.	

Remarque

L'effet de PACCLIM se rapproche de celui de la donnée de réglage SD42500 $SC_SD_MAX_PATH_ACCEL (Accélération maximale sur trajectoire). Contrairement à la donnée SD42500, PACCLIM fonctionne toutefois de manière synchrone par blocs.

Remarque

Lors du calcul de la valeur de limitation, la valeur de SD42500 $SC_SD_MAX_PATH_ACCEL est uniquement prise en compte si SD42502 $SC_IS_SD_MAX_PATH_ACCEL est réglé sur "TRUE".

Si PACCLIM et SD42500 $SC_SD_MAX_PATH_ACCEL sont tous deux actifs, la plus petite des deux valeurs de limitation prend effet.

Exemple

Code de programme	Commentaire
...	
N1000 G0 X0 Y0 F10000 G64 G710	
N1100 G1 X20 RNDM=5	
N1200 G1 Y20 **PACCLIM=0.5**	; Le chariot de déplacement axial accélère/décélère dans la direction Y à une vitesse max. de 0,5 m/ss.
N1300 G1 X0 Y40	; Le déplacement avec interpolation du chariot de déplacement axial dans les directions X et Y est accéléré/ralenti à une vitesse max. de 0,5 m/ss.
N1400 G1 Y0	
M30	

3.14.10 Programmation d'une tolérance de contour/orientation (CTOL, OTOL, ATOL)

Les adresses CTOL, OTOL et ATOL permettent d'adapter les tolérances d'usinage paramétrées dans les paramètres machine et les données de réglage pour les fonctions de compresseur, l'arrondissement et le lissage d'orientation dans le programme pièce.

Les valeurs de tolérance programmées s'appliquent jusqu'à ce qu'elles soient reprogrammées ou supprimées par l'affectation d'une valeur négative. Elles sont en outre supprimées à la fin du programme ou lors d'un reset. Après la suppression, les valeurs de tolérance paramétrées redeviennent effectives.

Syntaxe

```
CTOL=<Value>
OTOL=<Value>
ATOL[<Axis>]=<Value>
```

Signification

CTOL :	Adresse pour la programmation de la **tolérance de contour**			
	Domaine d'application :	• toutes les fonctions de compresseur • tous les modes d'arrondissement, à l'exception de G641 et G644		
	Arrêt du prétraitement des blocs :	Non		
	Prise d'effet :	Modale		
	<Value> :	La valeur de la tolérance de contour est exprimée par une cote.		
		Type :	REAL	
		Unité :	inch/mm (en fonction de la cotation actuellement réglée)	
		Plage de valeurs :	≥ 0 :	Valeur de tolérance
			< 0 :	Suppression de la valeur de tolérance programmée ⇒ La valeur de tolérance paramétrée dans le paramètre machine ou les données de réglage reprend effet.
OTOL :	Adresse pour la programmation de la **tolérance d'orientation**			
	Domaine d'application :	• toutes les fonctions de compresseur • au lissage de l'orientation ORISON • tous les modes d'arrondissement, à l'exception de G641, G644 et OSD		
	Arrêt du prétraitement des blocs :	Non		
	Prise d'effet :	Modale		
	<Value> :	La valeur de la tolérance d'orientation est exprimée par une cote angulaire.		
		Type :	REAL	
		Unité :	degré	
		Plage de valeurs :	≥ 0 :	Valeur de tolérance
			< 0 :	Suppression de la valeur de tolérance programmée ⇒ La valeur de tolérance paramétrée dans le paramètre machine ou les données de réglage reprend effet.

ATOL :	Adresse pour la programmation d'une **tolérance spécifique à l'axe**				
	Domaine d'application :	• toutes les fonctions de compresseur • au lissage de l'orientation ORISON • tous les modes d'arrondissement, à l'exception de G641, G644 et OSD			
	Arrêt du prétraitement des blocs :	Non			
	Prise d'effet :	Modale			
	<Axis> :	Nom de l'axe de canal pour lequel la tolérance programmée doit s'appliquer			
	<Value> :	Selon le type d'axe (linéaire ou rotatif), la valeur de la tolérance de l'axe est exprimée par une cote ou une cote angulaire.			
		Type :	REAL		
		Unité :	pour axes linéaires :	inch/mm (en fonction de la cotation actuellement réglée)	
			pour axes rotatifs :	degré	
		Plage de valeurs :	≥ 0 :	Valeur de tolérance	
			< 0 :	Suppression de la valeur de tolérance programmée ⇒ La valeur de tolérance paramétrée dans le paramètre machine ou les données de réglage reprend effet.	

Remarque

Les valeurs de tolérance CTOL et OTOL programmées spécifiques au canal ont une priorité plus élevée que les valeurs de tolérance ATOL programmées spécifiques à l'axe.

Remarque

Frames mis à l'échelle

Les frames mis à l'échelle agissent sur la tolérance programmée de la même manière que sur les positions des axes, c'est-à-dire que la tolérance relative reste identique.

Exemple

Code de programme	Commentaire
COMPCAD G645 G1 F10000	; Activer la fonction de compresseur COMPCAD.
X... Y... Z...	; Ici, les paramètres machine et données de réglage sont actifs.
X... Y... Z...	
X... Y... Z...	
CTOL=0.02	; A partir d'ici, une tolérance de contour de 0,02 mm est active.
X... Y... Z...	
X... Y... Z...	

Code de programme	Commentaire
X... Y... Z...	
ASCALE X0.25 Y0.25 Z0.25	; A partir d'ici, une tolérance de contour de 0,005 mm est active.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
CTOL=-1	; A partir d'ici, les paramètres machine et données de réglage sont de nouveau actifs.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

Variables système

Lecture avec arrêt du prétraitement des blocs

Les tolérances actuellement actives peuvent être lues dans le programme pièce et l'action synchrone par le biais des variables système suivantes :

- $AC_CTOL
  Tolérance de contour spécifique au canal, qui était effective lors du traitement du bloc principal actuel.
  Lorsqu'aucune tolérance de contour n'est effective, $AC_CTOL fournit la racine de la somme des carrés des tolérances appliquées aux axes géométriques.
- $AC_OTOL
  Tolérance d'orientation spécifique au canal, qui était effective lors du traitement du bloc principal actuel.
  Lorsqu'aucune tolérance d'orientation n'est effective, $AC_OTOL fournit la racine de la somme des carrés des tolérances appliquées aux axes d'orientation lorsqu'une transformation d'orientation est active, sinon la valeur "-1".
- $AA_ATOL[<axe>]
  Tolérance de contour spécifique à l'axe, qui était effective lors du traitement du bloc principal actuel.
  Lorsqu'une tolérance de contour est effective, $AA_ATOL[<axe géométrique>] fournit la tolérance de contour divisée par la racine du nombre d'axes géométriques.
  Lorsqu'une tolérance et une transformation d'orientation sont effectives, $AA_ATOL[<axe d'orientation>] fournit la tolérance d'orientation divisée par la racine du nombre d'axes d'orientation.

Remarque

Lorsqu'aucune valeur de tolérance n'a été programmée, les variables $A ne sont pas suffisamment différenciées pour permettre une distinction entre les tolérances de chacune des fonctions.

De tels cas sont susceptibles de se produire lorsque les paramètres machine et les données de réglage définissent des tolérances différentes pour les fonctions de compresseur, l'arrondissement et le lissage de l'orientation. Les variables système fournissent alors la valeur la plus grande des fonctions en cours. Lorsqu'une fonction de compresseur est p. ex. active avec une tolérance d'orientation égale à 0,1° et un lissage de l'orientation ORISON égal à 1°, $AC_OTOL fournit la valeur "1". Si l'on désactive le lissage de l'orientation, $AC_OTOL fournit la valeur "0,1".

Lecture sans arrêt du prétraitement des blocs

Les tolérances actuellement effectives peuvent être lues dans le programme pièce par le biais des variables système suivantes :

- $P_CTOL
  Tolérance de contour spécifique au canal actuellement effective.
- $P_OTOL
  Tolérance d'orientation spécifique au canal actuellement effective.
- $PA_ATOL
  Tolérance de contour spécifique à l'axe actuellement effective.

Autres conditions

Les tolérances programmées avec CTOL, OTOL et ATOL agissent également sur des fonctions qui dépendent indirectement de ces tolérances :

- La limitation de l'erreur cordale lors du calcul de la consigne
- Les fonctions de base du mode formes libres

Les fonctions d'arrondissement suivantes ne sont **pas** influencées par la programmation de CTOL, OTOL et ATOL :

- Arrondissement de l'orientation avec OSD
  OSD n'utilise pas de tolérance mais une distance par rapport à la transition entre blocs.
- Arrondissement avec G644
  G644 ne sert pas pour l'usinage mais pour l'optimisation des changements d'outil et d'autres déplacements sans usinage.
- Arrondissement avec G645
  G645 se comporte presque toujours comme G642 et utilise ainsi les tolérances programmées. La valeur de tolérance issue du paramètre machine PM33120 $MA_PATH_TRANS_POS_TOL n'est utilisée que pour les transitions tangentielles entre blocs avec un saut de courbure, par ex. une transition tangentielle cercle-droite. À ces emplacements, la trajectoire d'arrondissement peut aussi se trouver sur l'extérieur du contour programmé, où de nombreuses applications sont moins tolérantes. De plus, une petite tolérance réglée de manière fixe est en général suffisante pour compenser les variations de courbure, dont le programmateur CN n'a pas à se soucier.

3.14.11 Mode de changement de bloc pour couplage actif (CPBC)

L'instruction CPBC détermine que le critère de changement de bloc doit être rempli pour qu'un changement de bloc puisse avoir lieu dans le programme pièce lorsque le couplage est actif.

Syntaxe

```
CPBC[<axe asservi>] = <critère>
```

Signification

CPBC :	Critère de changement de bloc pour couplage actif	
<axe asservi> :	Descripteur de l'axe asservi	
<critère> :	Critère de changement de bloc	
	Type :	STRING
	Valeur	**Signification :** Le changement de bloc s'effectue
	"NOC"	indépendamment de l'état du couplage
	"IPOSTOP"	à la valeur de consigne du synchronisme
	"COARSE"	lors du synchronisme "grossier" côté valeur réelle
	"FINE"	lors du synchronisme "fin" côté valeur réelle

Exemple

Code de programme

```
; le changement de bloc s'effectue pour :
;  - couplage avec l'axe asservi X2 == actif
;  - synchronisme côté consigne == actif
CPBC[X2]="IPOSTOP"
```

3.15 Fonctions d'axe

3.15.1 Permutation d'axe, permutation de broche (RELEASE, GET, GETD)

Un ou plusieurs axes ou broches ne peuvent être interpolés que dans un canal. Si l'on prévoit d'utiliser un axe alternativement dans deux canaux différents (par exemple dans le cas d'un changeur de palettes), il faut d'abord le libérer dans le canal actuel, puis le prendre en charge dans l'autre canal. L'axe est permuté entre les canaux.

Extensions de permutation d'axe

Un axe/broche peut être remplacé avec un arrêt du prétraitement et une synchronisation entre le prétraitement et l'exécution des blocs ou encore sans arrêt du prétraitement. Par ailleurs, une permutation d'axe est également possible via :

- une rotation conteneur d'axes AXCTSWE ou AXCTWED via un `GET/GETD` implicite.
- Frame avec rotation lorsque cet axe est combiné avec d'autres axes.
- Actions synchrones, voir Actions synchrones au déplacement, "Permutation d'axe `RELEASE`, `GET`".

Constructeur de machine

Veuillez tenir compte des indications du constructeur de machine. Des paramètres machines configurables doivent définir de manière univoque un axe pour la permutation d'axe dans tous les canaux et des paramètres machines peuvent aussi modifier le comportement de la permutation d'axe.

Syntaxe

```
RELEASE (nom d'axe, nom d'axe, ...) ou RELEASE (S1)
GET (nom d'axe, nom d'axe, ...) ou GET (S2)
GETD (nom d'axe, nom d'axe, ...) ou GETD (S3)
```

Avec GETD (GET Directly), un axe est extrait directement d'un autre canal. Cela signifie que, pour ce GETD, il n'est pas nécessaire de programmer un RELEASE approprié dans un autre canal. Cela signifie aussi que désormais, une autre communication de canal doit être mise en place (par exemple marques wait).

Signification

`RELEASE (nom d'axe, nom d'axe, …)` :	Déblocage de/des axe(s)
`GET (nom d'axe, nom d'axe, …)` :	Prise en charge de/des axe(s)
`GETD (nom d'axe, nom d'axe, …)` :	Prise en charge directe de/des axe(s)
`nom d'axe` :	Affectation des axes dans le système : AX1, AX2, ... ou indication du nom des axes de la machine
`RELEASE(S1)` :	Libération des broches S1, S2, ...
`GET(S2)` :	Prise en charge des broches S1, S2, ...
`GETD(S3)` :	Prise en charge directe des broches S1, S2, ...

Demande GET sans arrêt du prétraitement

Lorsque, après une demande GET **sans** arrêt du prétraitement, l'axe est de nouveau débloqué avec `RELEASE(Axe)` ou `WAITP(Axe)`, le `GET` qui suit entraîne un `GET` **avec** arrêt du prétraitement.

⚠ PRUDENCE

Affectation des axes modifiée

Un axe ou une broche pris(e) en charge avec GET reste assigné à ce canal, même après un RESET de touche ou de programme.

En cas de redémarrage du programme, l'affectation des axes permutés ou des broches permutées doit être programmée si les axes ou les broches doivent être utilisés dans leurs canaux d'origine.

A la mise sous tension de la CN, les axes et les broches sont affectés aux canaux spécifiés dans les paramètres machine.

Exemples

Exemple 1 : permuter des axes entre deux canaux

Sur 6 axes sont utilisés pour le traitement dans le canal 1 : les 1er, 2e, 3e et 4e axes.
Les 5e et 6e axes sont utilisés dans le canal 2 pour le changement de pièce.

L'axe 2 doit pouvoir être permuté entre les deux canaux et, lors de la mise sous tension de la CN, affecté au canal 1.

Programme "MAIN" dans le canal 1 :

Code de programme	**Commentaire**
`INIT (2,"PERMUT2")`	`; Sélection du programme PERMUT2 dans le canal 2.`
`N… START (2)`	`; Démarrage du programme dans le canal 2.`
`N… GET (AX2)`	`; Prise en charge de l'axe AX2.`
`...`	
`N… RELEASE (AX2)`	`; Libération de l'axe AX2.`
`N… WAITM (1,1,2)`	`; Attente du repère WAIT dans les canaux 1 et 2 pour la synchronisation dans les deux canaux.`
`...`	`; Poursuite de l'exécution après la permutation d'axe.`
`N… M30`	

Programme "PERMUT2" dans le canal 2 :

Programmation	**Commentaire**
`N… RELEASE (AX2)`	
`N160 WAITM (1,1,2)`	`; Attente du repère WAIT dans les canaux 1 et 2 pour la synchronisation dans les deux canaux.`
`N150 GET (AX2)`	`; Prise en charge de l'axe AX2.`

Programmation	Commentaire
...	; Poursuite de l'exécution après la permutation d'axe.
N… M30	

Exemple 2 : permuter des axes sans synchronisation

Lorsque l'axe ne doit pas être synchronisé, GET ne génère pas d'arrêt du prétraitement.

Programmation	Commentaire
N01 G0 X0	
N02 RELEASE(AX5)	
N03 G64 X10	
N04 X20	
N05 GET(AX5)	; Si aucune synchronisation n'est requise, ce bloc ne sera pas exécutable.
N06 G01 F5000	; Bloc non exécutable.
N07 X20	; Bloc non exécutable, car position X comme dans N04.
N08 X30	; Premier bloc exécutable après N05.
...	

Exemple 3 : activer une permutation d'axes sans arrêt du prétraitement des blocs

Prérequis : la permutation d'axe sans arrêt du prétraitement doit être configurée via un paramètre machine.

Programmation	Commentaire
N010 M4 S100	
N011 G4 F2	
N020 M5	
N021 SPOS=0	
N022 POS[B]=1	
N023 WAITP(B)	; L'axe B devient un axe banalisé.
N030 X1 F10	
N031 X100 F500	
N032 X200	
N040 M3 S500	; L'axe ne déclenche pas d'arrêt du prétraitement / de REORG.
N041 G4 F2	
N050 M5	
N099 M30	

Lorsque la broche ou l'axe B sont déplacés directement après le bloc N023 comme **axe AP** par exemple sur 180 degrés puis à nouveau sur 1 degré, alors cet axe redevient un axe banalisé et ne déclenche pas d'arrêt du prétraitement dans le bloc N40.

Plus d'informations

Conditions nécessaires à la permutation d'axe

- L'axe doit être défini avec les paramètres machine dans tous les canaux qui utilisent cet axe.
- Le canal auquel l'axe est affecté lors de la mise sous tension doit être défini à l'aide du paramètre machine spécifique à un **axe**.

Description

Libération de l'axe : RELEASE

Lors de la libération de l'axe, respecter les règles suivantes :

1. L'axe ne doit participer à aucune transformation.
2. En cas de couplage d'axes (positionnement tangentiel), tous les axes du groupe doivent être libérés.
3. Il est impossible de permuter un axe de positionnement concurrent dans cet état.
4. Dans le cas d'un axe maître Gantry (axe de portique), tous les axes asservis sont aussi permutés.
5. Dans le cas d'un couplage d'axes (déplacement conjugués, couplage pilote, réducteur électronique), seul l'axe pilote du groupe peut être libéré.

Prise en charge de l'axe : GET

La permutation d'axe devient effective avec cette instruction. Le canal dans lequel l'instruction a été programmée prend entièrement en charge l'axe.

Effets de GET :

Permutation d'axe avec synchronisation :

Un axe doit toujours être synchronisé s'il a été affecté entre-temps à un canal différent ou à l'AP, et qu'il n'y a pas eu synchronisation avant GET par "WAITP", G74 ou effacement de la distance restant à parcourir.

- Un arrêt de déroulement a lieu (comme pour STOPRE).
- L'exécution du programme est interrompue jusqu'à ce que la permutation d'axe soit terminée.

"GET" automatique

Si un axe est disponible par principe dans le canal, mais rendu momentanément indisponible comme axe de canal, alors un GET est automatiquement exécuté. Si le ou les axes sont déjà synchronisés, aucun arrêt du prétraitement des blocs n'a lieu.

Paramétrage du comportement de permutation d'axe

Le moment de fin de prise en compte des axes peut être paramétré de différentes façons :

- La permutation d'axe entre deux canaux s'effectue automatiquement même si l'axe a banalisé avec WAITP (comportement tel que jusqu'à présent)
- Lors de la demande d'une rotation du conteneur d'axes, tous les axes assignables au canal à traiter du conteneur d'axes sont prélevés dans le canal à l'aide de GET ou de GETD implicites. Une permutation d'axe n'est ensuite à nouveau autorisée qu'une fois la rotation du conteneur d'axe terminée.
- Lorsqu'un bloc intermédiaire a été intercalé, une vérification est effectuée lors de l'exécution du bloc pour déterminer si une réorganisation est nécessaire ou non. Une réorganisation est nécessaire uniquement si les états de l'axe figurant dans ce bloc ne correspondent **pas** aux états actuels de l'axe.
- À la place d'un bloc GET avec arrêt du prétraitement des blocs et synchronisation entre le prétraitement et l'exécution des blocs, une permutation d'axe peut aussi avoir lieu sans arrêt du prétraitement. Un seul bloc intermédiaire est alors généré avec la demande GET. Pendant l'exécution de ce bloc durant l'exécution du programme, ce dernier vérifie si les états de l'axe dans le bloc correspondent bien aux états d'axe actuels.

3.15.2 Transmettre un axe à un autre canal (AXTOCHAN)

Avec l'instruction de langage `AXTOCHAN`, un axe peut être demandé pour transmettre cet axe à un autre canal. L'axe peut aussi bien être amené au canal correspondant par le programme pièce CN que par une action synchrone.

Syntaxe

```
AXTOCHAN(nom d'axe,numéro de canal[,nom d'axe,numéro de canal[,...]])
```

Signification

Elément	Description
`AXTOCHAN` :	Demande d'axe pour un canal donné
`nom d'axe` :	Affectation des axes dans le système : X, Y, ... ou indication des noms d'axe machine concernés. Le canal à exécuter ne doit pas être son canal propre et ce ne doit pas être non plus le canal qui possède actuellement le droit d'interpolation pour l'axe
`numéro de canal` :	Numéro du canal devant être assigné à l'axe

Remarque

Axe de positionnement concurrent et axe exclusivement contrôlé par l'AP

Un axe AP ne peut pas changer de canal en tant qu'axe de positionnement concurrent. Un axe exclusivement contrôlé par l'AP ne peut pas être affecté au programme CN.

Informations complémentaires : Description fonctionnelle Axes et broches

Exemple

AXTOCHAN dans le programme CN

Les axes X et Y sont connus dans le 1er et dans le 2e canal. Actuellement, c'est le canal 1 qui détient le droit d'interpolation et le programme suivant démarre dans le canal 1:

```
Code de programme                Commentaire
N110 AXTOCHAN(Y,2)               ; Translater l'axe Y dans le 2e canal.
N111 M0
N120 AXTOCHAN(Y,1)               ; Aller reprendre l'axe Y (banalisé).
N121 M0
N130 AXTOCHAN(Y,2,X,2)           ; Translater l'axe Y et l'axe X dans le 2e canal (axes
                                 banalisés).
N131 M0
N140 AXTOCHAN(Y,2)               ; Translater l'axe Y dans le 2e canal (programme CN).
N141 M0
```

Informations complémentaires

AXTOCHAN dans le programme CN

Un GET est alors exécuté uniquement en cas de demande de l'axe pour le programme CN dans un canal séparé et la modification d'état réelle est attendue. Lorsque l'axe est demandé pour un autre canal ou s'il doit devenir un axe banalisé dans un canal séparé, alors seulement la demande est générée.

AXTOCHAN à partir d'une action synchrone

Lorsqu'un axe est demandé pour un canal séparé alors AXTOCHAN est représenté sur un GET à partir d'une action synchrone. Dans ce cas, l'axe devient axe banalisé à la première demande pour le canal propre. À la deuxième demande, l'axe est affecté au programme CN d'une manière analogue à la demande GET dans le programme CN. En ce qui concerne la demande GET à partir d'une action synchrone, se reporter au chapitre "Actions synchrones au déplacement".

3.15.3 Fonctions d'axe (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)

"AXNAME" est utilisé par exemple pour la création de cycles à caractère général lorsque les noms des axes ne sont pas connus.

"AX" est utilisé pour la programmation indirecte d'axes géométriques et d'axes synchrones. Le descripteur d'axe est alors enregistré dans une variable de type AXIS ou est fourni par une instruction telle que "AXNAME" ou "SPI".

"SPI" est utilisé lorsque des fonctions d'axe sont programmées pour une broche (par exemple une broche synchrone).

"AXTOSPI" est utilisé pour convertir un descripteur d'axe en indice de broche (fonction inverse de "SPI").

"AXSTRING" est utilisé pour convertir un descripteur d'axe (type de données AXIS) en chaîne de caractères (fonction inverse de "AXNAME").

"ISAXIS" est utilisé dans des cycles à caractère général, pour vérifier l'existence d'un axe géométrique bien défini et pour éviter qu'un appel consécutif de $P_AXNX soit interrompu du fait d'une erreur.

"MODAXVAL" est utilisé pour déterminer la position modulo d'axes rotatifs modulo.

Syntaxe

```
AXNAME("chaîne de caractères")
AX[AXNAME("chaîne de caractères")]
SPI(n)

AXTOSPI(A) ou AXTOSPI(B) ou AXTOSPI(C)
AXSTRING( SPI(n) )
ISAXIS(<numéro d'axe géométrique>)
<Position modulo>=MODAXVAL(<axe>,<position d'axe>)
```

Signification

AXNAME :	Convertit la chaîne de caractères d'entrée en un descripteur d'axe ; la chaîne de caractères d'entrée doit comporter un nom d'axe valide.
AX :	Descripteur d'axe variable
SPI :	Conversion du numéro de broche indiqué en descripteur d'axe ; le paramètre de transfert doit être un numéro de broche valide.
n :	numéro de broche
AXTOSPI :	Convertit un descripteur d'axe en un indice de broche de type Integer. "AXTOSPI" correspond à la fonction d'inversion pour "SPI".
X, Y, Z :	Descripteur d'axe de type AXIS comme variable ou comme constante
AXSTRING :	La chaîne de caractères avec numéro de broche attribué.
ISAXIS :	vérifie si l'axe géométrique indiqué existe
MODAXVAL :	Détermination de la position modulo des axes rotatifs modulo, qui correspond au reste modulo par rapport à la zone modulo paramétrée (0 à 360 degrés en réglage par défaut ; les paramètres machine PM30340 MODULO_RANGE_START et PM30330 $MA_MODULO_RANGE permettent de modifier le début et la taille de la zone modulo).

Remarque

Extensions SPI

La fonction d'axe SPI(n) est également utilisable pour la lecture et l'écriture de composantes de frame. Il est ainsi possible, par exemple, d'écrire des frames avec la syntaxe `$P_PFRAME[SPI(1),TR]=2.22`.

La programmation supplémentaire de positions d'axe avec l'adresse `AX[SPI(1)]=<position d'axe>` permet de déplacer un axe, à condition que la broche soit en mode de positionnement ou en mode axe.

Exemples

Exemple 1 : AXNAME, AX, ISAXIS

Code de programme	Commentaire
OVRA[AXNAME("axe transversal")]=10	; Correction d'axe transversal
AX[AXNAME("axe transversal")]=50.2	; Position de fin de l'axe transversal
OVRA[S1]=70	; Correction de la broche 1
AX[SPI(1)]=180	; Position de fin de la broche 1
IF ISAXIS(1)== FALSE GOTOF SUITE	; Abscisse existante ?
AX[$P_AXN1]=100	; Déplacement de l'abscisse
SUITE :	

Exemple 2 : AXSTRING

En cas de programmation avec AXSTRING[SPI(n)], ce n'est plus l'indice de l'axe auquel la broche est affectée qui est sorti en tant que numéro de broche, mais la chaîne de caractères "Sn".

Code de programme	Commentaire
AXSTRING[SPI(2)]	; Sortie de la chaîne de caractères "S2".

Exemple 3 : MODAXVAL

La position modulo de l'axe rotatif modulo A doit être déterminée.

Le calcul part de la position d'axe 372.55.

La zone modulo paramétrée s'étend de 0 à 360 degrés :

PM30340 MODULO_RANGE_START = 0

PM30330 $MA_MODULO_RANGE = 360

Code de programme	Commentaire
R10=MODAXVAL(A,372.55)	; Position modulo calculée R10 = 12.55.

Exemple 4 : MODAXVAL

Si le descripteur d'axe programmé ne se rapporte pas à un axe rotatif modulo, la valeur à convertir (<position d'axe>) est retournée telle quelle.

Code de programme	Commentaire
R11=MODAXVAL(X,372.55)	; X est l'axe linéaire ; R11 = 372.55.

3.15.4 Axes géométriques permutables (GEOAX)

La fonction "Axes géométriques permutables" permet de remplacer les axes géométriques configurés dans les paramètres machine par d'autres axes de canal.

Syntaxe

```
GEOAX(<n>,<axe de canal>,<n>,<axe de canal>,<n>,<axe de canal>)
GEOAX()
```

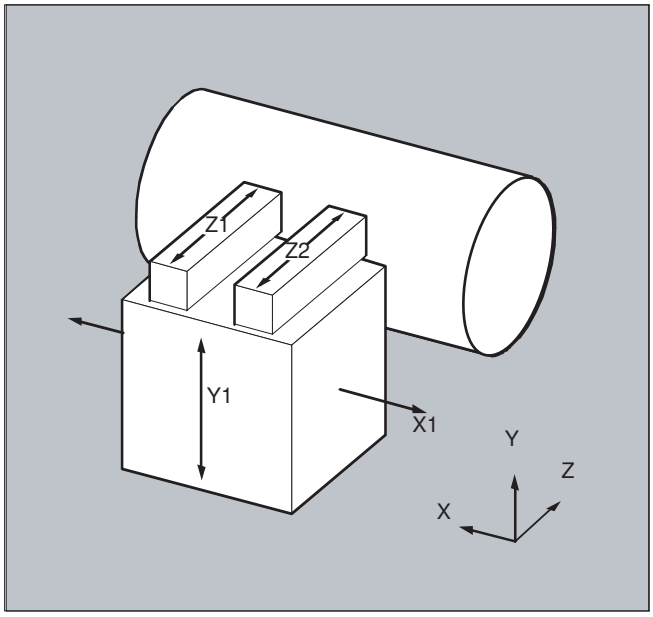
Signification

`GEOAX(...)`	Fonction de permutation des axes géométriques. **Remarque :** `GEOAX()` sans indication de paramètre réactive la configuration de base des axes géométriques définie dans les paramètres machine.
`<n>`	Numéro de l'axe géométrique qui sera remplacé par l'axe de canal indiqué. Plage de valeurs : 0, 1, 2, 3 **Remarque :** 0: l'axe de canal indiqué est supprimé du groupe d'axes géométriques sans remplacement 1: 1. axe géométrique ≙ axe de coordonnées X (abscisse) du SCP 2: 2. axe géométrique ≙ axe de coordonnées Y (ordonnée) du SCP 3: 3. axe géométrique ≙ axe de coordonnées Z (cote) du SCP
`<Axe de canal>`	Nom de l'axe de canal qui doit être intégré dans le groupe d'axes géométriques.

Exemples

Exemple 1 : deux axes permutés de façon alternée comme axe géométrique

Un chariot porte-outil peut être déplacé par le biais des axes de canal X1, Y1, Z1, Z2 :



Les axes géométriques sont configurés de sorte que d'abord, après la mise sous tension, Z1 soit actif en tant que 3e axe géométrique avec le nom d'axe géométrique "Z" et qu'il forme le groupe d'axes géométriques avec X1 et Y1.

Dans le programme pièce, les axes Z1 et Z2 doivent ensuite intervenir de façon alternée comme axe géométrique Z :

```
Code de programme                Commentaire
...
```

Code de programme	Commentaire
N100 GEOAX (3,Z2)	; L'axe de canal Z2 intervient en tant que 3e axe géométrique (Z).
N110 G1 ...	
N120 GEOAX(3,Z1)	; L'axe de canal Z1 intervient en tant que 3e axe géométrique (Z).
...	

Exemple 2 : permutation des axes géométriques en présence de 6 axes de canal

Une machine comporte 6 axes de canal désignés par XX, YY, ZZ, U ,V ,W.

Le préréglage de la configuration des axes géométriques effectué dans les paramètres machine est le suivant :

axe de canal XX = 1er axe géométrique (axe X)

axe de canal YY = 2e axe géométrique (axe Y)

axe de canal ZZ = 3e axe géométrique (axe Z)

Code de programme	Commentaire
N10 GEOAX()	; La configuration de base des axes géométriques est activée.
N20 G0 X0 Y0 Z0 U0 V0 W0	; Tous les axes accostent la position 0 en vitesse rapide.
N30 GEOAX(1,U,2,V,3,W)	; L'axe de canal U devient le premier (X), V devient le deuxième (Y)
	; et W devient le troisième axe géométrique (Z).
N40 GEOAX(1,XX,3,ZZ)	; L'axe de canal XX devient le premier (X), ZZ devient le troisième
	; axe géométrique (Z). L'axe de canal V reste le deuxième
	: axe géométrique (Y).
N50 G17 G2 X20 I10 F1000	; Cercle complet dans le plan X/Y. Déplacement des
	; axes de canal XX et V.
N60 GEOAX(2,W)	; L'axe de canal W devient le deuxième axe géométrique (Y).
N80 G17 G2 X20 I10 F1000	; Cercle complet dans le plan X/Y. Déplacement des
	; axes de canal XX et W.
N90 GEOAX()	; Réinitialisation à l'état de base.
N100 GEOAX(1,U,2,V,3,W)	; L'axe de canal U devient le premier (X), V devient le deuxième
	; (Y) et W devient le troisième axe géométrique (Z).
N110 G1 X10 Y10 Z10 XX=25	; Les axes de canal U, V, W se déplacent respectivement à la
	; position 10. L'axe XX, en tant qu'axe supplémentaire, se déplace à la position 25.
N120 GEOAX(0,V)	; V est supprimé du groupe d'axes géométriques.
	; U et W restent premier (X) et troisième
	; axe géométrique (Z).
	; Le deuxième axe géométrique (Y) reste non affecté.

Code de programme	Commentaire
N130 GEOAX(1,U,2,V,3,W)	; L'axe de canal U reste premier (X), V devient ; le deuxième (Y) et W reste le troisième axe géométrique (Z).
N140 GEOAX(3,V)	; V devient le troisième axe géométrique (Z), écrasant W ; qui est ainsi supprimé du groupe d'axes ; géométriques. Le deuxième axe géométrique (Y) ; n'est toujours pas affecté.

Paramètres machine

Configuration d'axes

Affectation des axes géométriques, des axes supplémentaires et des axes machine aux axes de canal :

- PM10000 $MN_AXCONF_MACHAX_NAME_TAB
- PM20050 $MC_AXCONF_GEOAX_ASIGN_TAB
- PM20060 $MC_AXCONF_GEOAX_NAME_TAB
- PM20070 $MC_AXCONF_MACHAX_USED
- PM20080 $MC_AXCONF_CHANAX_NAME_TAB
- PM35000 $MA_SPIND_ASSIGN_TO_MACHAX

Comportement au reset

Comportement au reset des affectations d'axes géométriques modifiées :

- PM20110 $MC_RESET_MODE_MASK, bit 12
- PM20118 $MC_GEOAX_CHANGE_RESET

Comportement de départ CN

- PM20112 $MC_START_MODE_MASK, bit 12

Notification au programme utilisateur AP

Option de paramétrage de l'instruction M qui sera transmise à l'interface CN/AP en présence d'une permutation d'axes géométriques :

- PM22532 $MC_GEOAX_CHANGE_M_CODE

Autres conditions à prendre en compte

Aucune permutation d'axes géométriques

- Quand l'une des fonctions suivantes est active, aucune permutation d'axes géométriques n'est possible :
  - Transformation
  - Interpolation de type spline
  - Correction du rayon d'outil
  - Correction fine d'outil
- L'axe géométrique et un autre axe de canal portent le même nom.
- L'un des axes concernés par la permutation d'axes géométriques est impliqué dans une action qui dure au-delà des limites de bloc. Par exemple : axe de positionnement pour l'ensemble des blocs ou axe asservi d'un couplage d'axes.

Axes rotatifs

Les axes rotatifs ne peuvent pas être définis comme axes géométriques.

Etat d'un axe après le remplacement

Un axe remplacé par commutation dans le groupe d'axes géométriques est programmable en tant qu'axe supplémentaire au moyen de son nom d'axe de canal après la commutation.

Frames, zones de protection, limitations de la zone de travail

La permutation des axes géométriques supprime tous les frames, les zones de protection et les limitations de la zone de travail.

Coordonnées polaires

Une permutation des axes géométriques avec `GEOAX` met les coordonnées polaires modales à la valeur 0 de manière analogue à un changement de plan avec `G17-G19`.

DRF, DO

Un éventuel décalage par manivelle (DRF) ou un décalage d'origine externe (DO) reste actif après la permutation.

Configuration de base des axes géométriques

L'instruction `GEOAX()` appelle la configuration de base du groupe d'axes géométriques.

Après un POWER ON et en cas de commutation en mode de fonctionnement "Prise de référence", on retourne automatiquement à la configuration de base.

Correction de longueur d'outil

Une correction de longueur d'outil active reste active après la permutation. Pour les axes géométriques qui viennent d'être intégrés ou dont la position a été permutée, on considère cependant que le déplacement correspondant n'a pas encore été exécuté. Pour la première instruction de déplacement de ces axes géométriques, le déplacement résultant est donc égal à la somme de la correction de longueur d'outil et du déplacement programmé.

Les axes géométriques qui gardent leur position dans le groupe d'axes après la permutation conservent également leur état relatif à la correction de longueur d'outil.

Configuration d'axe géométrique en cas de transformation active

- La configuration d'axe géométrique définie dans les paramètres machine pour une transformation active ne peut pas être modifiée avec la fonction "Axes géométriques permutables".
- En présence d'une configuration d'axe géométrique différente pour une transformation, différents blocs de données doivent être définis dans les paramètres machine de transformation.
- L'activation d'une transformation supprime une configuration d'axe géométrique modifiée avec `GEOAX`.
- En ce qui concerne les axes géométriques, les paramétrages d'axes géométriques spécifiques aux transformations pour les transformations actives ont la priorité sur les paramétrages concernant la permutation d'axes géométriques.
  Exemple : une transformation est active. D'après les paramètres machine, la transformation doit être conservée après un reset de canal. Toutefois, après un reset de canal, la configuration de base des axes géométriques doit également être rétablie. La configuration d'axe géométrique qui a été définie pour la transformation est conservée.
- Le réglage de base de la configuration d'axe géométrique reprend effet lors de la désactivation d'une transformation.

Mode JOG, fonction machine REF

La configuration d'axe géométrique définie dans les paramètres machine prend effet lors du passage en mode JOG, fonction machine REF (accostage du point de référence).

3.15.5 Conteneur d'axes (AXCTSWE, AXCTSWED, AXCTSWEC)

Les instructions "AXCTSWE" ou "AXCTSWED" permettent de débloquer la rotation des conteneurs d'axes spécifiés.

L'instruction "AXCTSWEC" permet d'annuler un déblocage de rotation du conteneur d'axes ayant déjà été spécifié.

Syntaxe

```
AXCTSWE(<ID>)
AXCTSWED(<ID>)
AXCTSWEC(<ID>)
```

Signification

AXCTSWE :	Déblocage de la rotation du conteneur d'axes L'exécution du programme ne s'arrête pas avec "AXCTSWE". La rotation se produit dès que le déblocage a été ordonné à tous les canaux impliqués dans le conteneur d'axes.	
AXCTSWED :	Déblocage de la rotation du conteneur d'axes sans prise en compte des autres canaux impliqués. **Remarque** • Variante d'instruction destinée à la simplification de la mise en service du programme pièce ou de l'action synchrone. • Le comportement par rapport aux autres canaux impliqués dans le conteneur d'axes peut être spécifié à l'aide du paramètre machine suivant : PM12760 $MN_ AXCT_FUNCTION_MASK, bit 0	
AXCTSWEC :	Annulation du déblocage de la rotation du conteneur d'axes **Remarque** Le déblocage de la rotation du conteneur d'axes ne peut être annulé que si la rotation n'a pas encore commencé : $AN_AXCTSWA[<conteneur d'axes>] == 0 Au sujet de la variable système, voir "Conteneur d'axes (AXCTSWE, AXCTSWED, AXCTSWEC) (Page 912)"	
<ID> :	Descripteur du conteneur d'axes ou d'un axe de conteneur :	
	CT<numéro> :	Descripteur par défaut d'un axe de conteneur : PM12750 $MN_AXCT_NAME_TAB Exemple : "CT1"
	<conteneur> :	Descripteur spécifique à l'utilisateur d'un conteneur d'axes : PM12750 $MN_AXCT_NAME_TAB Exemple : "CONTENEUR_1
	<axe> :	Descripteur d'un axe de conteneur connu dans le canal

Remarque

Pas

Le pas de rotation d'un conteneur d'axes est réglé par la donnée de réglage suivante :

SD41700 $SN_AXCT_SWWIDTH

Informations complémentaires

Diagnostic

La lecture de l'état actuel d'un conteneur d'axe peut s'effectuer avec la variable système suivante :

Variable système	Type	Description
$AC_AXCTSWA[<nom>]	BOOL	Etat spécifique au canal du conteneur d'axes
$AN_AXCTSWA[<conteneur d'axes>]	BOOL	Etat spécifique à la NCU du conteneur d'axes

Variable système	Type	Description
$AN_AXCTSWE[<conteneur d'axes>]	INT	Etat de la rotation du conteneur d'axes spécifique à l'emplacement La variable système indique **bit par bit** l'état des emplacements du conteneur d'axes. Chaque bit correspond à un emplacement.
$AN_AXCTAS[<conteneur d'axes>]	INT	Nombre d'emplacements (slots) autour desquels le conteneur d'axes a été permuté.

Rotation du conteneur d'axes avec GET/GETD implicite

Le paramètre machine ci-après permet de définir que, avec l'instruction AXCTSWE, tous les axes de conteneur du canal peuvent être récupérées dans le canal au moyen d'un "GET/GETD" implicite. Une permutation d'axes ne redevient possible qu'après une rotation de conteneur réussie.

PM10722 $MN_AXCHANGE_MASK, bit 1 = 1

Remarque

La rotation d'un conteneur d'axes avec "GET / GETD" implicite n'est **pas** exécutée pour un axe à l'état "axe d'exécution de bloc" (par ex. axe AP), car l'axe devrait quitter cet état pour permettre la rotation.

3.15.6 Attente de la position d'axe définitive (WAITENC)

L'instruction de langage "WAITENC" permet d'attendre dans le programme CN que les positions d'axes synchronisées ou restaurées pour les axes configurés avec PM34800 $MA_WAIT_ENC_VALID = 1 soient disponibles.

Une interruption peut avoir lieu en mode d'attente, par ex. par le démarrage d'un ASUP ou par le changement de mode de fonctionnement en JOG. Le mode d'attente est éventuellement repris avec la poursuite du programme.

Remarque

Le mode d'attente est affiché dans l'interface utilisateur au moyen de l'état d'arrêt "Attendre système de mesure".

Syntaxe

"WAITENC" peut être programmé dans la section d'un programme CN quelconque.

La programmation doit être réalisée dans un bloc spécifique :

```
...
WAITENC
...
```

Exemple

"WAITENC" est par ex. utilisé dans le programme d'application commandé par événement .../_N_CMA_DIR/_N_PROG_EVENT_SPF ainsi que le montre l'exemple d'application suivant.

Exemple d'application : retrait d'outil après POWER OFF avec transformation d'orientation

Un usinage avec orientation de l'outil a été interrompu par une coupure de courant.
Le programme d'application commandé par événement .../_N_CMA_DIR/_N_PROG_EVENT_SPF est appelé lors du démarrage s'en suivant.

Dans le programme d'application commandé par événement, "WAITENC" permet d'attendre des positions d'axe synchronisées ou restaurées afin de pouvoir calculer par la suite une grille pour orienter le SCP dans la direction de l'outil.

Code de programme	Commentaire
...	
IF $P_PROG_EVENT == 4	; Démarrage.
IF $P_TRAFO <> 0	; Transformation a été activée.
WAITENC	; Attente des positions valables des axes d'orientation.
TOROTZ	; Rotation de l'axe Z du SCP en direction de l'axe d'outil.
ENDIF	
M17	
ENDIF	
...	

En mode JOG, l'outil peut par la suite, par un déplacement de retrait, être dégagé dans la direction de l'axe d'outil.

3.15.7 Commutation de jeu de paramètres programmable (SCPARA)

L'instruction SCPARA permet de demander pour un axe la commutation à un jeu de paramètres défini.

Remarque

Pas de commutation de jeu de paramètres pendant l'usinage du filetage

Lors d'une opération de filetage G33 et de taraudage G331 / G332, le jeu de paramètres est sélectionné par la commande et ne peut être modifié.

Commutation de jeu de paramètres bloquée

Une commutation de jeu de paramètres peut également être demandée via l'interface CN/AP. Pour éviter tout conflit de commutation, il est possible de bloquer la commutation de jeu de paramètres de la CN via l'interface CN/AP (SCPARA) :

DB31, ... DBX9.3 (spécification de jeu de paramètres bloquée par CN)

Remarque

Si une commutation de jeu de paramètres est demandée par SCPARA alors que la commutation de jeu de paramètres est bloquée par l'interface CN/AP, la commutation sera refusée sans génération de message d'erreur.

Syntaxe

```
SCPARA[<axe>]=<valeur>
```

Signification

SCPARA :	Instruction : commuter le jeu de paramètres	
<axe> :	Descripteur d'axe (axe de canal)	
	Type :	AXIS
<valeur> :	Numéro de jeu de paramètres : 1, 2, 3, ... numéro de jeu de paramètres max.	

Exemple

Code de programme	Commentaire
...	
N110 SCPARA[X] = 3	; Sélection : Axe X, 3e jeu de paramètres
...	

Informations complémentaires

Déblocage de la commutation de jeu de paramètres

La commutation de jeu de paramètres doit être débloquée de manière explicite :

MD35590 $MA_PARAMSET_CHANGE_ENABLE[<axe>]

Lecture du numéro de jeu de paramètres

La variable système $AA_SCPAR permet de lire le numéro du jeu de paramètres sélectionné (jeu de paramètres prescrit).

3.15.8 Activation/désactivation de l'adaptation (CADAPTON, CADAPTOF)

Les procédures CADAPTON() et CADAPTOF() prédéfinies permettent d'activer, d'actualiser et de désactiver les adaptations prédéfinies pour l'adaptation des paramètres de dynamique ou de régulation à partir du programme pièce. Elles sont également utilisées dans le CYCLE782 (Page 1129).

Syntaxe

```
...
CADAPTON(<Result>,<Axis>,<InVar>[,<InValue>])
...
CADAPTOF(<Result>,<Axis>,<InVar>)
...
```

Signification

`CADAPTON()` :	Activer la relation d'adaptation		
`CADAPTOF()` :	Désactiver la relation d'adaptation **Remarque :** Les relations d'adaptation effectives en permanence (PM16501 = 1) ignorent `CADAPTOF()`, une valeur d'entrée éventuellement programmée précédemment (`<InVal>`) reste effective.		
`<Result>` :	Variable de résultat : Valeur de retour pour l'état (paramètre Call by reference)		
	Type de données :	INT	
	Valeur :	0	Aucune erreur
		1	Aucune table d'adaptation valide paramétrée
		2	Paramètre `<Axis>` non valable
		3	Paramètre `<InVar>` non valable
		4	Réservé
		5	Paramètre `<InVal>` non valable
`<Axis>` :	Nom d'axe machine de l'axe d'entrée de la relation d'adaptation		
	Type de données :	AXIS	
	Plage de valeurs :	Noms d'axe machine définis dans le canal	
	Remarque : Ce paramètre permet d'appeler les adaptations qui ont introduit une valeur correspondant au paramètre `<Axis>` dans le MD16504 $MN_CADAPT_INPUT_AX. Les autres paramètres `<InVar>` et `<InVal>` sont affectés à cet axe.		
`<InVar>` :	Grandeur d'entrée de la relation d'adaptation		
	Type de données :	INT	
	Valeur :	1	Inertie de l'axe
		2	Position de l'axe
		3	Vitesse de l'axe

<InVal> :	Valeur d'entrée de la relation d'adaptation Paramètre optionnel pour l'adaptation intelligente de la charge (<InVar>=1).		
	Type de données :	REAL	
	Valeur :	> 0	Inertie actuelle
		= 0	La charge est indéfinie ou inconnue. Les adaptations affectées fournissent dans ce cas la valeur de remplacement 1.0 en tant que grandeur de sortie.
		< 0	L'appel se termine avec <Status> = 5 (paramètre <InVal> non valable). La dernière valeur d'entrée activée reste active.
	Remarque : Sans programmation d'une valeur d'entrée, la dernière valeur programmée ou, après un démarrage de la commande, la valeur par défaut (= 0) prend effet.		

Exemple

Code de programme	**Commentaire**
`DEF INT RESULT`	`; Définition des variables de résultat`
`...`	
`CADAPTON(RESULT,MX1,2)`	`; Activer l'adaptation`
`IF RESULT <> 0`	`; Exploitation des variables de résultat (requis après chaque instruction CADAPTON/ CADAPTOF)`
`   MSG("CADAPT-RESULT=" << RESULT)`	
`   STOPRE`	
`   SETAL(61000)`	
`ENDIF`	
`...`	
`CADAPTOF(RESULT,MX1,2)`	`; Désactiver l'adaptation`
`...`	`; Exploitation des variables de résultat`
`...`	

Informations complémentaires

Axes Link

La fonction ne peut pas être utilisée sur des axes Link (fonction NCU-Link). La variable de résultat fournirait dans ce cas la valeur de retour 2 ("Paramètre <Axis> non valable").

Recherche de bloc

- **Recherche de bloc sans calcul**
  Les instructions CADAPTON/CADAPTOF programmées entre le début du programme et le bloc de destination sont ignorées. L'utilisateur doit choisir la destination de recherche de sorte que les adaptations importantes pour la phase d'exécution soient sélectionnées.
- **Recherche de bloc avec calcul au contour ou au point final du bloc**
  Les instructions d'activation ou de désactivation ainsi que les valeurs d'entrée éventuellement programmées sont collectées et l'état actuel à la fin de la recherche est émis avec les blocs d'action de recherche.
- **Recherche de bloc avec calcul en mode Test de programme (SERUPRO)**
  Avec la désactivation de SERUPRO, l'état des adaptations atteint par les instructions CADAPTON/CADAPTOF est transféré dans le fonctionnement réel.

3.15.9 Adaptation du filtre d'à-coup FIR au mode dynamique (CALCFIR)

Pour obtenir un effet d'amortissement et une précision de contour identiques après un changement du mode dynamique pour tous les axes du canal pour lesquels un filtre d'à-coup passe-bas FIR est activé, l'instruction de langage CN CALCFIR permet de calculer et d'activer les réglages de filtre FIR dépendants de la dynamique.

Application

CALCFIR est utilisé dans le cycle constructeur CUST_832.

Déroulement standard

CYCLE832 est programmé au début d'un programme CN pour l'usinage des surfaces de forme libre. Lors de l'exécution du programme, CYCLE832 appelle le cycle constructeur CUST_832, puis l'instruction de langage CN CALCFIR insérée par le constructeur de machines dans CUST_832 s'exécute.

Prise d'effet

Les réglages de filtre écrasés par CALCFIR restent actifs jusqu'à ce que le prochain appel de CALCFIR les écrasent à leur tour. Cela est toujours nécessaire lorsqu'une modification du mode dynamique actuel est programmée par une instruction du groupe G 59 (mode dynamique pour interpolation de trajectoire).

Comportement si démarrage de la commande / remise à zéro du canal / remise à zéro à la fin du programme

Lors du démarrage de la commande et de la remise à zéro du canal / à la fin du programme, les réglages de filtre FIR sont activés automatiquement d'une manière adaptée à la position d'effacement du groupe G 59.

Conditions

- L'option "Top Speed" est sélectionnée.

  Remarque

  Pour obtenir des résultats optimaux avec "Top Speed", il est recommandé d'utiliser en plus l'option "Top Surface".

- Pour les axes du groupe d'interpolation, les conditions suivantes doivent être remplies :
  - Le filtre d'à-coup passe-bas FIR est activé.
  - Les réglages de filtre FIR dépendants de la dynamique sont configurés.
  - L'écrasement des réglages de filtre par CALCFIR est autorisé.
- La limitation des à-coups SOFT/SOFTA est activée.

Syntaxe

CALCFIR est programmé dans un bloc qui lui est propre. L'appel dans le cycle constructeur CUST_832 a lieu immédiatement après la programmation du mode dynamique :

```
...
DYN...
CALCFIR
...
```

Signification

DYN... :	Instruction du groupe G 59 pour la sélection du mode dynamique	
	DYNNORM :	Activation de la **dynamique normale**
	DYNPOS :	Activation de la dynamique pour **mode positionnement, taraudage**
	DYNROUGH :	Activation de la dynamique pour **ébauche**
	DYNSEMIFIN :	Activation de la dynamique pour **préfinition**
	DYNFINISH :	Activation de la dynamique pour **finition**
	DYNPREC :	Activation de la dynamique pour **superfinition**
CALCFIR :	Procédure prédéfinie pour l'adaptation de dynamique des filtres d'à-coup passe-bas FIR	

Remarque

CALCFIR exécute un NEWCONFIG implicite afin de rendre le résultat du calcul de filtre FIR actif dans les paramètres machine spécifiques à l'axe pour le paramétrage des filtres d'à-coup passe-bas FIR comme pour CPRECON (Page 871). Ce NEWCONFIG implicite a pour conséquence que d'autres paramètres machine activés par NEWCONFIG peuvent être activés.

Autres conditions à prendre en compte

Utilisation dans des actions synchrones

CALCFIR n'est **pas** programmable dans des actions synchrones.

3.16 couplages d'axes

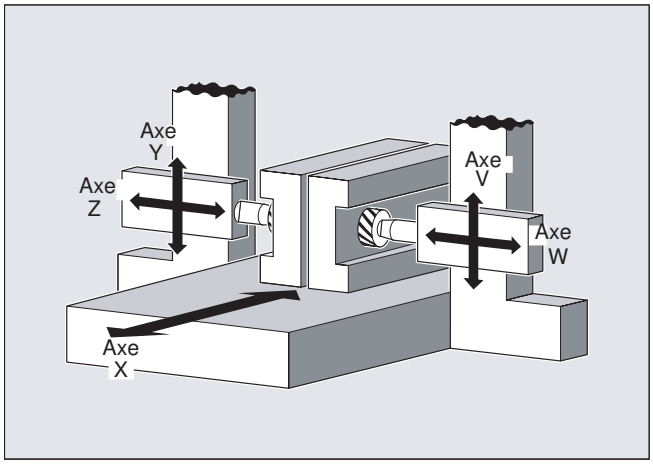
3.16.1 Déplacements conjugués (TRAILON, TRAILOF)

Lors du déplacement d'un axe pilote défini, les axes conjugués qui lui sont affectés (= axes asservis) décrivent les courses qui découlent de ce déplacement, compte tenu d'un facteur de couplage.

L'axe pilote et les axes asservis constituent un groupe d'axes à déplacements conjugués.

Domaines d'application

- Déplacement d'un axe par le biais d'un autre axe simulé. L'axe pilote est un axe simulé et l'axe conjugué un axe réel. Ainsi, l'axe réel peut être déplacé en tenant compte du facteur de couplage.
- Usinage bilatéral avec 2 groupe d'axes à déplacements conjugués :
  1. Axe pilote Y, axe conjugué V
  2. Axe pilote Z, axe conjugué W



Syntaxe

```
TRAILON(<axe asservi>,<axe pilote>,<facteur de couplage>)
TRAILOF(<axe asservi>,<axe pilote>,<axe pilote 2>)
TRAILOF(<axe asservi>)
```

Signification

`TRAILON` :	Instruction pour l'activation et la définition d'un groupe d'axes à déplacements conjugués	
	Prise d'effet :	modale
`<Axe asservi>` :	Paramètre 1 : désignation de l'axe conjugué (asservi) **Remarque :** un axe conjugué peut également être l'axe pilote d'autres axes conjugués. Ceci permet de réaliser des groupes d'axes à déplacements conjugués interdépendants.	

<Axe pilote> :	Paramètre 2 : désignation de l'axe pilote
<Facteur de couplage> :	Paramètre 3 : facteur de couplage Le facteur de couplage indique le rapport souhaité entre la course de l'axe conjugué et celle de l'axe pilote : <facteur de couplage> = course de l'axe conjugué/course de l'axe pilote
	Type : REAL
	Réglage par défaut : 1
	En entrant une valeur négative, les déplacements de l'axe pilote et de l'axe conjugué s'effectuent en sens opposé. Si le facteur de couplage n'est pas indiqué lors de la programmation, le facteur de couplage 1 est pris en compte automatiquement.
TRAILOF :	Instruction de désactivation d'un groupe d'axes à déplacements conjugués
	Prise d'effet : modale
	TRAILOF avec 2 paramètres désactive uniquement le couplage à l'axe pilote indiqué : TRAILOF(<axe asservi>,<axe pilote>) Si un axe conjugué possède 2 axes pilotes, TRAILOF peut être appelé avec 3 paramètres pour désactiver les deux couplages : TRAILOF(<axe asservi>,<axe pilote>,<axe pilote 2>) La programmation de TRAILOF sans indication d'axe pilote fournit le même résultat : TRAILOF(<axe asservi>)

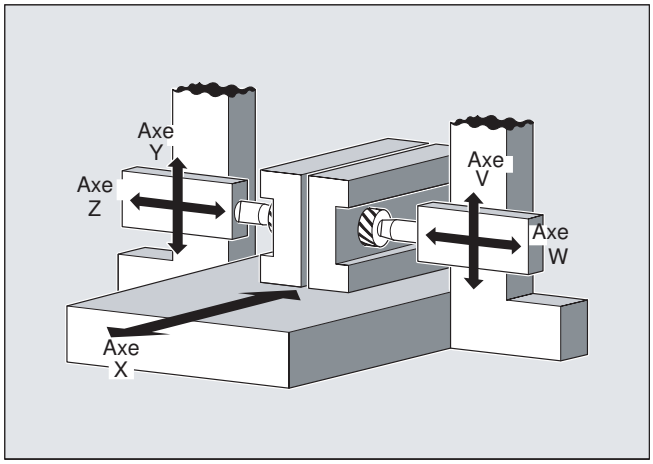
Remarque

Les déplacements conjugués s'effectuent toujours dans le système de coordonnées de base (SCB).

Le nombre de groupes d'axes à déplacements conjugués que vous pouvez activer simultanément dépend uniquement des combinaisons d'axes possibles sur la machine.

Exemple

La pièce doit être usinée simultanément sur deux faces avec la configuration d'axes représentée ci-contre. Pour ce faire, former deux groupes d'axes à déplacements conjugués.



Code de programme	Commentaire
…	
N100 TRAILON(V,Y)	; Activation du 1er groupe d'axes à déplacements conjugués
N110 TRAILON(W,Z,-1)	; Activation du 2e groupe d'axes à déplacements conjugués. Facteur de couplage négatif : l'axe conjugué se déplace en sens opposé par rapport à l'axe pilote.
N120 G0 Z10	; Déplacement des axes Z et W en sens opposé.
N130 G0 Y20	; Déplacement des axes Y et V dans le même sens.
…	
N200 G1 Y22 V25 F200	; Superposition d'un déplacement dépendant et d'un déplacement indépendant de l'axe conjugué V.
…	
TRAILOF(V,Y)	; Désactivation du 1er groupe d'axes à déplacements conjugués.
TRAILOF(W,Z)	; Désactivation du 2e groupe d'axes à déplacements conjugués.

Informations complémentaires

Types d'axes

Un groupe d'axes à déplacements conjugués peut combiner à volonté axes linéaires et axes rotatifs. Un axe simulé peut aussi être défini comme axe pilote.

Axes conjugués

A un axe conjugué, vous pouvez affecter deux axes pilotes à la fois. L'affectation s'effectue dans des groupes d'axes à déplacements conjugués différents.

Un axe conjugué peut être programmé avec toutes les instructions de déplacement disponibles (G0, G1, G2, G3, ...). En plus des déplacements définis indépendamment, l'axe conjugué décrit les courses déduites de ses axes pilotes avec les facteurs de couplage.

Limitation dynamique

La limitation dynamique dépend du mode d'activation du groupe d'axes à déplacements conjugués :

- Activation dans le programme pièce
  Lorsque l'activation s'effectue dans le programme pièce et que tous les axes pilotes sont des axes du programme dans le canal d'activation, la dynamique de chaque axe conjugué est prise en compte lors du déplacement des axes pilotes, de sorte à ce qu'aucun axe conjugué ne soit surchargé.
  Lorsque l'activation s'effectue dans le programme pièce avec des axes pilotes qui ne sont pas des axes de programme dans le canal d'activation ($AA_TYP ≠ 1), la dynamique de l'axe conjugué n'est pas prise en compte lors du déplacement des axes pilotes. Dans le cas d'axes conjugués, dont la dynamique est inférieure à celle requise pour le couplage, il peut ainsi en résulter une surcharge.
- Activation dans une action synchrone
  Lorsque l'activation s'effectue dans une action synchrone, la dynamique des axes conjugués n'est pas prise en compte lors du déplacement des axes pilotes. Dans le cas d'axes conjugués, dont la dynamique est inférieure à celle requise pour le couplage, il peut ainsi en résulter une surcharge.

⚠ PRUDENCE

Surcharge de l'axe

Lorsqu'un groupe d'axes à déplacements conjugués est activé dans

- des actions synchrones,
- le programme pièce avec des axes pilotes qui ne sont pas des axes de programme dans le canal de l'axe conjugué,

il relève de la responsabilité expresse de l'utilisateur / du constructeur de machines de prévoir des mesures adaptées, afin que les déplacements de l'axe pilote n'entraînent pas de surcharge des axes conjugués.

Etat du couplage

L'interrogation de l'état du couplage d'un axe peut être réalisée dans le programme pièce au moyen de la variable système :

$AA_COUP_ACT[<axe>]

Valeur	Signification
0	aucun couplage activé
8	déplacements conjugués actifs

Affichage de la distance restant à parcourir par les axes conjugués pour des axes rotatifs modulo

Si les axes pilote et conjugués sont des axes rotatifs modulo, les déplacements de l'axe pilote de n * 360°, où n = 1, 2, 3... , s'additionnent dans l'affichage de la distance restant à parcourir des axes conjugués jusqu'à la désactivation du couplage.

Exemple : section de programme avec TRAILON, axe pilote B et axe asservi C

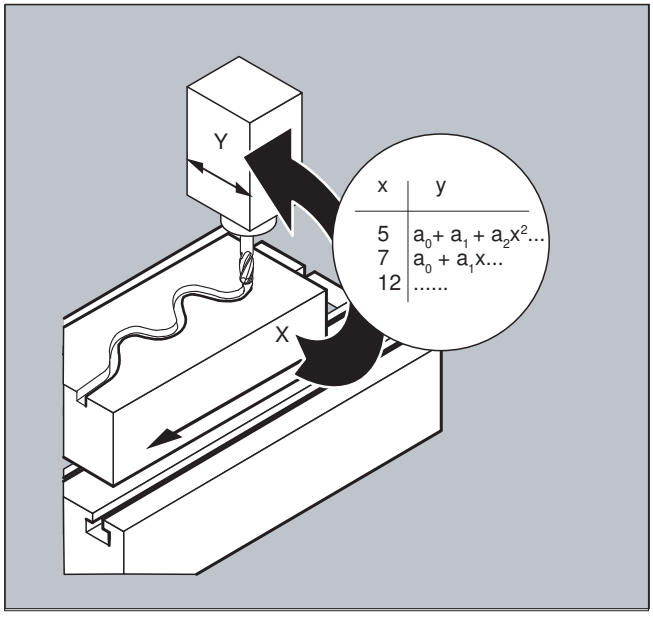
Code de programme	**Commentaire**
`TRAILON(C,B,1)`	`; Activation du couplage`
`G0 B0`	`; Position de départ`
	`; Affichage de la distance restant à parcourir au début du bloc :`
`G91 B360`	`; B=360, C=360`
`G91 B720`	`; B=720, C=1080`
`G91 B360`	`; B=360, C=1440`

3.16.2 Tables de courbes (CTAB)

Les tables de courbes permettent de programmer des rapports de position et de vitesse entre deux axes (axe pilote et axe asservi). La définition d'une table de courbes s'effectue dans le programme pièce.

Application

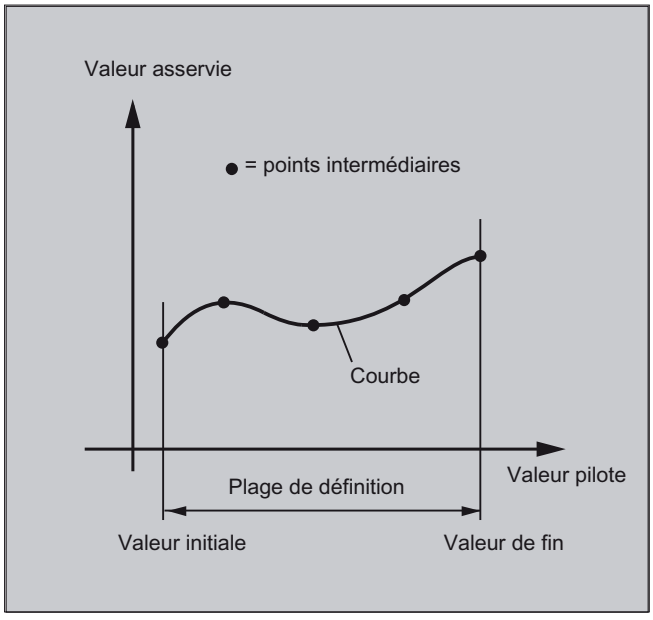
Les tables de courbes remplacent des cames mécaniques. la table de courbes est à la base du couplage de deux axes par valeur pilote, puisqu'elle crée le lien fonctionnel entre valeur pilote et valeur asservie : A partir des positions en corrélation réciproque de l'axe asservi et de l'axe pilote, la commande calcule conformément à la programmation, un polynôme équivalent à la came qu'il est appelé à remplacer.



3.16.2.1 Définition de tables de courbes (CTABDEF, CATBEND)

Une table de courbes représente un programme pièce ou une partie d'un programme pièce qui est identifié(e) par l'instruction CTABDEF en début et l'instruction CTABEND en fin de programme.

Dans cette section du programme pièce, on met en correspondance, par le biais d'instructions de déplacement, des positions isolées de l'axe pilote avec des positions univoques de l'axe asservi ; ces positions servent de points intermédiaires pour le calcul d'une courbe sous forme d'un polynôme du 5e degré au maximum.



Condition

La définition de tables de courbes requiert la réservation d'un espace mémoire au moyen d'une configuration correspondante des paramètres machine (→ constructeur de machine !).

Syntaxe

```
CTABDEF(<axe asservi>,<axe pilote>,<n>,<périodicité>[,<lieu
d'enregistrement>])
...
CTABEND
```

Signification

`CTABDEF( ) :`	Début de la définition de tables de courbes
`CTABEND :`	Fin de la définition de tables de courbes
`<Axe asservi> :`	Axe dont il s'agit de calculer le déplacement au moyen de la table de courbes
`<Axe pilote> :`	Axe fournissant les valeurs pilotes pour le calcul du déplacement de l'axe asservi

<n> :	Numéro (**ID**) de la table de courbes Le numéro d'une table de courbes est univoque et indépendant du lieu d'enregistrement. Des tables portant le même numéro ne peuvent pas se trouver à la fois dans la mémoire statique et dans la mémoire dynamique de la CN.	
<périodicité> :	Périodicité de la table	
	0	la table n'est pas périodique (elle n'est traitée qu'une seule fois, même dans le cas d'axes rotatifs)
	1	la table est périodique par rapport à l'axe pilote
	2	la table est périodique par rapport à l'axe pilote et l'axe asservi
<lieu de stockage> :	Indication du lieu d'enregistrement (facultative)	
	"SRAM"	La table de courbes est créée dans la mémoire **statique** de la CN.
	"DRAM"	La table de courbes est créée dans la mémoire **dynamique** de la CN.
	Remarque : si aucune valeur n'est programmée pour ce paramètre, le lieu d'enregistrement par défaut paramétré avec PM20905 $MC_CTAB_DEFAULT_MEMORY_TYPE est utilisé.	

Remarque

Ecraser

Une table de courbes est écrasée lorsque son numéro (<n>) est utilisé pour la nouvelle définition d'une table de courbes (exception : la table de courbes est active dans un couplage d'axes ou bloquée avec CTABLOCK). **L'écrasement n'est pas accompagné d'un avertissement correspondant !**

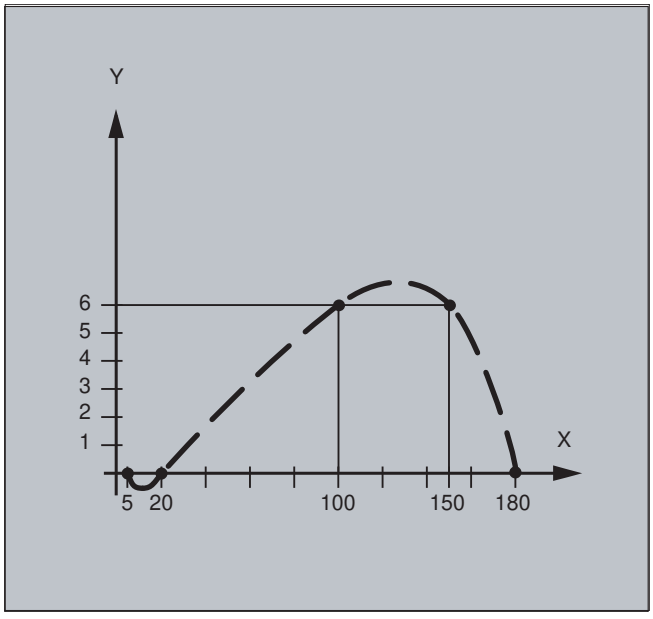
Exemples

Exemple 1 : section de programme comme définition d'une table de courbes

Une partie d'un programme doit servir, sans être modifiée, à la définition d'une table de courbes. L'instruction d'arrêt du prétraitement des blocs STOPRE qui y figure peut y être conservée et sera réactivée aussitôt que cette section de programme ne sera plus utilisée pour la définition d'une table de courbes, c'est-à-dire, dès que CTABDEF et CTABEND auront été supprimés.

```
Code de programme                Commentaire
…
CTABDEF(Y,X,1,1)                 ; Définition d'une table de courbes.
…
IF NOT ($P_CTABDEF)
STOPRE
ENDIF
…
CTABEND
```

Exemple 2 : définition d'une table de courbes non périodique



Code de programme	**Commentaire**
N100 CTABDEF(Y,X,3,0)	; Début de la définition d'une table de courbes non périodique ayant le numéro 3.
N110 X0 Y0	; 1re instruction de déplacement, définit les valeurs de départ et le 1er point intermédiaire : valeur pilote : 0, valeur asservie : 0
N120 X20 Y0	; 2e point intermédiaire : valeur pilote : 0…20, valeur asservie : valeur de départ...0
N130 X100 Y6	; 3e point intermédiaire : valeur pilote : 20…100, valeur asservie : 0…6
N140 X150 Y6	; 4e point intermédiaire : valeur pilote : 100…150, valeur asservie : 6…6
N150 X180 Y0	; 5e point intermédiaire : valeur pilote : 150…180, valeur asservie : 6…0
N200 CTABEND	; Fin de la définition. La table de courbes est générée dans sa représentation interne, en tant que polynôme du 5e degré au maximum. Le calcul de la courbe avec les points intermédiaires indiqués dépend du mode d'interpolation à effet modal sélectionné (interpolation circulaire, linéaire, de type spline). L'état du programme pièce avant le début de la définition est restauré.

Exemple 3 : Définition d'une table de courbes périodique

Définition d'une table de courbes périodique ayant le numéro 2, une plage de valeurs pilotes de 0 à 360, un déplacement de l'axe asservi de 0 à 45 avec retour à 0 :

```
Code de programme                                   Commentaire
N10 DEF REAL DEPPOS
N20 DEF REAL GRADIENT
N30 CTABDEF(Y,X,2,1)                                ; Début de la définition.
N40 G1 X=0 Y=0
N50 POLY
N60 PO[X]=(45.0)
N70 PO[X]=(90.0) PO[Y]=(45.0,135.0,-90)
N80 PO[X]=(270.0)
N90 PO[X]=(315.0) PO[Y]=(0.0,-135.0,90)
N100 PO[X]=(360.0)
N110 CTABEND                                        ; Fin de la définition.
; Test de la courbe par couplage de Y à X :
N120 G1 F1000 X0
N130 LEADON(Y,X,2)
N140 X360
N150 X0
N160 LEADOF(Y,X)

N170 DEPPOS=CTAB(75.0,2,GRADIENT)                   ; Lecture de la fonction de la ta-
                                                    ble à la valeur pilote 75.0.
N180 G0 X75 Y=DEPPOS                                ; Positionnement de l'axe pilote
                                                    et de l'axe asservi.
Quand le couplage est activé, il n'est pas nécessaire de synchroniser l'axe asservi.
N190 LEADON(Y,X,2)
N200 G1 X110 F1000
N210 LEADOF(Y,X)
N220 M30
```

plus d'informations

Valeur de départ et valeur de fin de la table de courbes

Le couple de positions axiales corrélées que l'on indique en premier (il s'agit de la première instruction de déplacement) dans la définition de la table de courbes est considéré comme valeur de départ de la plage de définition de la table de courbes. La valeur finale de la plage de définition de la table de courbes est constituée par la dernière instruction de déplacement.

Langage disponible

Pour définir la table de courbes, vous disposez du langage CN dans son intégralité.

Remarque

Les indications suivantes ne sont pas autorisées dans les définitions de tables de courbes :

- Arrêt du prétraitement des blocs
- Variations brusques dans le déplacement de l'axe pilote (en cas de changement de transformation par exemple),
- Instruction de déplacement seule pour l'axe asservi,
- Inversion du déplacement de l'axe pilote, autrement dit la position de l'axe pilote ne doit jamais être ambiguë,
- Programmation des instructions CTABDEF et CTABEND dans des niveaux de programme différents.

Effet d'instructions modales

Toutes les instructions à effet modal qui interviennent dans la définition de la table de courbes deviennent caduques dès que la définition est terminée. En d'autres termes, le programme pièce dans lequel a eu lieu la définition, présente le même état avant et après la définition de la table de courbes.

Affectations à des paramètres R

Les affectations à des paramètres R dans la définition d'une table de courbes sont réinitialisées après CTABEND.

Exemple :

Code de programme	**Commentaire**
...	
R10=5 R11=20	; R10=5
...	
CTABDEF	
G1 X=10 Y=20 F1000	
R10=R11+5	; R10=25
X=R10	
CTABEND	
...	; R10=5

Activation de ASPLINE, BSPLINE, CSPLINE

Si une spline ASPLINE, BSPLINE ou CSPLINE est activée à l'intérieur d'une définition de table de courbes CTABDEF ... CTABEND , au moins un point de départ doit être programmé avant cette activation de spline. Cela devrait éviter une activation immédiate après CTABDEF, compte tenu que la spline dépend de la position d'axe courante avant la définition de table de courbes.

Exemple :

Code de programme
...

Code de programme
CTABDEF(Y,X,1,0)
X0 Y0
ASPLINE
X=5 Y=10
X10 Y40
...
CTABEND

Réutilisation de tables de courbes

Le lien fonctionnel créé entre l'axe pilote et l'axe asservi par la table de courbes est mémorisé sous le numéro de table qui a été choisi ; il est conservé au-delà de la fin du programme et d'un POWER OFF si la table a été enregistrée dans la mémoire statique de la CN (SRAM).

Une table créée dans la mémoire dynamique (DRAM) est supprimée lors d'un POWER ON et doit éventuellement être recréée.

Une fois créée, une table de courbes est utilisable pour des combinaisons quelconques entre axe pilote et axe asservi, indépendamment des axes qui ont servi à la créer.

Écrasement de tables de courbes

Dès que vous redéfinissez une table de courbe sous un numéro de table déjà utilisé, vous écrasez la table existante.

Exception : dans un couplage d'axes, une table de courbes est active ou bloquée avec CTABLOCK.

Remarque

L'écrasement d'une table de courbes n'est pas accompagné d'un avertissement particulier !

Définition de table de courbe active ?

La variable système `$P_CTABDEF` permet à tout moment d'interroger le système à partir du programme pièce, afin de savoir si une définition de table de courbes est active.

Annulation de la définition d'une table de courbes

Après la mise entre parenthèses des instructions de définition de la table de courbes, la section du programme pièce peut à nouveau être utilisée comme un programme pièce réel.

Chargement de tables de courbes avec "Exécution d'un programme externe"

Pour l'exécution externe de tables de courbes, la taille du tampon de rechargement (DRAM) doit être sélectionnée à l'aide de PM18360 $MN_MM_EXT_PROG_BUFFER_SIZE de sorte que celui-ci puisse contenir la définition complète des tables de courbes. Si ce n'est pas le cas, l'exécution du programme pièce est annulée et une alarme est émise.

Sauts de l'axe asservi

En l'absence de déplacement,
des sauts de l'axe asservi sont tolérables en fonction du paramètre machine :
MD20900 $MC_CTAB_ENABLE_NO_LEADMOTION

3.16.2.2 Vérification de l'existence d'une table de courbes (CTABEXISTS)

L'instruction CTABEXISTS permet de vérifier la présence d'un numéro donné de table de courbes dans la mémoire de la CN.

Syntaxe

```
CTABEXISTS(<n>)
```

Signification

CTABEXISTS :	Vérification de la présence de la table de courbes de numéro <n> dans la mémoire statique ou dynamique de la CN	
	0	La table n'existe pas.
	1	La table existe
<n> :	Numéro (**ID**) de la table de courbes	

3.16.2.3 Suppression d'une table de courbes (CTABDEL)

CTABDEL permet de supprimer des tables de courbes.

Remarque

Vous ne pouvez pas effacer des tables de courbes quand elles sont activées dans un couplage d'axes.

Syntaxe

```
CTABDEL(<n>)
CTABDEL(<n>,<m>)
CTABDEL(<n>,<m>,<lieu d'enregistrement>)
CTABDEL()
CTABDEL(,,<lieu d'enregistrement>)
```

Signification

CTABDEL :	Instruction de suppression de tables de courbes
<n> :	Numéro (**ID**) de la table de courbes à supprimer Lors de la suppression d'une plage de tables de courbes CTABDEL(<n>,<m>), <n> indique le numéro de la première table de courbes de la plage.
<m> :	Lors de la suppression d'une plage de tables de courbes CTABDEL(<n>,<m>), <m> indique le numéro de la dernière table de courbes de la plage. <m> doit être supérieur à <n> !

`<lieu de stockage>:`	Indication du lieu d'enregistrement (facultative) Lors d'une suppression **sans** indication de lieu d'enregistrement, les tables de courbes indiquées sont supprimées dans les mémoires statique et dynamique de la CN. Lors d'une suppression **avec** indication de lieu d'enregistrement, seules les tables de courbes indiquées se trouvant dans la mémoire spécifiée sont supprimées. Les autres tables de courbes sont conservées.	
	`"SRAM"`	Suppression dans la mémoire **statique** de la CN
	`"DRAM"`	Suppression dans la mémoire **dynamique** de la CN

Si `CTABDEL` est programmé sans indication de la table de courbes à supprimer, **toutes** les tables de courbes ou toutes celles qui se trouvent dans la mémoire spécifiée sont supprimées :

`CTABDEL():`	Supprime toutes les tables de courbes dans les mémoires statique et dynamique de la CN
`CTABDEL(,,"SRAM"):`	Supprime toutes les tables de courbes dans la mémoire statique de la CN
`CTABDEL(,,"DRAM"):`	Supprime toutes les tables de courbes dans la mémoire dynamique de la CN

Remarque

Si, en cas de suppression multiple `CTABDEL(<n>,<m>)` ou `CTABDEL()`, l'une au moins des tables de courbes à supprimer est active dans un couplage, l'instruction de suppression n'est pas exécutée, autrement dit **aucune** des tables de courbes adressées n'est supprimée.

3.16.2.4 Verrouillage de tables de courbes contre la suppression et l'écrasement (CTABLOCK, CTABUNLOCK)

Les tables de courbes peuvent être protégées contre la suppression et l'écrasement par inadvertance au moyen d'un verrouillage. Ce dernier peut à tout moment de nouveau être annulé.

Syntaxe

Verrouillage :

```
CTABLOCK(<n>)
CTABLOCK(<n>,<m>)
CTABLOCK(<n>,<m>,<lieu d'enregistrement>)
CTABLOCK()
CTABLOCK(,,<lieu d'enregistrement>)
```

Annulation du verrouillage :

```
CTABUNLOCK(<n>)
CTABUNLOCK(<n>,<m>)
CTABUNLOCK(<n>,<m>,<lieu d'enregistrement>)
CTABUNLOCK()
CTABUNLOCK(,,<lieu d'enregistrement>)
```

Signification

CTABLOCK :	Instruction de **verrouillage** contre la suppression/l'écrasement	
CTABUNLOCK :	Instruction d'**annulation** du verrouillage contre la suppression/l'écrasement CTABUNLOCK libère les tables de courbes verrouillées avec CTABLOCK. Les tables de courbes qui agissent dans un couplage actif restent verrouillées et ne peuvent pas être supprimées. Le verrouillage avec CTABLOCK est annulé aussitôt que le couplage actif est désactivé. Cette table peut donc être effacée. Un nouvel appel de CTABUNLOCK n'est pas nécessaire.	
<n> :	Numéro (**ID**) de la table de courbes à verrouiller/déverrouiller Lors du verrouillage/déverrouillage d'une plage de tables de courbes CTABLOCK(<n>,<m>)/CTABUNLOCK(<n>,<m>), <n> indique le numéro de la première table de courbes de la plage.	
<m> :	Lors du verrouillage/déverrouillage d'une plage de tables de courbes CTABLOCK(<n>,<m>)/CTABUNLOCK(<n>,<m>), <m> indique le numéro de la dernière table de courbes de la plage. <m> doit être supérieur à <n> !	
<lieu de stockage> :	Indication du lieu d'enregistrement (facultative) Lors du verrouillage/déverrouillage **sans** indication de lieu d'enregistrement, les tables de courbes indiquées sont verrouillées/déverrouillées dans les mémoires statique et dynamique de la CN. Lors du verrouillage/déverrouillage **avec** indication de lieu d'enregistrement, seules les tables de courbes indiquées se trouvant dans la mémoire spécifiée sont verrouillées/déverrouillées. Les tables de courbes restantes ne sont pas verrouillées/déverrouillées.	
	"SRAM"	Verrouillage/déverrouillage dans la mémoire **statique** de la CN
	"DRAM"	Verrouillage/déverrouillage dans la mémoire **dynamique** de la CN

Si CTABLOCK/CTABUNLOCK est programmé sans indication de la table de courbes à verrouiller/déverrouiller, **toutes** les tables de courbes ou toutes celles qui se trouvent dans la mémoire spécifiée sont verrouillées/déverrouillées.

CTABLOCK() :	Verrouille toutes les tables de courbes dans les mémoires statique et dynamique de la CN
CTABLOCK(,,"SRAM") :	Verrouille toutes les tables de courbes dans la mémoire statique de la CN
CTABLOCK(,,"DRAM") :	Verrouille toutes les tables de courbes dans la mémoire dynamique de la CN
CTABUNLOCK() :	Déverrouille toutes les tables de courbes dans les mémoires statique et dynamique de la CN
CTABUNLOCK(,,"SRAM") :	Déverrouille toutes les tables de courbes dans la mémoire statique de la CN
CTABUNLOCK(,,"DRAM") :	Déverrouille toutes les tables de courbes dans la mémoire dynamique de la CN

3.16.2.5 Tables de courbes : détermination des propriétés de la table (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD)

Ces instructions permettent d'interroger les principales propriétés d'une table de courbes (numéro de la table, état de verrouillage, lieu d'enregistrement, périodicité).

Syntaxe

```
CTABID(<p>)
CTABID(<p>,<lieu d'enregistrement>)
CTABISLOCK(<n>)
CTABMEMTYP(<n>)
TABPERIOD(<n>)
```

Signification

CTABID :	Indique le **numéro de la table** inscrit comme <p>ème table de courbes dans la mémoire spécifiée. Exemple : CTABID(1,"SRAM") indique le numéro de la première table de courbes dans la mémoire statique de la CN. La première table de courbes correspond à celle dont le numéro est le plus élevé. **Remarque :** si l'ordre des tables de courbes est modifié dans la mémoire entre des appels consécutifs de CTABID, p. ex. pour la suppression de tables de courbes avec CTABDEL, CTABID(<p>,...) peut indiquer une autre table de courbes que précédemment avec le même numéro <p>.	
CTABISLOCK :	Indique l'**état de verrouillage** de la table de courbes portant le numéro <n> :	
	0	La table n'est pas verrouillée
	1	La table est verrouillée par CTABLOCK
	2	La table est verrouillée par le couplage actif
	3	La table est verrouillée par CTABLOCK et par le couplage actif
	-1	La table n'existe pas.
CTABMEMTYP :	Indique le **lieu d'enregistrement** de la table de courbes portant le numéro <n> :	
	0	table dans la mémoire statique de la CN
	1	table dans la mémoire dynamique de la CN
	-1	La table n'existe pas.
CTABPERIOD :	Indique la **périodicité** de la table de courbes portant le numéro <n> :	
	0	la table n'est pas périodique
	1	la table est périodique dans l'axe pilote
	2	la table est périodique dans l'axe pilote et dans l'axe asservi
	-1	La table n'existe pas.
<p> :	Numéro d'entrée en mémoire	
<n> :	Numéro (**ID**) de la table de courbes	

<lieu de stockage> :	Indication du lieu d'enregistrement (facultative)	
	"SRAM"	Mémoire **statique** de la CN
	"DRAM"	Mémoire **dynamique** de la CN
	Remarque : si aucune valeur n'est programmée pour ce paramètre, le lieu d'enregistrement par défaut paramétré avec PM20905 $MC_CTAB_DEFAULT_MEMORY_TYPE est utilisé.	

3.16.2.6 Lecture des valeurs de la table de courbes (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX)

Les valeurs suivantes de la table de courbes peuvent être lues dans le programme pièce :

- Valeurs de l'axe asservi et de l'axe pilote au début et à la fin d'une table de courbes
- Valeurs de l'axe asservi au début et à la fin d'un segment de courbe
- Valeur de l'axe asservi correspondant à une valeur de l'axe pilote
- Valeur de l'axe pilote correspondant à une valeur de l'axe asservi
- Valeur minimale et valeur maximale de l'axe asservi
  - dans la plage de définition entière de la table de courbes
    ou
  - dans un intervalle défini de la table de courbes

Syntaxe

```
CTABTSV(<n>,<gradient>[,<axe asservi>])
CTABTEV(<n>,<gradient>[,<axe asservi>])
CTABTSP(<n>,<gradient>[,<axe pilote>])
CTABTEP(<n>,<gradient>[,<axe pilote>])
CTABSSV(<valeur pilote>,<n>,<gradient>[,<axe asservi>])
CTABSEV(<valeur pilote>,<n>,<gradient>[,<axe asservi>])
CTAB(<valeur pilote>,<n>,<gradient>[,<axe asservi>,<axe pilote>]
CTABINV(<valeur asservie>,<valeur approchée>,<n>,<gradient>[,<axe
asservi>,<axe pilote>]
CTABTMIN(<n>[,<axe asservi>])
CTABTMAX(<n>[,<axe asservi>])
CTABTMIN(<n>,<a>,<b>[,<axe asservi>,<axe pilote>])
CTABTMAX(<n>,<a>,<b>[,<axe asservi>,<axe pilote>])
```

Signification

CTABTSV :	Lecture de la valeur de l'axe asservi au **début** de la table de courbes portant le numéro <n>
CTABTEV :	Lecture de la valeur de l'axe asservi à la **fin** de la table de courbes portant le numéro <n>
CTABTSP :	Lecture de la valeur de l'axe pilote au **début** de la table de courbes portant le numéro <n>

CTABTEP :	Lecture de la valeur de l'axe pilote à la **fin** de la table de courbes portant le numéro <n>
CTABSSV :	Lecture de la valeur de l'axe asservi au **début** du segment de courbe appartenant à la valeur de l'axe pilote spécifié (<valeur pilote>)
CTABSEV :	Lecture de la valeur de l'axe asservi à la **fin** du segment de courbe appartenant à la valeur de l'axe pilote spécifié (<valeur pilote>)
CTAB :	Lecture de la valeur de l'axe asservi correspondant à la valeur spécifiée de l'axe pilote (<valeur pilote>)
CTABINV :	Lecture de la valeur de l'axe pilote correspondant à la valeur spécifiée de l'axe asservi (<valeur asservie>)
CTABTMIN :	Définition de la **valeur minimale** de l'axe asservi : • dans la plage de définition entière de la table de courbes ou • dans un intervalle défini <a> ... <b>
CTABTMAX :	Définition de la **valeur maximale** de l'axe asservi : • dans la plage de définition entière de la table de courbes ou • dans un intervalle défini <a> ... <b>
<n> :	Numéro (**ID**) de la table de courbes
<Gradient> :	Le paramètre <gradient> indique la **pente** de la fonction représentant la table de courbes à la position déterminée
<Axe asservi> :	Axe dont il s'agit de calculer le déplacement au moyen de la table de courbes (facultatif)
<Axe pilote> :	Axe fournissant les valeurs pilotes pour le calcul du déplacement de l'axe asservi (facultatif)
<Valeur asservie> :	Valeur de l'axe asservi pour la lecture de la valeur correspondante de l'axe pilote pour CTABINV
<Valeur pilote> :	Valeur de l'axe pilote : • pour la lecture de la valeur correspondante de l'axe asservi pour CTAB ou • pour la sélection du segment de courbe pour CTABSSV/CTABSEV
<Valeur approchée> :	L'affectation d'une valeur de l'axe pilote à une valeur de l'axe asservi pour CTABINV ne doit pas toujours être univoque. C'est la raison pour laquelle CTABINV requiert, comme paramètre, une valeur approchée de la valeur recherchée de l'axe pilote.
<a> :	Limite **inférieure** de l'intervalle des valeurs pilotes pour CTABTMIN/CTABTMAX
<b> :	Limite **supérieure** de l'intervalle des valeurs pilotes pour CTABTMIN/CTABTMAX
	Remarque : L'intervalle des valeurs pilotes <a> ... <b> doit se situer dans la plage de définition de la table de courbes.

Exemples

Exemple 1 :

Déterminer les valeurs de l'axe asservi et de l'axe pilote au début et à la fin de la table de courbes ainsi que les valeurs minimale et maximale de l'axe asservi dans la plage de définition entière de la table de courbes.

```
Code de programme                          Commentaire
N10 DEF REAL STARTPOS
N20 DEF REAL ENDPOS
N30 DEF REAL STARTPARA
N40 DEF REAL ENDPARA
N50 DEF REAL MINVAL
N60 DEF REAL MAXVAL
N70 DEF REAL GRADIENT
...
N100 CTABDEF(Y,X,1,0)                      ; Début de la définition de la table
N110 X0 Y10                                ; Position de départ du 1er segment de table
N120 X30 Y40                               ; Position finale du 1er segment de table =
                                           position de départ du 2e segment de table
N130 X60 Y5                                ; Position finale du 2e segment de table = ...
N140 X70 Y30
N150 X80 Y20
N160 CTABEND                               ; Fin de la définition de la table.
...
N200 STARTPOS=CTABTSV(1,GRADIENT)          ; Valeur de l'axe asservi au début de la table
                                           de courbes = 10
N210 ENDPOS=CTABTEV(1,GRADIENT)            ; Valeur de l'axe asservi à la fin de la table
                                           de courbes = 20
N220 STARTPARA=CTABTSP(1,GRADIENT)         ; Valeur de l'axe pilote au début de la table
                                           de courbes = 0
N230 ENDPARA=CTABTEP(1,GRADIENT)           ; Valeur de l'axe pilote à la fin de la table
                                           de courbes = 80
N240 MINVAL=CTABTMIN(1)                    ; Valeur minimale de l'axe asservi pour Y=5
N250 MAXVAL=CTABTMAX(1)                    ; Valeur maximale de l'axe asservi pour Y=40
```

Exemple 2 :

Déterminer les valeurs de l'axe asservi au début et à la fin du segment de courbe correspondant à la valeur de l'axe pilote X=30.

```
Code de programme                          Commentaire
N10 DEF REAL STARTPOS
N20 DEF REAL ENDPOS
N30 DEF REAL GRADIENT
...
N100 CTABDEF(Y,X,1,0)                      ; Début de la définition de la table.
N110 X0 Y0                                 ; Position de départ du 1er segment de table
```

Code de programme	Commentaire
N120 X20 Y10	; Position finale du 1er segment de table = position de départ du 2e segment de table
N130 X40 Y40	; Position finale du 2e segment de table = ...
N140 X60 Y10	
N150 X80 Y0	
N160 CTABEND	; Fin de la définition de la table.
...	
N200 STARTPOS=CTABSSV(30.0,1,GRADIENT)	; Position de départ Y dans le 2e segment = 10
N210 ENDPOS=CTABSEV(30.0,1,GRADIENT)	; Position finale Y dans le 2e segment = 40

Informations complémentaires

Utilisation dans des actions synchrones

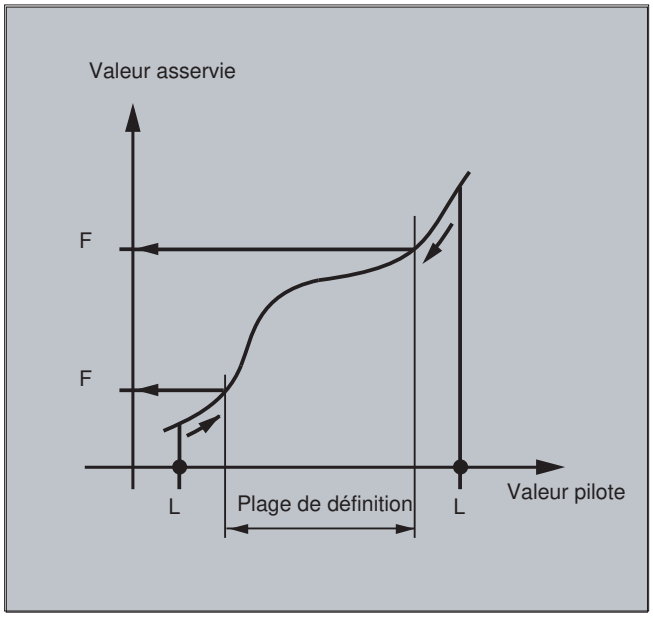
Il est également possible de lire dans des actions synchrones l'ensemble des instructions de lecture des valeurs d'une table de courbes (voir aussi le chapitre "Actions synchrones au déplacement").

Pour l'utilisation des instructions CTABINV, CTABTMIN et CTABTMAX, il faut s'assurer que :

- la puissance de la CN soit suffisante au moment de l'exécution
  ou
- le nombre de segments de la table de courbes soit interrogé avant l'appel afin de pouvoir subdiviser, le cas échéant, la table concernée

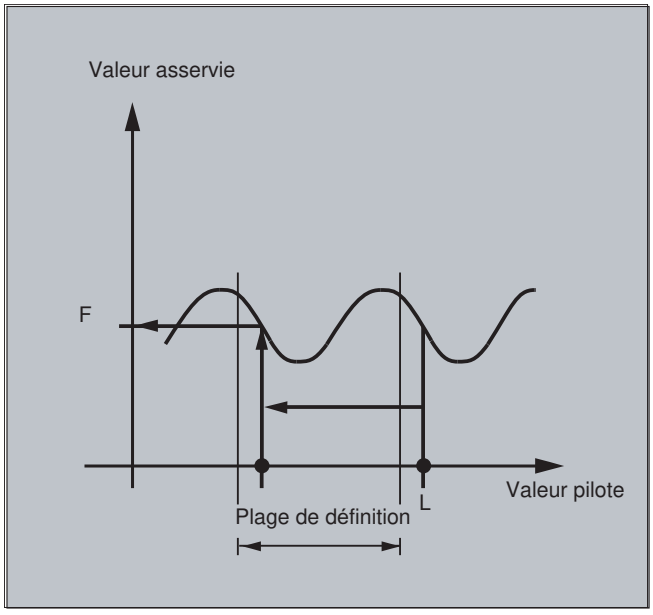
CTAB dans le cas de tables de courbes non périodiques

Lorsque la <valeur pilote> se situe en-dehors de la plage de définition, la valeur asservie fournie est la limite supérieure ou inférieure :



CTAB dans le cas de tables de courbes périodiques

Lorsque la <valeur pilote> indiquée se situe en dehors de la plage de définition, la valeur pilote modulo de la plage de définition est évaluée et la valeur asservie correspondante est fournie :



Valeur approchée pour CTABINV

Pour fournir la valeur pilote recherchée, l'instruction CTABINV requiert une valeur approchée. CTABINV fournit la valeur pilote la plus proche de la valeur approchée. Il peut p. ex. s'agir de la valeur pilote du dernier cycle d'interpolation.

Pente de la fonction représentant la table de courbes

L'indication de la pente (<gradient>) permet de calculer la vitesse de l'axe pilote ou de l'axe asservi à la position correspondante.

Indication de l'axe pilote ou de l'axe asservi

L'indication facultative de l'axe pilote ou de l'axe asservi est nécessaire lorsque l'axe pilote et l'axe asservi sont configurés avec des unités de longueur différentes.

CTABSSV, CTABSEV

Dans les cas suivants, les instructions CTABSSV et CTABSEV ne conviennent **pas** pour l'interrogation des segments programmés :

- des cercles ou des développantes sont programmés,
- le chanfreinage ou l'arrondissement avec CHF/RND est activé,
- le lissage avec G643 est activé,
- le compactage de bloc CN (COMP...) est activé.

3.16.2.7 Tables de courbes : Vérification de l'utilisation des ressources (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL)

Ces instructions permettent au programmeur de s'informer sur l'occupation actuelle des ressources requises par les tables de courbes, segments de tables et polynômes.

Syntaxe

```
CTABNO
CTABNOMEM(<lieu d'enregistrement>)
CTABFNO(<lieu d'enregistrement>)
CTABSEGID(<n>,<lieu d'enregistrement>)
CTABSEG(<lieu d'enregistrement>,<type de segment>)
CTABFSEG(<lieu d'enregistrement>,<type de segment>)
CTABMSEG(<type de segment>,<type de segment>)
CTABPOLID(<n>)
CTABPOL(<lieu d'enregistrement>)
CTABFPOL(<lieu d'enregistrement>)
CTABMPOL(<lieu d'enregistrement>)
```

Signification

`CTABNO` :	Détermination du nombre total de tables de courbes **définies** (dans les mémoires statique et dynamique de la CN)
`CTABNOMEM` :	Détermination du nombre de tables de courbes **définies** dans le `<lieu d'enregistrement>` indiqué
`CTABFNO` :	Détermination du nombre de tables de courbes qu'il est **encore possible** d'enregistrer dans le `<lieu d'enregistrement>` indiqué
`CTABSEGID` :	Détermination du nombre de segments de courbe avec le `<type de segment>` spécifié qu'il est possible d'utiliser dans la table de courbes de numéro `<n>`
`CTABSEG` :	Détermination du nombre de segments de courbe **utilisés** avec le `<type de segment>` spécifié dans le `<lieu d'enregistrement>` indiqué
`CTABFSEG` :	Détermination du nombre de segments de courbe avec le `<type de segment>` spécifié qu'il est **encore possible** d'enregistrer dans le `<lieu d'enregistrement>` indiqué
`CTABMSEG` :	Détermination du nombre **maximum possible** de segments de courbe avec le `<type de segment>` spécifié qu'il est possible d'enregistrer dans le `<lieu d'enregistrement>` indiqué
`CTABPOLID` :	Détermination du nombre de polynômes de courbe utilisés par la table de courbes de numéro `<n>`
`CTABPOL` :	Détermination du nombre de polynômes de courbes **utilisés** dans le `<lieu d'enregistrement>` indiqué
`CTABFPOL` :	Détermination du nombre de polynômes de courbes qu'il est **encore possible** d'enregistrer dans le `<lieu d'enregistrement>` indiqué
`CTABMPOL` :	Détermination du nombre **maximum possible** de polynômes de courbes qu'il est possible d'enregistrer dans le `<lieu d'enregistrement>` indiqué
`<n>` :	Numéro (**ID**) de la table de courbes

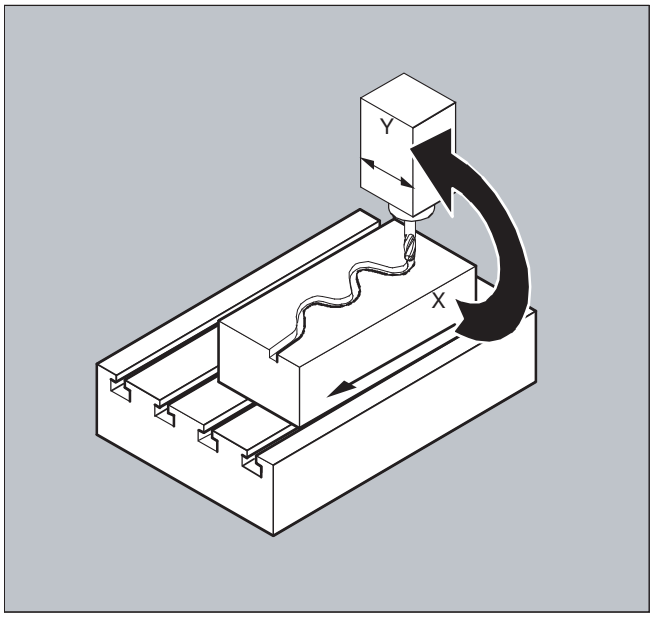
<lieu de stockage>:	Indication du lieu d'enregistrement (facultative)	
	"SRAM"	Mémoire **statique** de la CN
	"DRAM"	Mémoire **dynamique** de la CN
	Remarque : si aucune valeur n'est programmée pour ce paramètre, le lieu d'enregistrement par défaut paramétré avec PM20905 $MC_CTAB_DEFAULT_MEMORY_TYPE est utilisé.	
<type de segment>:	Indication du type de segment (facultative)	
	"L"	Segments linéaires
	"P"	Segments polynômiaux
	Remarque : si aucune valeur n'est programmée pour ce paramètre, la somme des segments linéaires et polynômiaux est fournie.	

3.16.3 Couplage de deux axes par valeur pilote (LEADON, LEADOF)

Remarque

Cette fonction n'est pas disponible pour SINUMERIK 828D.

Quand il y a couplage de deux axes par valeur pilote, l'axe pilote et l'axe asservi se déplacent de façon synchrone. Une position donnée de l'axe asservi correspond sans équivoque à une position de l'axe pilote (simulée le cas échéant) sur la base d'une table de courbe ou d'un polynôme en découlant.



Axe pilote est le nom donné à l'axe qui fournit les valeurs d'entrée pour la table de courbe. **Axe asservi** est le nom donné à l'axe qui prend les positions calculées sur la base de la table de courbe.

Couplage par valeur réelle et couplage par valeur de consigne

Comme valeurs pilotes, autrement dit comme valeurs de départ pour déterminer les positions de l'axe asservi, on peut utiliser :

- les valeurs réelles de l'axe pilote : Couplage par valeur réelle
- Valeurs de consigne de la position de l'axe pilote : Couplage par valeur de consigne

Le couplage par valeur pilote se réfère toujours au système de coordonnées de base.

Pour créer des tables de courbe, voir chapitre "Tables de courbe".

Syntaxe

```
LEADON(<axe asservi>,<axe pilote>,<n>)
LEADOR(<axe asservi>,<axe pilote>)
```

ou désactivation sans indication de l'axe pilote :

```
LEADOF(<axe asservi>)
```

Le couplage par valeur pilote peut être activé et désactivé aussi bien depuis le programme pièce que pendant les déplacements depuis une action synchrone.

Signification

`LEADON` :	Activer le couplage par valeur pilote
`LEADOF` :	Désactiver le couplage par valeur pilote
`<Axe asservi>` :	Axe asservi
`<Axe pilote>` :	Axe pilote
`<n>` :	Numéro de la table de courbe
`$SA_LEAD_TYPE` :	Permutation entre le couplage par valeur réelle et le couplage par valeur de consigne

Désactiver le couplage par valeur pilote, LEADOF

Après la désactivation du couplage par valeur pilote, l'axe asservi redevient axe de commande normal.

Couplage de deux axes par valeur pilote et différents états de fonctionnement, RESET

Selon le réglage qui a été fait dans les paramètres machine, les couplages par valeur pilote sont désactivés avec RESET.

Exemple de couplage par valeur pilote à partir d'une action synchrone

Sur une presse, un couplage mécanique conventionnel entre un axe pilote (axe d'emboutissage) et les axes d'un système de transfert est à remplacer par un couplage électronique.

Il démontre comment on peut remplacer sur une presse un système de transfert mécanique par un système de transfert électronique. Les phases de couplage et de découplage sont réalisées sous forme **d'actions synchrones statiques**.

Les axes de transfert et les axes auxiliaires sont pilotés par l'axe pilote VP (axe d'emboutissage) en tant qu'axes asservis, par le biais des tables de courbe.

Axes asservis

X - Axe d'avance ou axe longitudinal
YL - Axe de fermeture ou axe transversal
ZL - Axe de course
U - Avance par rouleaux, axe auxiliaire
V - Tête de dressage, axe auxiliaire
W - Graissage, axe auxiliaire

Actions

Parmi les actions synchrones, on trouve :

- couplage, `LEADON(<axe asservi>,<axe pilote>,<numéro de la table de courbe>)`
- Découplage, `LEADOF(<axe asservi>,<axe pilote>)`
- Préréglage des mémoires de valeurs réelles, `PRESETON(<axe>,<valeur>)`
- Définition du mémento, `$AC_MARKER[i]=<valeur>`
- Type de couplage : valeur pilote réelle/virtuelle
- Accostage de positions d'axes, `POS[<axe>]=<valeur>`

Conditions

Des entrées TOR rapides, des variables temps réel `$AC_MARKER` et des comparaisons de positions, combinées avec l'opérateur logique AND, sont exploitées comme conditions.

Remarque

Dans l'exemple qui suit, nous avons utilisé des retours de lignes, des retraits et des caractères **gras** pour une meilleure lisibilité du programme. Pour la commande, tout ce qui figure sous un numéro de ligne représente effectivement une ligne de programme.

Commentaire

Code de programme	**Commentaire**
	; Définit toutes les actions synchrones statiques.
	; **** Remise à zéro des mémentos
N2 $AC_MARKER[0]=0 $AC_MARKER[1]=0 $AC_MARKER[2]=0 $AC_MARKER[3]=0 $AC_MARKER[4]=0 $AC_MARKER[5]=0 $AC_MARKER[6]=0 $AC_MARKER[7]=0	
	; **** E1 0=>1 Couplage transfert ACTIF
N10 IDS=1 EVERY ($A_IN[1]==1) AND ($A_IN[16]==1) AND ($AC_MARKER[0]==0) DO LEADON(X,LW,1) LEADON(YL,LW,2) LEADON(ZL,LW,3) $AC_MARKER[0]=1	
	; **** E1 0=>1 Couplage dispositif d'avance à rouleaux ACTIF
N20 IDS=11 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_MARKER[5]==0) DO LEADON(U,LW,4) PRESETON(U,0) $AC_MARKER[5]=1	
	; **** E1 0->1 Couplage tête de dressage ACTIF
N21 IDS=12 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_MARKER[6]==0) DO LEADON(V,LW,4) PRESETON(V,0) $AC_MARKER[6]=1	
	; **** E1 0->1 Couplage graissage ACTIF

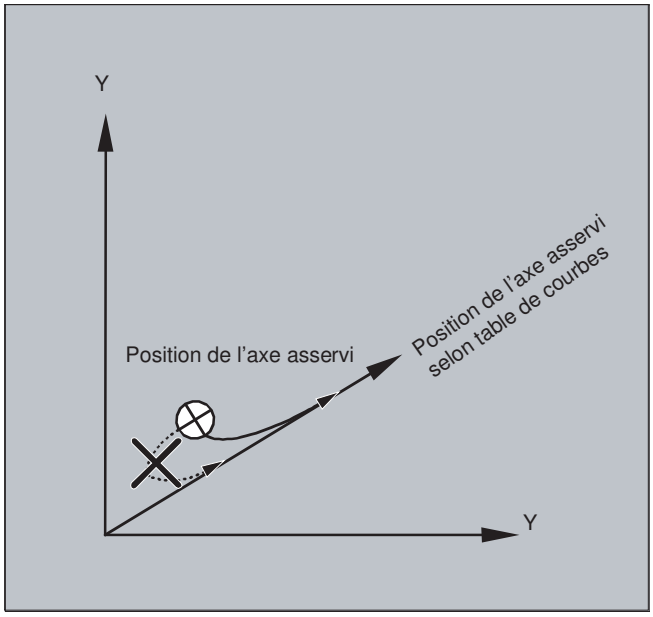
Code de programme	**Commentaire**
N22 IDS=13 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_MARKER[7]==0)	
DO LEADON(W,LW,4) PRESETON(W,0) $AC_MARKER[7]=1	
	; **** E2 0=>1 Couplage NON ACTIF
N30 IDS=3 EVERY ($A_IN[2]==1)	
DO LEADOF(X,LW) LEADOF(YL,LW) LEADOF(ZL,LW) LEADOF(U,LW) LEADOF(V,LW) LEADOF(W,LW) $AC_MARKER[0]=0	
$AC_MARKER[1]=0 $AC_MARKER[3]=0 $AC_MARKER[4]=0 $AC_MARKER[5]=0 $AC_MARKER[6]=0 $AC_MARKER[7]=0	
....	
N110 G04 F01	
N120 M30	

Description

Le couplage par valeur pilote exige la synchronisation de l'axe pilote et de l'axe asservi. Cette synchronisation est obtenue uniquement si l'axe asservi se trouve, au moment où le couplage par valeur pilote est activé, dans la plage de tolérance de la courbe calculée avec la table.

La plage de tolérance pour la position de l'axe asservi est définie par le paramètre machine `PM 37200 : COUPLE_POS_POL_COARSE A_LEAD_TYPE .`

Si, lors de l'activation du couplage par valeur pilote, l'axe asservi n'est pas encore positionné correctement, le synchronisme s'enclenchera automatiquement dès que la position de consigne calculée pour l'axe asservi approchera de sa position effective. Pendant la synchronisation, l'axe asservi est déplacé dans la direction définie par sa vitesse de consigne (calculée à partir de la vitesse de l'axe pilote et CTAB).

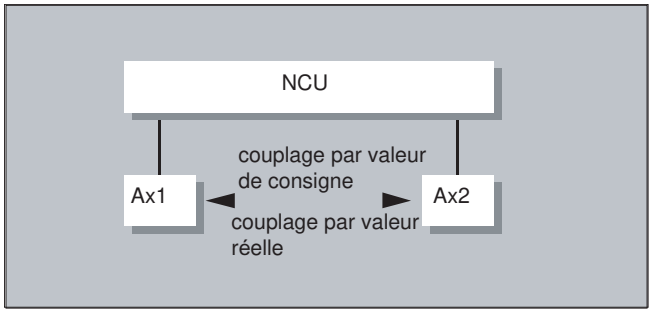


Pas de synchronisation

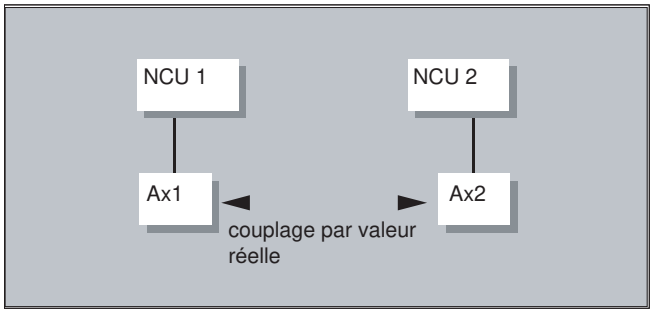
Si la position calculée de l'axe asservi s'éloigne de sa position effective lors de l'activation du couplage par valeur pilote, il n'y a pas de synchronisation.

Couplage par valeur réelle et couplage par valeur de consigne

Comparé au couplage par la valeur réelle, le couplage par la valeur de consigne fournit un meilleur synchronisme entre l'axe pilote et l'axe asservi ; c'est la raison pour laquelle il est préréglé en version standard.



Le couplage par la valeur de consigne est possible uniquement si l'axe pilote et l'axe asservi sont interpolés par la même NCU. Quand l'axe pilote est un axe externe, le couplage de l'axe asservi avec cet axe pilote ne peut se faire que par des valeurs réelles.



Le **basculement** est possible par le biais de la donnée de réglage $SA_LEAD_TYPE.

Le basculement entre couplage par valeur réelle et couplage par valeur de consigne doit toujours se faire quand l'axe asservi est en état d'immobilisation. En effet, une resynchronisation après le basculement n'a lieu que dans cet état.

Exemple d'application

Quand la machine est soumise à de fortes vibrations, la lecture des valeurs réelles ne peut pas se faire correctement. Pour mettre en œuvre le couplage par valeur pilote dans un transfert sur presse, il peut être nécessaire de basculer du couplage par valeur réelle au couplage par valeur de consigne, dans les phases d'usinage où les vibrations sont les plus fortes.

Simulation de valeurs pilotes dans un couplage par valeur de consigne

Par le biais d'un paramètre machine, on peut séparer l'interpolateur de l'axe pilote et le servo. On peut alors, dans un couplage par valeur de consigne, créer des valeurs de consigne sans qu'il y ait un déplacement réel de l'axe pilote.

Les valeurs pilotes créées par un couplage par la valeur de consigne peuvent être lues dans les variables suivantes, afin d'être utilisées dans des actions synchrones par exemple :

- $AA_LEAD_P	Valeur pilote de position
- $AA_LEAD_V	Valeur pilote de vitesse

Création de valeurs pilotes

Vous pouvez aussi créer des valeurs pilotes avec d'autres procédés que vous avez vous-même programmés. Les valeurs pilotes ainsi générées sont écrites dans les variables

- $AA_LEAD_SP	Valeur pilote de position
- $AA_LEAD_SV	Valeur pilote de vitesse

et lues à partir de celles-ci. Pour utiliser ces variables, il convient de définir la donnée de réglage comme suit : $SA_LEAD_TYPE = 2.

Etat du couplage

Vous pouvez interroger l'état du couplage dans le programme pièce CN avec la variable système suivante :

```
$AA_COUP_ACT[axe]
```
0 : aucun couplage activé
16 : couplage par valeur pilote activé

Gestion des états pour les actions synchrones

Les phases de commutation et de couplage sont gérées par des variables temps réel :

```
$AC_MARKER[i] = n
```
avec :
numéro de mémento i
valeur d'état n

3.16.4 Réducteur électronique (EG)

La fonction "Réducteur Electronique" permet, après un bloc de déplacement linéaire, de faire piloter le déplacement d'un **axe asservi** en fonction de 5 **axes pilotes** au maximum. Les relations entre les axes pilotes et l'axe asservi sont définis, pour chaque axe pilote, par un facteur de couplage.

Le trajet de l'axe asservi est calculé en additionnant les différents trajets des axes pilotes multipliés par les facteur de couplage respectifs. Lors de l'activation d'un groupe d'axes EG, une synchronisation de l'axe asservi à une position définie peut avoir lieu. Un groupe de réducteurs électroniques peut être :

- défini,
- activé,
- désactivé,
- effacé

.

Le déplacement de l'axe asservi peut être déduit des :

- valeurs de consignes des axes pilotes,
- valeurs réelles des axes pilotes.

Dans le cadre de l'extension, des couplages non linéaires entre les axes pilotes et les axes asservis peuvent être réalisés par des **tables de courbes** (voir chapitre Mode de déplacement). Les réducteurs électroniques peuvent être montés en cascade, c'est-à-dire que l'axe asservi d'un réducteur électronique peut être l'axe pilote d'un autre réducteur électronique.

3.16.4.1 Définition d'un réducteur électronique (EGDEF)

Un groupe d'axes EG est déterminé par l'indication de l'axe asservi et d'au moins un (au plus cinq) axes pilotes avec le type de couplage respectif.

Condition

Condition pour la définition d'un groupe d'axes EG :

un couplage d'axe ne doit pas être défini pour l'axe asservi (le cas échéant, un couplage existant doit être effacé avec EGDEL).

Syntaxe

```
EGDEF(axe asservi,axe pilote1,type de couplage1,axe pilote2,type de
couplage2,...)
```

Signification

`EGDEF` :	Définition d'un réducteur électronique	
`Axe asservi` :	Axe influencé par les axes pilotes	

`Axe pilote1`
`,...,`
`axe pilote5`
	Axes qui influencent l'axe asservi	

`Type de couplage1`
`,...,`
`type de couplage5`
	Type de couplage
	Les types de couplage pouvant être différents, ceux-ci doivent être indiqués pour tous les axes pilotes.	

	Valeur :	**Signification :**
	0	L'axe asservi est influencé par la **valeur réelle** de l'axe pilote correspondant.
	1	L'axe asservi est influencé par la **consigne** de l'axe pilote correspondant.

Remarque

Les facteurs de couplage sont préréglés avec la valeur zéro lors de la définition du groupe d'axes EG.

Remarque

EGDEF déclenche un arrêt du prétraitement des blocs. La définition de transmission avec EGDEF est alors à utiliser de manière inchangée si, sur les systèmes, un ou plusieurs axes pilotes influencent l'axe asservi par des **tables de courbes**.

Exemple

Code de programme	Commentaire
`EGDEF(C,B,1,Z,1,Y,1)`	`; Définition d'un groupe d'axes EG. Les axes pilotes B, Z, Y influencent l'axe asservi C par la valeur de consigne.`

3.16.4.2 Activation du réducteur électronique (EGON, EGONSYN, EGONSYNE)

Il existe 3 variantes pour activer un groupe d'axes EG.

Syntaxe

Variante 1 :

Le groupe d'axes EG est activé de façon sélective sans synchronisation par :
```
EGON(FA,"mode de changement de
bloc",AP1,Z1,N1,AP2,Z2,N2,...,AP5,Z5,N5)
```

Variante 2 :

Le groupe d'axes EG est activé de façon sélective avec synchronisation par :
```
EGONSYN(FA,"mode de changement de bloc",SynPosAA,
[,APi,SynPosAPi,Zi,Ni])
```

Variante 3 :

Le groupe d'axes EG est activé de façon sélective avec synchronisation et le mode d'accostage est défini par :
```
EGONSYNE(FA,"mode de changement de bloc",SynPosAA,mode
d'accostage[,APi,SynPosAPi,Zi,Ni])
```

Signification

Variante 1 :

FA	Axe asservi	
`mode de changement de bloc` :	Les modes suivants peuvent être utilisés :	
	`"NOC"`	Le changement de bloc s'effectue immédiatement.
	`"FINE"`	Le changement de bloc s'effectue au "synchronisme fin".
	`"COARSE"`	Le changement de bloc s'effectue au "synchronisme grossier".
	`"IPOSTOP"`	Le changement de bloc s'effectue à la valeur de consigne du synchronisme.
AP1, ... AP5	Axes pilotes	
Z1, ... Z5	Numérateur pour le facteur de couplage i	
N1, ... N5	Dénominateur pour le facteur de couplage i Facteur de couplage i = numérateur i / dénominateur i	

Seuls les axes pilotes spécifiés auparavant dans l'instruction EGDEF peuvent être programmés. Un axe pilote au moins doit être programmé.

Variante 2 :

FA	Axe asservi	
mode de changement de bloc :	Les modes suivants peuvent être utilisés :	
	"NOC"	Le changement de bloc s'effectue immédiatement.
	"FINE"	Le changement de bloc s'effectue au "synchronisme fin".
	"COARSE"	Le changement de bloc s'effectue au "synchronisme grossier".
	"IPOSTOP"	Le changement de bloc s'effectue à la valeur de consigne du synchronisme.
[,APi,SynPosAPi,Zi,Ni]	(ne pas écrire les crochets droits) Au minimum 1, au maximum 5 suites de :	
AP1, ... AP5	Axes pilotes	
PosSynAPi	Position de synchronisation pour l'axe pilote i	
Z1, ... Z5	Numérateur pour le facteur de couplage i	
N1, ... N5	Dénominateur pour le facteur de couplage i Facteur de couplage i = numérateur i / dénominateur i	

Seuls les axes pilotes spécifiés auparavant dans l'instruction EGDEF peuvent être programmés. Les "positions de synchronisme" programmés pour l'axe asservi (PosSynAA) et pour les axes pilotes (PosSynAP) sont des positions dans lesquelles le couplage d'axes est considéré *synchronisé*. Si le réducteur électronique n'est pas synchronisé à l'activation, l'axe asservi se déplace à la position de synchronisation définie.

Variante 3 :

Les paramètres sont identiques à ceux de la variante 2 avec en plus :

mode d'accostage :	Les modes suivants peuvent être utilisés :	
	"NTGT"	accoster l'entredent suivant avec une optimisation du temps
	"NTGP"	accoster l'entredent suivant avec une optimisation du parcours
	"ACN"	déplacer l'axe rotatif en sens de rotation négatif, en absolu
	"ACP"	déplacer l'axe rotatif en sens de rotation positif, en absolu
	"DCT"	Position de synchronisme programmée avec une optimisation du temps
	"DCP"	Position de synchronisme programmée avec une optimisation du parcours

La variante 3 influe uniquement sur les axes asservis à modulos couplés aux axes pilotes à modulos. L'optimisation du temps tient compte des limites de vitesse de l'axe asservi.

Informations complémentaires

Description des variantes d'activation

Variante 1 :

Les positions des axes pilotes et des axes asservis sont enregistrées, lors de l'activation, en tant que "positions de synchronisme". Les "positions de synchronisme" peuvent être lues à l'aide de la variable `$AA_EG_SYN`.

Variante 2 :

Si le groupe d'axes comprend des axes à valeur modulo, leurs positions sont ramenés à la plage définie par la valeur modulo. Ceci garantit que la position de synchronisation la plus proche est accostée (*synchronisation relative* : par exemple l'entredent suivant). Si le signal d'interface DB (30 +numéro d'axe), DBX 26 bit 4 "Déblocage forçage déplacement axe asservi" n'est pas à 1 pour l'axe asservi, la position de synchronisation n'est pas accostée. Dans ce cas, le programme est arrêté au bloc EGONSYN et l'alarme à effacement automatique 16771 est signalée jusqu'à ce que le signal mentionné ci-dessus soit mis à 1.

Variante 3 :

L'entredent (degrés) est calculé de la manière suivante : 360 * Zi/Ni. Si l'axe asservi est à l'arrêt au moment de l'activation, l'optimisation du parcours génère le même comportement que l'optimisation du temps.

Si l'axe asservi est déjà en déplacement, NTGP est synchronisé avec l'entredent suivant indépendamment de sa vitesse actuelle. Si l'axe asservi est déjà en déplacement, NTGP est synchronisé avec l'entredent suivant en fonction de sa vitesse actuelle. Le cas échéant, la vitesse est réduite.

Tables de courbes

Si une **table de courbes** est utilisée pour un axe pilote :

Ni	Le dénominateur du facteur de couplage linéaire (Ni) doit être mis à 0. (Le dénominateur 0 serait illicite pour des couplages linéaires). Pour la commande, le dénominateur 0 est l'identificateur Zi.
Zi	Zi doit être interprété comme le numéro de table de courbes à utiliser. La table de courbes avec le numéro indiqué doit être déjà définie lors de l'activation.
APi	L'indication de l'axe pilote (APi) correspond à l'indication de l'axe pilote lors du couplage par facteur (couplage linéaire).

Autres remarques concernant l'utilisation de tables de courbes, la cascade de réducteurs électroniques et leur synchronisation, voir :
Autres fonctions :
Description fonctionnelle Axes et broches

Comportement d'un réducteur électronique en cas de mise sous tension, RESET, changement de mode de fonctionnement, recherche de bloc

- Après la mise sous tension, **aucun** couplage n'est actif.
- Les couplages actifs sont conservés après RESET et changement de mode de fonctionnement.
- À recherche de bloc, les instructions d'activation, de désactivation, d'effacement et de définition du réducteur électronique ne sont ni exécutées ni accumulées, mais sautées.

Variables système du réducteur électronique

Les variables système du réducteur électronique permettent de déterminer, dans le programme pièce, les états courants d'un groupe d'axes EG et, le cas échéant, de déclencher des réactions adéquates.

Les variables système du réducteur électronique sont repérées de la manière suivante :

$AA_EG_ ...

ou

$VA_EG_ ...

Informations complémentaires
Description fonctionnelle Variables système

3.16.4.3 Désactivation du réducteur électronique (EGOFS, EGOFC)

Il existe 3 variantes pour désactiver un groupe d'axes EG actif.

Programmation

Variante 1 :

Syntaxe	Signification
`EGOFS(axe asservi)`	Le réducteur électronique est désactivé. L'axe asservi est freiné jusqu'à l'arrêt. Cette instruction déclenche un arrêt du prétraitement des blocs.

Variante 2 :

Syntaxe	Signification
`EGOFS(axe asservi,axe pilote 1, …,axe pilote 5)`	Ce paramétrage permet de supprimer **de façon sélective** l'influence de certains axes pilotes sur le déplacement de l'axe asservi.

Un axe pilote au moins doit être spécifié. L'influence des axes pilotes indiqués sur l'axe asservi est supprimée. Cette instruction déclenche un arrêt du prétraitement des blocs. S'il reste des axes pilotes actifs, l'axe asservi continue de se déplacer sous leur influence. Lorsque toutes les influences d'axes pilotes sont désactivées de cette manière, l'axe asservi est freiné jusqu'à l'arrêt.

Variante 3 :

Syntaxe	Signification
`EGOFC(broche asservie1)`	Le réducteur électronique est désactivé. La broche asservie continue de tourner à la vitesse qu'elle a à l'instant de la désactivation. Cette instruction déclenche un arrêt du prétraitement des blocs.

Remarque

Cette variante n'est autorisée que pour des broches.

3.16.4.4 Suppression de la définition d'un réducteur électronique (EGDEL)

La définition d'un groupe d'axes EG ne peut être effacée que si celui-ci a été désactivé auparavant.

Programmation

Syntaxe	Signification
`EGDEL(axe asservi)`	La définition du couplage du groupe d'axes est effacée. Vous pouvez à nouveau définir des groupes d'axes avec `EGDEF` jusqu'à ce que le nombre maximal de groupes d'axes actif simultanément soit atteint. Cette instruction déclenche un arrêt du prétraitement des blocs.

3.16.4.5 Avance par tour (G95) / réducteur électronique (FPR)

L'instruction FPR permet également d'indiquer l'axe asservi d'un réducteur électronique en tant qu'axe déterminant l'avance par tours. Dans ce cas, les règles suivantes s'appliquent :

- L'avance dépend de la vitesse de consigne de l'axe asservi du réducteur électronique.
- La vitesse de consigne est calculée à partir des vitesses des broches pilotes et des axes pilotes à valeur modulo (qui ne sont pas des axes à interpolation) et des facteurs de couplage correspondants.
- Les vitesses des axes pilotes linéaires et des axes pilotes sans valeur modulo ainsi que les déplacements forcé de l'axe asservi ne sont pas pris en considération.

3.16.5 Broche synchrone

En mode de "broche synchrone", il y a une broche pilote (BP) et une broche asservie (BA), qui forment la **paire de broches synchrones**. Quand le couplage est activé (mode "broche synchrone"), la broche asservie suit les déplacements de la broche pilote, en fonction du lien fonctionnel qui a été défini auparavant.

Les paires de broches synchrones peuvent être aussi bien configurées de manière fixe pour chaque machine à l'aide de paramètres machine spécifiques à un canal ou définies en fonction de l'application par le programme pièce de la CNC. Deux paires de broches synchrones sont utilisables simultanément par canal CN.

A partir du programme pièce, le couplage peut être

- défini ou modifié,
- activé,
- désactivé,
- effacé

.

De plus, en fonction de la version du logiciel,

- les conditions de synchronisme peuvent être contrôlées,
- le mode de changement de bloc peut être modifié,
- le type de couplage, soit couplage par valeur de consigne soit couplage par valeur réelle, peut être sélectionné ou le décalage angulaire entre broche pilote et broche asservi peut être prescrit

- à l'activation du couplage, une programmation précédente de la broche asservie peut être reprise
- un écart de synchronisme mesuré ou encore déjà connu peut être corrigé

.

3.16.5.1 Broche synchrone : Programmation (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC)

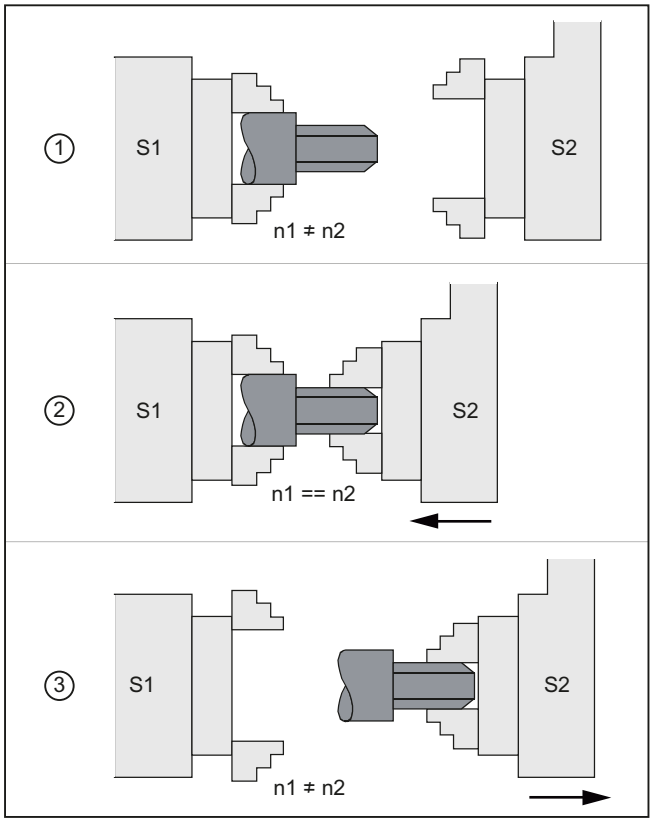
La fonction "Broche synchrone" permet un déplacement à vitesse de rotation synchrone de la broche asservie (BA) et de la broche pilote (BP) avec un rapport de transmission programmable.

Cette fonction dispose des modes suivants :

- Synchronisme de vitesse de rotation ($n_{BA} = n_{BP}$)
- Synchronisme de position ($\phi_{BA} = \phi_{BP}$)
- Synchronisme de position avec décalage angulaire ($\phi_{BA} = \phi_{BP}$)

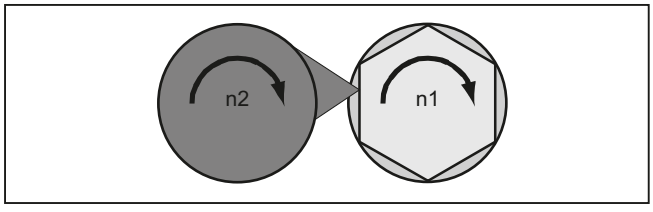
Exemples d'application :

- Transfert de pièce à la volée, par exemple, pour l'usinage de la face arrière, rapport de transmission : 1:1



① Synchronisation de la vitesse de rotation
② Transfert de la pièce
③ Usinage de la face arrière

- Usinage de plusieurs faces (polygonage), synchronisme de vitesse de rotation, rapport de transmission : n_1:n_2



Syntaxe

```
COUPDEF(<BA>,<BP>,<NBA>,<DBP>,<Changement de bloc>,<Mode de
couplage>)
COUPON(<BA>,<BP>,<POSBA>)
COUPONC(<BA>,<BP>)
COUPOF(<BA>,<BP>,<POSBA>,<POSBP>)
COUPOFS(<BA>,<BP>)
COUPOFS(<BA>,<BP>,<POSBA>)
```

```
COUPRES(<BA>,<BP>)
COUPDEL(<BA>,<BP>)
WAITC(<BA>,<Changement de bloc>,<BP>,<Changement de bloc>)
```

Remarque

Notation abrégée

Les instructions COUPOF, COUPOFS, COUPRES et COUPDEL permettent l'utilisation d'une notation abrégée sans indication de la broche pilote.

Signification

COUPDEF :	Définition/modification spécifique à l'utilisateur du couplage
COUPON :	activer le couplage. La broche asservie se synchronise sur la broche pilote à partir de la vitesse de rotation actuelle
COUPONC :	Prise en charge du couplage à l'activation par la programmation précédente de M3 S... ou M4 S.... Une vitesse de rotation différentielle de la broche asservie est aussitôt prise en charge.
COUPOF :	Désactiver le couplage. • avec changement de bloc immédiat : COUPOF(<S2>,<S1>) • Changement de bloc seulement après dépassement de la ou des positions de coupure <POSBA> ou <POSBP> : COUPOF(<S2>,<S1>,<POSBA>) COUPOF(<S2>,<S1>,<POSBA>,<POSBP>)
COUPOFS :	Désactivation d'un couplage avec arrêt de la broche asservie Changement de bloc aussi rapide que possible avec changement de bloc immédiat : COUPOFS(<S2>,<S1>) Changement de bloc uniquement après le passage de la position d'arrêt : COUPOFS(<S2>,<S1>,<POSBA>)
COUPRES :	Remettre à zéro les paramètres de couplage sur les PM et SD configurés
COUPDEL :	effacer le couplage défini par l'utilisateur
WAITC :	Attente de la condition de synchronisme (NOC sont supprimés sur l'interpolateur au changement de bloc)
<BA> :	Désignation de la broche asservie
Paramètres optionnels :	
<BP> :	Désignation de la broche pilote Indication avec numéro de broche : par ex. S2, S1
<NBA>, <DBP> :	Rapport de transmission entre BA et BP. <NBA> / <DBP> = numérateur / dénominateur Réglage par défaut : <NBA> / <DBP> = 1.0 ; indication facultative du dénominateur

<Changement de bloc> :	Mode de changement de bloc Le changement de bloc s'effectue :	
	"NOC"	sur le champ,
	"FINE"	lorsque le "Synchronisme fin" est atteint
	"COARSE"	lorsque le "Synchronisme grossier" est atteint
	"IPOSTOP"	lorsque IPOSTOP est atteint, c.-à-d. quand la valeur de consigne du synchronisme est atteinte (paramétrage par défaut)
	Le mode de changement de bloc est modal.	
<Mode de couplage> :	Type de couplage : couplage entre BA et BP	
	"DV"	Couplage par la valeur de consigne (préréglage)
	"AV"	Couplage par valeur réelle
	"VV"	Couplage de vitesse
	Le type de couplage est modal.	
<POSBA> :	décalage angulaire entre broche pilote et broche asservie	
	Plage de valeurs :	0°... 359,999°
<POSBA>,<POSBP> :	Positions d'arrêt de broche asservie et de broche pilote "Le changement de bloc est débloqué après le passage par POS_{BA}, POS_{BP}."	
	Plage de valeurs :	0°... 359,999°

Exemples

Utilisation de la broche pilote et de la broche asservie

Code de programme	**Commentaire**
	Broche pilote = broche maître = broche 1
	Broche asservie = broche 2
N05 M3 S3000 M2=4 S2=500	La broche pilote tourne à 3000 tr/min, la broche asservie tourne à 500 tr/min.
N10 COUPDEF(S2,S1,1,1,"NOC","Dv")	Définition du couplage (peut également être configuré).
...	
N70 SPCON	Placement de la broche pilote en asservissement de position (contrôle de consigne)
N75 SPCON(2)	Broche asservie intégrée dans la régulation de position.
N80 COUPON(S2,S1,45)	Couplage au vol sur position d'offset = 45 degrés.
...	
N200 FA[S2]=100	Vitesse de positionnement = 100 degrés/min
N205 SPOS[2]=IC(-90)	Déplacement forcé de 90 degrés dans le sens négatif.
N210 WAITC(S2,"Fine")	Attente du synchronisme "fin".
N212 G1 X... Y... F...	Usinage
...	

Code de programme	Commentaire
N215 SPOS[2]=IC(180)	Déplacement forcé de 180 degrés dans le sens positif.
N220 G4 S50	Durée d'attente = 50 tours de la broche maître
N225 FA[S2]=0	Activation de la vitesse configurée (PM).
N230 SPOS[2]=IC(-7200)	20 tours. Déplacement dans le sens négatif à la vitesse configurée.
...	
N350 COUPOF(S2,S1)	Découplage au vol, S=S2=3000
N355 SPOSA[2]=0	Arrêt de la broche asservie à 0 degrés.
N360 G0 X0 Y0	
N365 WAITS(2)	Attendre la broche 2.
N370 M5	Arrêter la broche asservie.
N375 M30	

Programmation d'une vitesse de rotation différentielle

Code de programme	Commentaire
	Broche pilote = broche maître = broche 1
	Broche asservie = broche 2
N01 M3 S500	Broche pilote tourne à 500 tr/min.
N02 M2=3 S2=300	Broche asservie tourne à 300 tr/min.
...	
N10 G4 F1	Temporisation de la broche maître.
N15 COUPDEF (S2,S1,-1)	Facteur de couplage avec un rapport de transmission de -1:1
N20 COUPON(S2,S1)	Activer le couplage. La vitesse de rotation de la broche asservie est définie à partir de la vitesse de rotation de la broche pilote et du facteur de couplage.
...	
N26 M2=3 S2=100	Programmation d'une vitesse de rotation différentielle.

Exemples de prise en charge d'un déplacement à la vitesse de rotation différentielle

Activer le 1er couplage avec COUPON, alors que la broche asservie a déjà été programmée

Code de programme	Commentaire
	Broche pilote = broche maître = broche 1
	Broche asservie = broche 2
N05 M3 S100 M2=3 S2=200	La broche pilote tourne à 100 tr/min, la broche asservie tourne à 200 tr/min.
N10 G4 F5	Temporisation = 5 secondes de la broche maître
N15 COUPDEF(S2,S1,1)	Le rapport de transmission BA par rapport à BP est de 1,0 (réglage par défaut).
N20 COUPON(S2,S1)	Couplage au vol sur la broche pilote.
N10 G4 F5	Broche asservie tourne à 100 tr/min.

Activer le 2e couplage avec COUPON, alors que la broche asservie a déjà été programmée

Code de programme	Commentaire
	Broche pilote = broche maître = broche 1
	Broche asservie = broche 2
N05 M3 S100 M2=3 S2=200	La broche pilote tourne à 100 tr/min, la broche asservie tourne à 200 tr/min.
N10 G4 F5	Temporisation = 5 secondes de la broche maître
N15 COUPDEF(S2,S1,1)	Le rapport de transmission BA par rapport à BP est de 1,0 (réglage par défaut).
N20 COUPONC(S2,S1)	Couplage au vol sur la broche pilote et application de la vitesse de rotation précédente à S2.
N10 G4 F5	S2 tourne à 100 tr/min + 200 tr/min = 300 tr/min

Activer le 3e couplage avec COUPON, la broche asservie étant à l'arrêt

Code de programme	Commentaire
	Broche pilote = broche maître = broche 1
	Broche asservie = broche 2
N05 SPOS=10 SPOS[2]=20	Broche asservie S2 en mode de positionnement.
N15 COUPDEF(S2,S1,1)	Le rapport de transmission BA par rapport à BP est de 1,0 (réglage par défaut).
N20 COUPON(S2,S1)	Couplage au vol sur la broche pilote.
N10 G4 F1	Le couplage est fermé, S2 s'arrête à 20 degrés.

Activer le 4e couplage avec COUPONC, la broche asservie étant à l'arrêt

Remarque

Mode de positionnement ou mode axe

Si la broche asservie se trouve en amont du couplage en mode de positionnement ou en mode axe, la broche asservie se comporte de la même façon pour COUPON(<BA>,<BP>) et pour COUPONC(<BA>,<BP>).

Remarque

Broche pilote et mode d'axe

Si la broche pilote se trouve en mode axe avant que le couplage ne soit défini, la valeur limite de vitesse transmise par le paramètre machine suivant agit également après activation du couplage :

PM32000 $MA_MAX_AX_VELO (vitesse d'axe maximale)

Afin d'éviter ce comportement, l'axe doit être commuté en mode broche (M3 S... ou M4 S...) avant la définition du couplage.

Informations complémentaires

Couplage configuré

La BP et la BA sont définies au moyen de paramètres machine pour le couplage configuré. Les broches configurées ne peuvent pas être modifiées dans le programme pièce. Le paramétrage du couplage peut être réalisé dans le programme pièce avec `COUPDEF` (condition : qu'aucune protection en écriture ne soit définie).

Couplage personnalisé

`COUPDEF` permet de définir un nouveau couplage ou de modifier un couplage existant dans le programme pièce. Si un couplage est déjà actif, il doit d'abord être supprimé avec `COUPDEL` avant la définition d'un nouveau couplage.

Un couplage est de façon complète avec :

```
COUPDEF(<BA>,<BP>,<RBA>,<RBP>, comportement de changement de bloc,
mode de couplage)
```

Broche asservie (BA) et broche pilote (BP)

Les noms d'axe de la BA et de la BP définissent le couplage sans ambiguïté. Ils doivent être programmés dans chaque instruction `COUPDEF`. Les autres paramètres de couplage sont modaux et doivent uniquement être programmés en cas de modification.

Exemple :

```
COUPDEF(S2,S1)
```

Rapport de transmission

Le rapport de transmission est indiqué comme rapport de vitesse de rotation entre BA et BP :

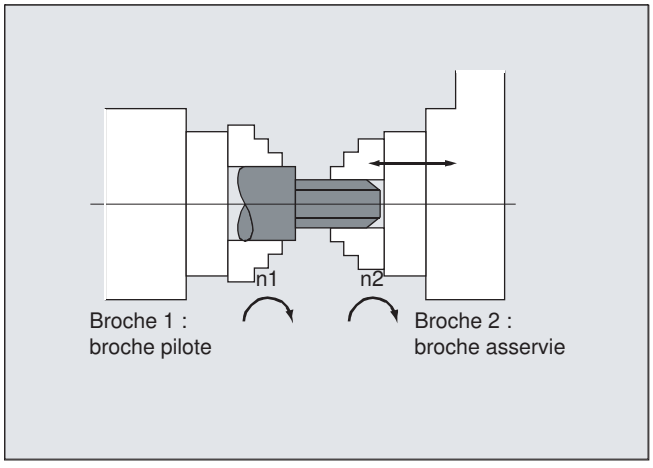
Broche asservie / broche pilote = numérateur / dénominateur

Le numérateur doit être programmé. Le dénominateur ne doit pas obligatoirement être programmé. La valeur par défaut 1.0 est alors définie pour le dénominateur.

Exemple :

Broche asservie S2 et broche pilote S1, rapport de transmission = 1 / 1

```
COUPDEF(S2, S1, 1.0)
```



Remarque

Le rapport de transmission peut également être modifié pendant que le couplage est activé et que les broches sont en rotation.

Mode de changement de bloc NOC, FINE, COARSE, IPOSTOP

Pour la programmation du mode de changement de bloc, la notation abrégée suivante est autorisée :

- `"NO"` : immédiat (préréglage)
- `"FI"` : lorsque le "Synchronisme fin" est atteint
- `"CO"` : lorsque le "Synchronisme grossier" est atteint
- `"IP"` : lorsque IPOSTOP est atteint, c.-à-d. quand la valeur de consigne du synchronisme est atteinte

Type de couplage

Remarque

La modification du type de couplage ne peut se faire que lorsque le couplage est désactivé.

Activation du mode synchrone COUPON, <POSBA>

- Activation du couplage avec un décalage angulaire quelconque entre BP et BA :
  - `COUPON(S2,S1)`
  - `COUPON(S2)`
- Activation du couplage avec un décalage angulaire `<POSBA>`
  `<POSBA>` se rapporte à la position 0° de la broche pilote dans le sens de rotation positif
  Plage de valeurs `<POSBA>` : 0°... 359,999°
  - `COUPON(S2,S1,30)`

Remarque

Le décalage angulaire peut aussi être modifié alors que le couplage est actif.

Positionnement de la broche asservie

Des BA peuvent aussi être positionnées dans la plage ±180° indépendamment de la BP quand le couplage de broches synchrones est activé.

- Positionnement de la BA avec `SPOS`
  Exemple : `SPOS[2]=IC(-90)`
  Des informations complémentaires sur `SPOS` figurent dans :
  Informations complémentaires
  Positionner des broches (SPOS, SPOSA, M19, M70, WAITS) (Page 121)

Vitesse de rotation différentielle

En mode de régulation de vitesse de rotation et lorsque le couplage de broches synchrones est activé, une vitesse de rotation différentielle résulte d'une correction avec signe d'une vitesse de rotation de BA suite à un déplacement de BP et d'une vitesse de rotation de BA suite à la programmation de broche :

- Couplage de broches synchrones avec `COUPONC`
- `S<BA>=<Vitesse de rotation>` [`M<BA>=<Sens de rotation>]`

Remarque

Autres conditions à prendre en compte

- Avec le sens de rotation `M3` / `M4`, la vitesse de rotation `S...` doit également être reprogrammée.
- La correction d'une vitesse de rotation de broche (`M<Sens de rotation> S<BA>`) par le déplacement de BP en présence d'un couplage de broches synchrones `COUPONC` ne prend effet que lorsque la correction est débloquée.
- La dynamique de la broche pilote doit être restreinte de sorte que les valeurs limites dynamiques de la broche asservie superposée ne soient pas dépassées.

Des informations complémentaires sur la vitesse de rotation différentielle figurent dans :
Informations complémentaires
Description fonctionnelle Axes et broches

Vitesse, accélération : AA, ACC, OVRA, VELOLIMA

La programmation de la vitesse et de l'accélération axiales d'une broche asservie s'effectue avec :

- `FA[SPI(S<n>)]` ou `FA[S<n>]` (vitesse axiale)
- `ACC[SPI(S<n>)]` ou `ACC[S<n>]` (accélération axiale)
- `OVRA[SPI(S<n>)]` ou `OVRA[S<n>]` (correction axiale)
- `VELOLIMA[SPI(S<n>)]` ou `VELOLIMA[S<n>]` (augmentation ou réduction de la vitesse axiale)

Avec `<n>` = 1, 2, 3, ... (Numéro des broches asservies)

Informations complémentaires
Régulation d'avance (Page 108)

Remarque

Une réduction ou augmentation de l'à-coup axial maximal n'a pas d'effet sur les broches.

Des informations complémentaires sur la dynamique axiale figurent dans :
Informations complémentaires
Description fonctionnelle Axes et broches

Comportement programmable de changement de bloc WAITC

`WAITC` permet de prédéfinir le comportement de changement de bloc en fonction de différentes conditions de synchronisme (grossier, fin, `IPOSTOP`), p. ex. après une modification des paramètres de couplage ou des opérations de positionnement. Si aucune condition de

synchronisme n'est spécifiée, le comportement de changement de bloc indiqué dans la définition de COUPDEF est appliqué.

Exemples

- Attendre que la condition de synchronisme définie dans FINE soit atteinte pour la broche asservie S2 et celle définie dans COARSE pour la broche asservie S4 : WAITC(S2,"FINE",S4,"COARSE")
- Attente de l'atteinte de la condition de synchronisme définie dans COUPDEF : WAITC( )

Désactivation du couplage COUPOF

COUPOF permet de prédéfinir le comportement de désactivation du couplage :

- Désactivation du couplage avec changement de bloc immédiat :
  - COUPOF(S2,S1) (avec indication de la broche pilote)
  - COUPOF(S2) (sans indication de la broche pilote)
- Désactivation du couplage après le passage de la position d'arrêt. Le changement de bloc a lieu après le passage de la position d'arrêt.
  - COUPOF(S2,S1,150) (position d'arrêt BA : 150°)
  - COUPOF(S2,S1,150,30) (position d'arrêt BA : 150°, BP : 30°)

Désactivation du couplage avec arrêt de la broche asservie COUPOFS

COUPOFS permet de prédéfinir le comportement de désactivation du couplage avec arrêt de la broche asservie :

- Désactivation du couplage avec arrêt de la broche asservie et changement de bloc immédiat :
  - COUPOFS(S2,S1) (avec indication de la broche pilote)
  - COUPOFS(S2) (sans indication de la broche pilote)
- Désactivation du couplage après le passage de la position d'arrêt avec arrêt de la broche asservie. Le changement de bloc a lieu après le passage de la position d'arrêt.
  - COUPOFS(S2,S1,150) (position d'arrêt BA : 150°)

Effacement de couplages COUPDEL

COUPDEL permet de supprimer le couplage :

- COUPDEL(S2,S1) (avec indication de la broche pilote)
- COUPDEL(S2) (sans indication de la broche pilote)

Remise à zéro des paramètres de couplage COUPRES

COUPRES active les valeurs du couplage définies dans les paramètres machine et des données de réglage :

- COUPRES(S2,S1) (avec indication de la broche pilote)
- COUPRES(S2) (sans indication de la broche pilote)

Variables système

- État de couplage courant de la broche asservie
  La lecture de l'état de couplage actuel d'une broche asservie peut s'effectuer codée sur bits via :
  `<Wert> = $AA_COUP_ACT[<FS>]`

Bit	<Valeur>	Signification
-	0	Aucun couplage activé
2	4	le couplage de broches synchrones est activé
Remarque • Toutes les autres valeurs se rapportent au mode axe • Si la broche est une broche asservie de plusieurs couplages, la valeur fournie en retour est la somme des états de couplage de tous les couplages.		

- Décalage angulaire courant
  La lecture du décalage angulaire actuel entre la broche asservie et la broche pilote peut être effectuée via :
  - `$AA_COUP_OFFS[<FS>]` (valeur de consigne du décalage angulaire)
  - `$VA_COUP_OFFS[<FS>]` (valeur effective du décalage angulaire)

  Exemple d'application
  Correction de la différence de décalage angulaire dans le programme CN après l'annulation du mode poursuite :
  Différence de décalage angulaire = décalage angulaire programmé - variable système

Informations complémentaires

Des informations détaillées sur les variables système figurent dans :

Tables de paramètres Variables système

3.16.6 Couplage générique (CP...)

Le "couplage générique" est une fonction de couplage générale dans laquelle sont regroupées toutes les caractéristiques de couplage des types de couplage existants (déplacement conjugué, couplage par valeur pilote, réducteur électronique et broche synchrone).

La fonction permet un type de programmation flexible :

- L'utilisateur peut sélectionner librement les caractéristiques de couplage requises pour son application (principe modulaire).
- Chaque caractéristique de couplage est programmable individuellement.
- Les caractéristiques de couplage d'un couplage défini (par ex. facteur de couplage) sont modifiables.
- L'utilisation ultérieure d'autres caractéristiques de couplage est possible.

- Le système de référence des coordonnées de l'axe asservi (système de coordonnées de base ou système de coordonnées machine) est programmable.
- Certaines caractéristiques de couplage peuvent aussi être programmées dans des actions synchrones.
  Informations complémentaires : Description fonctionnelle Actions synchrones

Remarque

Les appels de couplage antérieurs pour le déplacement conjugué (TRAIL*), le couplage par valeur pilote (LEAD*), les réducteurs électroniques (EG*) et les broches synchrones (COUP*) sont toujours pris en charge au moyen de cycles d'adaptation.

Vue d'ensemble de tous les mots-clés et des caractéristiques de couplage

Le tableau suivant présente une vue d'ensemble de tous les mots-clés du couplage générique et des caractéristiques de couplage ainsi programmables :

Mot-clé	Caractéristique de couplage / signification	Syntaxe
CPDEF	Création d'un module de couplage	CPDEF=(<FAx>)
CPDEL	Suppression d'un module de couplage	CPDEL=(<FAx>)
CPLA	Définition d'un axe pilote	CPLA[<FAx>]=(<LAx>)
CPLDEF	Définition d'un axe pilote et création d'un module de couplage (également possible avec CPDEF + CPLA)	CPLDEF[<FAx>]=(<LAx>) ou CPDEF=(<FAx>) CPLA[<FAx>]=(<LAx>)
CPLDEL	Suppression d'un axe pilote d'un module de couplage (également possible avec CPDEL + CPLA)	CPLDEL[<FAx>]=(<LAx>) ou CPDEL=(<FAx>) CPLA[<FAx>]=(<LAx>)
CPON	Activation d'un module de couplage	CPON=(<FAx>)
CPOF	Désactivation d'un module de couplage	CPOF=(<FAx>)
CPLON	Activation d'un axe pilote d'un module de couplage	CPLON[<FAx>]=<LAx>
CPLOF	Désactivation d'un axe pilote d'un module de couplage	CPLOF[<FAx>]=<LAx>
CPLNUM	Numérateur du facteur de couplage	CPLNUM[FAx,LAx]=<valeur>
CPLDEN	Dénominateur du facteur de couplage	CPLDEN[FAx,LAx]=<valeur>
CPLCTID	Numéro de la table de courbes	CPLCTID[FAx,LAx]=<valeur>

Mot-clé	Caractéristique de couplage / signification	Syntaxe		
CPLSETVAL	Référence du couplage	CPLSETVAL[FAx,LAx]="<référence du couplage>"		
		"<référence du couplage>" :	"CMDPOS"	Couplage par valeur de consigne
			"CMDVEL"	Couplage de vitesse
			"ACTPOS"	Couplage par valeur réelle
CPFRS	Système de référence des coordonnées	CPFRS[FAx]="<référence des coordonnées>"		
		"<référence des coordonnées>" :	"BCS"	Système de coordonnées de base
			"MCS"	Système de coordonnées machine
CPBC	Critère de changement de bloc	CPBC[FAx]="<critère de changement de bloc>"		
		"<critère de changement de bloc>" :	"NOC"	Le changement de bloc s'effectue indépendamment de l'état du couplage.
			"IPOSTOP"	Le changement de bloc s'effectue lors de la synchronisation de la valeur de consigne.
			"COARSE"	Le changement de bloc s'effectue lors de la synchronisation "grossière" de la valeur réelle.
			"FINE"	Le changement de bloc s'effectue lors de la synchronisation "fine" de la valeur réelle.
CPFPOS + CPON	Position synchrone de l'axe asservi lors de l'activation	CPON=FAx CPFPOS[FAx]=<valeur>		
CPLPOS + CPON	Position synchrone de l'axe pilote lors de l'activation	CPLPOS[FAx,LAx]=<valeur>		

Mot-clé	Caractéristique de couplage / signification	Syntaxe		
CPFMSON	Mode de synchronisation	CPFMSON[FAx]="<mode de synchronisation>"		
		"<mode de synchronisation>":	"CFAST"	Le couplage est fermé avec optimisation du temps.
			"CCOARSE"	Le couplage n'est activé que lorsque la position de l'axe asservi requise conformément à la loi de couplage se trouve dans la zone de la position actuelle de l'axe asservi.
			"NTGT"	L'entredent suivant est accosté avec optimisation du temps.
			"NTGP"	L'entredent suivant est accosté avec optimisation du parcours.
			"NRGT"	Le segment suivant est accosté avec optimisation du temps en fonction du rapport entre le nombre de spires et le nombre de dents.
			"NRGP"	Le segment suivant est accosté avec optimisation du parcours en fonction du rapport entre le nombre de spires et le nombre de dents.
			"ACN"	Seulement pour les axes rotatifs ! L'axe rotatif accoste la position synchrone dans le sens négatif de l'axe. La synchronisation s'effectue aussitôt.
			"ACP"	Seulement pour les axes rotatifs ! L'axe rotatif accoste la position synchrone dans le sens positif de l'axe. La synchronisation s'effectue aussitôt.
			"DCT"	Seulement pour les axes rotatifs ! L'axe rotatif accoste la position synchrone programmée avec optimisation du temps. La synchronisation s'effectue aussitôt.
			"DCP"	Seulement pour les axes rotatifs ! L'axe rotatif accoste la position synchrone programmée avec optimisation du

Mot-clé	Caractéristique de couplage / signification	Syntaxe		
				parcours. La synchronisation s'effectue aussitôt.
CPFMON	Comportement de l'axe asservi lors de l'activation	CPFMON[FAx]="<comportement à l'enclenchement>"		
		"<comportement à l'enclenchement>":	"STOP"	Seulement pour les broches ! Un mouvement actif de la broche asservie est arrêté avant l'enclenchement.
			"CONT"	Seulement pour les broches et les axes de l'exécution principale ! Le mouvement actuel de l'axe / la broche asservi(e) est pris en compte dans le couplage en tant que mouvement de départ.
			"ADD"	Seulement pour les broches ! Les composantes de déplacement du couplage agissent en plus du déplacement forcé actuel, c'est-à-dire que le mouvement actuel de l'axe / la broche asservi(e) est conservé en tant que déplacement forcé.
CPFMOF	Comportement de l'axe asservi lors d'une désactivation complète	CPFMOF[FAx]="<comportement à la désactivation>"		
		"<comportement à la désactivation>":	"STOP"	Arrêt de l'axe / la broche asservi(e). Un déplacement forcé actif est freiné jusqu'à l'immobilisation. Le couplage est ouvert ensuite.
			"CONT"	Seulement pour les broches et les axes de l'exécution principale ! La broche asservie continue de tourner à la vitesse qu'elle a à l'instant de la désactivation.
CPFPOS + CPOF	Position de coupure de l'axe asservi lors de la désactivation	CPOF=(FAx) CPFPOS[FAx]=<valeur>		

Mot-clé	Caractéristique de couplage / signification	Syntaxe		
CPMRESET	Comportement du couplage lors d'un RESET	CPMRESET[FAx]="<comportement reset>"		
		"<comportement reset>" :	"NONE"	L'état actuel du couplage est conservé.
			"ON"	Si le module de couplage correspondant est créé, le couplage est activé. Toutes les relations définies pour l'axe pilote sont activées. Cela se produit également si toutes ou des parties de ces relations de l'axe pilote sont déjà actives, c'est-à-dire qu'une resynchronisation a lieu même en présence d'un couplage entièrement activé.
			"OF"	Un déplacement forcé actif est freiné jusqu'à l'immobilisation. Le couplage est désactivé ensuite. Si le module de couplage correspondant a été créé sans définition explicite (CPDEF), le module de couplage est supprimé. Dans le cas contraire, il est tout de même créé, c'est-à-dire qu'il peut tout de même être utilisé.
			"OFC"	Seulement possible pour les broches ! La broche asservie continue de tourner à la vitesse qu'elle a à l'instant de la désactivation. Le couplage est désactivé. Si le module de couplage correspondant a été créé sans définition explicite (CPDEF), le module de couplage est supprimé. Dans le cas contraire, il est tout de même créé, c'est-à-dire qu'il peut tout de même être utilisé.
			"DEL"	Un déplacement forcé actif est freiné jusqu'à l'immobilisation. Le couplage est ensuite désactivé puis supprimé.
			"DELC"	Seulement possible pour les broches ! La broche asservie continue de tourner à la vitesse qu'el-

Mot-clé	Caractéristique de couplage / signification	Syntaxe		
				le a à l'instant de la désactivation. Le couplage est désactivé puis supprimé.
CPMSTART	Comportement du couplage au lancement du programme pièce	CPMSTART[FAx]="<comportement au lancement>"		
		"<comportement au lancement>" :	"NONE"	L'état actuel du couplage est conservé.
			"ON"	Couplage activé. Toutes les relations définies pour l'axe pilote sont activées. Cela se produit également si toutes ou des parties de ces relations de l'axe pilote sont déjà actives, c'est-à-dire qu'une resynchronisation a lieu même en présence d'un couplage entièrement activé.
			"OF"	Le couplage est désactivé. Si le module de couplage correspondant a été créé sans définition explicite (CPDEF), le module de couplage est supprimé. Dans le cas contraire, il est tout de même créé, c'est-à-dire qu'il peut tout de même être utilisé.
			"DEL"	Le couplage est désactivé puis supprimé.
CPMPRT	Comportement du couplage au lancement du programme pièce lors d'une recherche en mode test du programme	CPMPRT[FAx]="<comportement au lancement>"		
		"<comportement au lancement>" :	voir CPMSTART	
CPLINTR	Valeur de décalage pour la valeur d'entrée d'un axe pilote	CPLINTR[FAx,LAx]=<valeur>		
CPLINSC	Facteur d'échelle pour la valeur d'entrée d'un axe pilote	CPLINSC[FAx,LAx]=<valeur>		
CPLOUTTR	Valeur de décalage pour la valeur de sortie d'un couplage	CPLOUTTR[FAx,LAx]=<valeur>		
CPLOUTSC	Facteur d'échelle pour la valeur de sortie d'un couplage	CPLOUTSC[FAx,LAx]=<valeur>		
CPSYNCOP	Valeur de seuil pour la synchronisme "grossière" de position	CPSYNCOP[FAx]=<valeur>		
CPSYNFIP	Valeur de seuil pour la synchronisation "fine" de position	CPSYNFIP[FAx]=<valeur>		
CPSYNCOP2	Deuxième valeur de seuil pour la synchronisation "grossière" de position	CPSYNCOP2[FAx]=<valeur>		

Mot-clé	Caractéristique de couplage / signification	Syntaxe		
CPSYNFIP2	Deuxième valeur de seuil pour la synchronisation "fine" de position	CPSYNFIP2[FAx]=<valeur>		
CPSYNCOV	Valeur de seuil pour la synchronisation "grossière" de la vitesse	CPSYNCOV[FAx]=<valeur>		
CPSYNFIV	Valeur de seuil pour la synchronisation "fine" de la vitesse	CPSYNFIV[FAx]=<valeur>		
CPMBRAKE	Comportement de l'axe asservi lors de certains signaux et ordres d'arrêt	CPMBRAKE[FAx]=<valeur codée sur bits>		
CPMVDI	Comportement de l'axe asservi en réaction à certains signaux d'interface CN/AP	CPMVDI[FAx]=<valeur codée sur bits>		
CPMALARM	Inhibition des générations d'alarme spéciales relatives au couplage	CPMALARM[FAx]=<valeur codée sur bits>		
CPSETTYPE	Type de couplage	CPSETTYPE[FAx]="<type de couplage>"		
		"<type de couplage>":	"CP"	Libre programmabilité
			"TRAIL"	Type de couplage "déplacement conjugué"
			"LEAD"	Type de couplage "couplage par valeur pilote"
			"EG"	Type de couplage "réducteur électronique"
			"COUP"	Type de couplage "broche synchrone"

FAx: Axe/broche asservi(e)

LAx: Axe/broche pilote

Remarque

Les caractéristiques de couplage qui ne sont pas programmées de manière explicite (dans le programme pièce ou dans les actions synchrones) prennent effet avec leur réglage par défaut.

En fonction du paramétrage du mot-clé CPSETTYPE, des caractéristiques de couplage préréglées peuvent aussi prendre effet à la place des réglages par défaut (CPSETTYPE="CP").

Informations complémentaires

Pour obtenir des informations détaillées sur le couplage générique, voir :

- Description fonctionnelle Axes et broches

3.16.7 Positionnement tangentiel

3.16.7.1 Définir le couplage (TANG)

La procédure TANG(...) prédéfinie permet de définir un couplage tangentiel entre un axe rotatif en tant qu'axe asservi et deux axes géométriques en tant qu'axes pilotes. L'axe asservi est alors aligné en continu sur la tangente à la trajectoire des axes pilotes.

Remarque

facteur de couplage

Un facteur de couplage de 1 ne doit pas obligatoirement être programmé de manière explicite. Un facteur de couplage de -1 provoque la rotation du sens de l'axe tangentiel.

Syntaxe

```
TANG(<Axe asservi>, <Axe pilote_1>, <Axe pilote_2>, <Facteur de
couplage>, <Système de coordonnées>, <Optimisation>)
```

Signification

`TANG(...)` :	Définition d'un couplage tangentiel		
`<Axe asservi>` :	Nom d'axe de l'axe asservi (axe rotatif)		
	Type de données :	AXIS	
	Plage de valeurs :	Noms d'axes de canal	
`<Axe pilote_1>` `<Axe pilote_2>` :	Noms d'axes des axes pilotes (axes géométriques) [1]		
	Type de données :	AXIS	
	Plage de valeurs :	Noms des axes géométriques du canal	
`<Facteur de couplage>` :	Facteur n de variation angulaire de l'axe asservi pour modifier la tangente à la trajectoire des axes pilotes : Variation angulaire$_{\text{Axe asservi}}$ = Variation angulaire$_{\text{Tangente à la trajectoire}}$ * n		
	Type de données :	REAL	
	Valeur par défaut :	1,0	
`<Système de coordonnées>` :	Système de coordonnées actif [2]		
	Type de données :	CHAR	
	Valeur :	`"B"` :	Système de coordonnées de base (valeur par défaut)
		`"W"` :	Système de coordonnées pièces (non disponible)

<Optimisation> :	Type d'optimisation		
	Type de données :	CHAR	
	Valeur :	"S" :	Standard (par défaut) La dynamique de l'axe rotatif n'a aucun effet sur les axes pilotes. Si la dynamique de l'axe rotatif est plus importante que nécessaire pour l'asservissement, cette méthode est suffisamment précise. Si la dynamique de l'axe rotatif n'est pas suffisamment importante pour suivre la modification des tangentes à la trajectoire, l'orientation de l'axe rotatif s'écarte de l'orientation de consigne le long d'une trajectoire d'arrondissement non définie.
		"P" :	La dynamique de l'axe rotatif est prise en compte lors de la planification de trajectoire des axes pilotes. A cet effet, deux paramètres supplémentaires doivent être indiqués lors de l'activation du couplage tangentiel avec TANGON() : • Trajectoire d'arrondissement • Tolérance angulaire Voir chapitre "Activer le couplage (TANGON) (Page 976)" **Remarque** En relation avec les transformations cinématiques, il est recommandé d'utiliser le type d'optimisation "P".

Remarque

Les valeurs par défaut ne doivent pas obligatoirement être programmées de manière explicite.

[1] **Remarque**

Comme axes pilotes du couplage tangentiel, il est nécessaire d'utiliser les axes géométriques qui effectuent un retrait selon la trajectoire programmée dans le système de coordonnées machine (SCM) par rapport à la position de base de la machine. Par exemple, sur une fraiseuse avec tête orientable, si le cycle d'orientation CYCLE800 est utilisé, l'interpolation a lieu, selon la configuration du cycle, dans le SCP par ex. avec les axes géométriques X et Y. Cependant, le couplage tangentiel doit être défini avec comme axes pilotes les axes géométriques qui effectuent un retrait selon la trajectoire programmée dans le SCM. De plus, il convient d'utiliser comme axes pilotes les axes géométriques à l'état **non orienté** de la machine.

[2] **Remarque**

Le système de coordonnées de base (SCB) ne doit pas être pivoté par rapport au SCM. Si le SCB est pivoté par ex. via l'instruction ROT ou via le cycle d'orientation CYCLE800, le positionnement tangentiel ne fonctionne plus correctement.

3.16.7.2 Activer la génération de bloc intermédiaire (TLIFT)

Si la modification de la tangente de l'axe asservi dépasse à un point donné de la trajectoire programmée des axes pilotes la valeur limite configurée dans le paramètre machine PM37400 $MA_EPS_TLIFT_TANG_STEP, la suite de la planification de la trajectoire s'effectue en fonction du comportement aux angles réglé. Sans l'utilisation de la procédure TLIFT(...) prédéfinie, le déplacement s'effectue en fonction du comportement d'arrondissement programmé en relation avec TANG(...) (Page 973) et TANGON(...) (Page 976).

Activer la génération de bloc intermédiaire

Avec la programmation de TLIFT(...) après TANG(...), le prétraitement insère un bloc intermédiaire généré automatiquement par la commande en cas de détection d'un angle à cet endroit de la trajectoire.

Lors de l'exécution du programme, les axes pilotes sont alors arrêtés lorsque le bloc intermédiaire est atteint. Dans le bloc intermédiaire, l'axe asservi est pivoté avec la dynamique d'axe maximale en direction de la tangente à la trajectoire du bloc suivant. Ensuite, le déplacement des axes pilotes selon la trajectoire programmée se poursuit.

Désactiver la génération de bloc intermédiaire

Pour désactiver la génération de bloc intermédiaire, le couplage tangentiel doit être de nouveau défini au moyen de TANG(...), mais sans activation consécutive de la génération de bloc intermédiaire au moyen de TLIFT(...).

Syntaxe

```
TLIFT(<Axe asservi>)
```

Signification

`TLIFT(...)` :	Activation de la détection d'angle avec calcul de bloc intermédiaire	
`<Axe asservi>` :	Nom d'axe de l'axe asservi (axe rotatif)	
	Type de données :	AXIS
	Plage de valeurs :	Noms d'axes de canal

Vitesse de rotation de l'axe asservi

Axe d'interpolation

Si l'axe asservi se déplaçait déjà en tant qu'axe d'interpolation avant l'activation du couplage tangentiel, le mouvement de rotation s'effectue dans le bloc intermédiaire en tant qu'axe d'interpolation.

En indiquant le rayon de référence avec `FGREF[<Axe>]=0.001`, le mouvement de rotation s'effectue à la vitesse d'axe maximale paramétrée :

PM32000 $MA_MAX_AX_VELO[<Axe asservi>]

Axe de positionnement

Si l'axe asservi ne se déplaçait pas encore en tant qu'axe d'interpolation avant l'activation du couplage tangentiel, la rotation s'effectue dans le bloc intermédiaire en tant qu'axe de positionnement.

Le mouvement de rotation s'effectue alors à la vitesse d'axe de positionnement paramétrée :

PM32060 $MA_POS_AX_VELO[<Axe asservi>]

3.16.7.3 Activer le couplage (TANGON)

La procédure TANGON(...) prédéfinie permet d'activer un couplage tangentiel défini au préalable avec TANG(...) (Page 973). L'axe asservi est ensuite aligné en continu sur la tangente à la trajectoire lors du déplacement suivant des axes pilotes.

Angle de l'axe asservi

L'angle que prend l'axe asservi par rapport à la tangente à la trajectoire dépend du rapport de transmission indiqué dans TANG(...), de l'angle de décalage configuré dans le paramètre machine PM37402 $MA_TANG_OFFSET et de l'angle de décalage indiqué avec TANGON(...), qui s'y ajoute.

Optimisation "P"

Si la valeur "P" a été indiquée comme paramètre d'optimisation lors de la définition du couplage tangentiel (TANG(...)), le paramètre "Trajectoire d'arrondissement" et, de manière facultative, le paramètre "Tolérance angulaire" doivent être indiqués lors de l'activation du couplage.

Si la valeur 0 est spécifiée comme tolérance angulaire, seul le paramètre "Trajectoire d'arrondissement" est effectif.

Si une valeur supérieure à 0 est spécifiée comme tolérance angulaire, la trajectoire d'arrondissement effective est la valeur la plus faible entre la trajectoire d'arrondissement paramétrée et la trajectoire d'arrondissement due à la tolérance angulaire paramétrée.

Si la dynamique de l'axe asservi n'est pas suffisante pour respecter les conditions paramétrées, la vitesse tangentielle des axes pilotes est réduite en conséquence.

Syntaxe

```
TANGON(<Axe asservi>, <Angle de décalage>, <Trajectoire
d'arrondissement>, <Tolérance angulaire>)
```

Signification

`TANGON(...)` :	Activation d'un couplage tangentiel	
`<Axe asservi>` :	Nom d'axe de l'axe asservi (axe rotatif)	
	Type de données :	AXIS
	Plage de valeurs :	Noms d'axes de canal

<Angle de décalage> :	Angle de décalage de l'axe asservi par rapport à la tangente à la trajectoire Le point de référence est l'origine de l'axe rotatif.	
	Type de données :	REAL
<Trajectoire d'arrondissement> :	Trajectoire d'arrondissement maximale admissible Si la trajectoire d'arrondissement a augmenté en raison des conditions de dynamique, la vitesse tangentielle des axes pilotes est réduite.	
	Type de données :	REAL
<Tolérance angulaire> :	Tolérance maximale admissible par rapport à l'angle indiqué entre l'origine de l'axe asservi et la tangente à la trajectoire	
	Type de données :	REAL

3.16.7.4 Désactiver le couplage (TANGOF)

La procédure TANGOF(...) prédéfinie permet de désactiver un couplage tangentiel défini avec TANG(...) (Page 973) et activé avec TANGON(...) (Page 976). L'axe asservi n'est ensuite plus aligné sur la tangente à la trajectoire des axes pilotes. Cependant, le couplage de l'axe asservi avec les axes pilotes est maintenu même après la désactivation, ce qui empêche par ex. les fonctions suivantes :

- Changement de plan
- Permutation d'axes géométriques
- Définition d'un nouveau couplage tangentiel pour l'axe asservi

L'annulation complète du lien entre l'axe asservi et les axes pilotes a lieu uniquement après la suppression du couplage via TANGDEL(...) (Page 978).

Programmation

```
TANGOF(<Axe asservi>)
```

Signification

TANGOF(...) :	Désactivation d'un couplage tangentiel	
<Axe asservi> :	Nom d'axe de l'axe asservi (axe rotatif)	
	Type de données :	AXIS
	Plage de valeurs :	Noms d'axes de canal

3.16.7.5 Supprimer un couplage (TANGDEL)

Un couplage tangentiel défini avec TANG(...) (Page 973) est maintenu même après la désactivation du couplage tangentiel avec TANGOF(...) (Page 977). Le couplage tangentiel existant continue alors à empêcher par ex. les fonctions suivantes :

- Changement de plan
- Permutation d'axes géométriques
- Définition d'un nouveau couplage tangentiel pour l'axe asservi

La procédure TANGDEL(...) prédéfinie permet de supprimer le couplage tangentiel existant après la désactivation du couplage tangentiel avec TANGOF(...)

Syntaxe

```
TANGDEL(<Axe asservi>)
```

Signification

TANGDEL(...) :	Suppression d'un couplage tangentiel défini avec TANG()	
	Prise d'effet :	Non modal
<Axe asservi> :	Nom de l'axe asservi dont le couplage tangentiel doit être supprimé	
	Type de données :	AXIS
	Plage de valeurs :	Noms d'axes de canal

Exemples

Changement d'axe pilote

Avant qu'un nouveau couplage tangentiel avec un autre axe pilote ne puisse être défini pour l'axe asservi, le couplage tangentiel existant doit d'abord être supprimé.

Code de programme	Commentaire
N10 **TANG**(A, X, Y, 1)	; Définir un couplage tangentiel pour l'axe asservi A : A avec X et Y
N20 TANGON(A)	; Activer le couplage tangentiel pour l'axe asservi A
N30 X10 Y20	
...	
N80 TANGOF(A)	; Désactiver le couplage tangentiel pour l'axe asservi A
N90 **TANGDEL**(A)	; Supprimer le couplage tangentiel pour l'axe asservi A
...	
N120 **TANG**(A, X, Z)	; Définir un nouveau couplage tangentiel pour l'axe asservi A
N130 TANGON(A)	; Activer le nouveau couplage tangentiel pour l'axe asservi A
...	

Permutation d'axes géométriques

Avant qu'une permutation d'axes géométriques ne puisse être effectuée pour un couplage existant, le couplage doit d'abord être supprimé.

Code de programme	**Commentaire**
N10 GEOAX(2, Y1)	; 2ème axe géométrique = Axe machine Y1
N20 TANG(A, X, Y)	; Définir un couplage tangentiel pour l'axe asservi A
N30 TANGON(A, 90)	; Activer le couplage tangentiel pour l'axe asservi A
N40 G2 F8000 X0 Y0 I0 J50	; Bloc de déplacement
N50 TANGOF(A)	; Désactiver le couplage tangentiel pour l'axe asservi A
N60 **TANGDEL**(A)	; Supprimer le couplage tangentiel pour l'axe asservi A
N70 **GEOAX**(2, Y2)	; 2ème axe géométrique = Axe machine Y2
N80 TANG(A, X, Y)	; Définir un nouveau couplage tangentiel pour l'axe asservi A
N90 TANGON(A, 90)	; Activer le nouveau couplage tangentiel pour l'axe asservi A
...	

3.16.8 Couplage maître/esclave (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS)

Le "couplage maître/esclave" permet d'effectuer :

- le couplage des axes esclaves avec l'axe maître lorsque les axes concernés sont à l'arrêt,
- le couplage et le découplage de broches en mode asservissement de vitesse qui sont **en rotation**,
- la configuration dynamique.

Remarque

Mode Positionnement

Pour les axes et les broches en mode de positionnement, la fermeture et l'ouverture du couplage ne s'opèrent qu'à l'arrêt.

Syntaxe

```
MASLON(<esclave_1>,<esclave_2>...)
MASLOF(<esclave_1>,<esclave_2>...)
MASLOFS(<esclave_1>,<esclave_2>...)
```

Configuration dynamique :

```
MASLDEF(<esclave_1>,<esclave_2>,...,<maître>)
MASLDEL(<esclave_1>,<esclave_2>...)
```

Signification

MASLON :	Activation d'un couplage maître/esclave temporaire	
	<esclave_x>, ... :	axe esclave 1 ... n
MASLOF :	Ouverture d'un couplage maître/esclave actif	
	<esclave_1>, ... :	axe esclave 1 ... n
MASLOFS :	Ouverture du couplage maître/esclave et freinage automatique de la broche asservie (voir Remarque "Comportement au couplage de broches en mode asservissement de vitesse")	
	<esclave_1>, ... :	axe esclave 1 ... n
MASLDEF :	Création/modification du couplage maître/esclave à partir du programme pièce	
	<esclave_1>, ... :	axe esclave 1 ... n
	<maître> :	Axe maître
MASLDEL :	Ouverture du couplage maître/esclave et suppression de la définition du groupe	
	<esclave_1>, ... :	axe esclave 1 ... n
	Remarque : Les définitions maître/esclave configurées dans les paramètres machine sont conservées.	

Remarque

Comportement au couplage de broches en mode asservissement de vitesse

Pour les broches en mode asservissement de vitesse, le comportement au couplage est défini explicitement par MASLON, MASLOF, MASLOFS et MASLDEL dans le paramètre machine suivant :

MD37263 $MA_MS_SPIND_COUPLING_MODE

Avec le réglage par défaut MD37263 = 0, la fermeture et l'ouverture du couplage des axes esclaves s'opèrent uniquement lorsque les axes concernés sont immobilisés. MASLOFS correspond à MASLOF.

Lorsque MD37263 = 1, l'instruction de couplage est exécutée immédiatement, et donc même en mouvement. MASLON ferme immédiatement le couplage et MASLOFS ou MASLOF l'ouvre immédiatement. Les broches esclaves en rotation à cet instant conservent leurs vitesses de rotation pour MASLOF jusqu'à la reprogrammation des vitesses. Au contraire, avec MASLOFS, elles sont automatiquement freinées.

Remarque

L'arrêt implicite du prétraitement des blocs n'est pas exécuté avec MASLOF/MASLOFS . Du fait de l'absence de cet arrêt du prétraitement des blocs, les variables système $P ne fournissent pas de valeurs actualisées pour les axes esclaves jusqu'au moment de la reprogrammation.

Remarque

Pour l'axe esclave, PRESETON permet de synchroniser la valeur réelle sur la même valeur que l'axe maître. Cela exige une brève désactivation du couplage permanent maître/esclave pour que la valeur réelle de l'axe esclave non référencé puisse être définie à la valeur de l'axe maître avec un POWER ON. Le couplage permanent est ensuite rétabli.

Le couplage permanent maître/esclave est activé par le réglage suivant du paramètre machine :
MD37262 $MA_MS_COUPLING_ALWAYS_ACTIVE = 1
Ce réglage n'a pas d'effet sur les instructions de couplage temporaire.

Exemples

Exemple 1 : Forçage des valeurs réelles pour l'axe esclave d'un couplage maître/esclave

Pour un couplage maître/esclave permanent, PRESETON permet de forcer la valeur réelle de l'axe esclave sur la valeur de l'axe maître.

Code de programme	Commentaire
`$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=0`	`; Désactiver le couplage permanent de l'axe esclave`
`NEWCONF`	`; Activer la modification des paramètres machine`
`STOPRE`	
`MASLOF(Y1)`	`; désactiver le couplage temporaire`
`PRESETON(AX2,$VA_IM(M_AX))`	`; Valeur réelle de l'axe esclave = valeur réelle de l'axe maître`
`$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=1`	`; Activer le couplage permanent de l'axe esclave`
`NEWCONF`	`; Activer la modification des paramètres machine`

Exemple 2 : configuration dynamique d'un couplage maître/esclave

Pour que le couplage avec une autre broche puisse se réaliser après une rotation du conteneur d'axes, il convient de supprimer le couplage précédent et sa configuration et de reconfigurer le nouveau couplage.

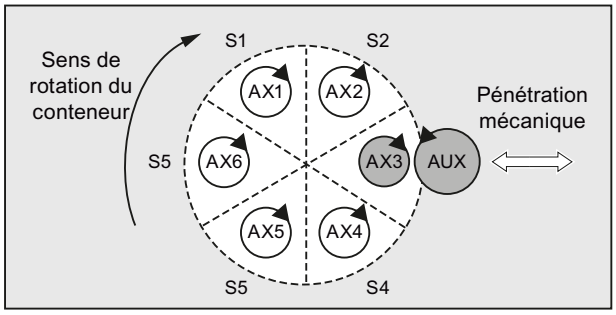


Figure 3-14 Avant la rotation du conteneur d'axes

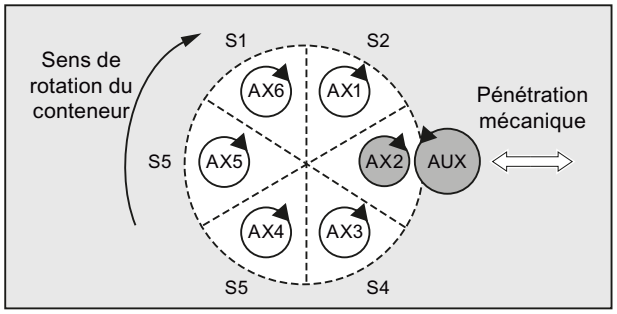


Figure 3-15 Après rotation du conteneur d'axes de la valeur d'un emplacement :

Code de programme	**Commentaire**
`MASLDEF(AUX,S3)`	`; AUX : esclave, S3 : maître = AX3`
`MASLON(AUX)`	`; Activer le couplage`
`M3=3 S3=4000`	`; Rotation maître`
`MASLDEL(AUX)`	`; Séparer et supprimer couplage`
`AXCTSWE(CT1)`	`; Validation de la rotation du conteneur d'axes`
`MASLDEF(AUX,S3)`	`; AUX : esclave, S3 : maître = AX2`

Informations complémentaires

- Description fonctionnelle Technologies
- Description fonctionnelle Fonctions de base

3.17 Actions synchrones

3.17.1 Définition d'une action synchrone

Une action synchrone se définit dans un bloc de programme pièce. Aucune autre instruction que les constituants de l'action synchrone ne doit être programmée dans ce bloc.

Une action synchrone se compose des éléments suivants :

Validité, n° d'identification (en option)	Partie condition (en option)			Partie action lorsque la condition est remplie			Partie action lorsque la condition n'est pas remplie (en option)		
	Fréquence	Instruction G (en option)	Condition	Mot-clé	Instruction G (en option)	Actions	Mot-clé	Instruction G (en option)	Actions
--- [1] ID=<n°> IDS=<n°>	--- [1] WHENEVER FROM WHEN EVERY	G...	Expression logique	DO	G...	Action 1 ... Action n	ELSE	G...	Action 1 ... Action n

[1] non programmée

Syntaxe

```
DO <action 1> ... <action n>
<fréquence> [<fonction G>] <condition> DO <action 1> ... <action n>
ID=<n°> <fréquence> [<fonction G>] <condition> DO <action 1> ...
<action n>
IDS=<n°> <fréquence> [<fonction G>] <condition> DO <action 1> ...
<action n>
IDS=<n°> <fréquence> [<fonction G>] <condition> DO <action 1...n>
ELSE <action 1...n>
```

Plus d'informations

Description fonctionnelle Actions synchrones

3.18 Oscillation

3.18.1 Oscillation asynchrone (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB)

Un axe d'oscillation va et vient, avec une avance donnée, entre deux points d'inversion de sens 1 et 2 jusqu'à désactivation de l'oscillation.

Pendant l'oscillation, d'autres axes peuvent être interpolés librement. Une pénétration continue peut être obtenue par un déplacement de contournage ou avec un axe de positionnement. Il n'existe toutefois **aucun lien** entre l'oscillation et le mouvement de pénétration.

Caractéristiques de l'oscillation asynchrone

- L'oscillation asynchrone agit au-delà des limites du bloc et est spécifique à un axe.
- Une activation de l'oscillation, synchronisée avec le bloc, est assurée par le programme pièce.
- Une interpolation commune de plusieurs axes et une correction de la course d'oscillation ne sont pas possibles.

Programmation

Les instructions suivantes permettent d'activer et d'agir sur l'oscillation asynchrone à partir du programme pièce, en fonction de l'exécution du programme CN.

Les valeurs programmées sont inscrites dans les données de réglage correspondantes pendant l'exécution du programme et restent actives jusqu'à la prochaine modification.

Syntaxe

```
OSP1[<axe>]=<valeur> OSP2[<axe>]=<valeur>
OST1[<axe>]=<valeur> OST2[<axe>]=<valeur>
FA[<axe>]=<valeur>
OSCTRL[<axe>]=(<options de réglage>,<option de désactivation>)
OSNSC[<axe>]=<valeur>
OSE[<axe>]=<valeur>
OSB[<axe>]=<valeur>
OS[<axe>]=1
OS[<axe>]=0
```

Signification

`<axe>` :	Nom de l'axe d'oscillation		
`OS` :	Activer/désactiver l'oscillation		
	Valeur :	1	**Activer** l'oscillation
		0	**Désactiver** l'oscillation
`OSP1` :	Définir la position à partir du point d'inversion de sens 1		

OSP2 :	Définir la position à partir du point d'inversion de sens 2 **Remarque :** si un déplacement relatif est actif, la position est calculée de manière relative par rapport à la dernière position d'inversion correspondante programmée dans le programme CN.		
OST1 :	Définir le temps d'arrêt au point d'inversion de sens 1 en [s]		
OST2 :	Définir le temps d'arrêt au point d'inversion de sens 2 en [s]		
	<valeur> :	-2	l'interpolation est poursuivie sans attente de l'arrêt précis
		-1	attente de l'arrêt précis grossier
		0	attente de l'arrêt précis fin
		>0	attente de l'arrêt précis fin puis écoulement du temps d'arrêt indiqué **Remarque :** l'unité du temps d'arrêt est identique à celle programmée avec G4.
FA :	Définir la vitesse d'avance La vitesse d'avance appliquée ici est celle définie pour l'axe de positionnement. Si aucune vitesse d'avance n'a été définie, c'est la valeur qui figure dans le paramètre machine qui s'applique.		
OSCTRL :	Indique les options d'activation et de désactivation Les valeurs d'option 0 - 3 verrouillent le comportement au point d'inversion à la désactivation. L'une des variantes 0 - 3 peut être sélectionnée. Une combinaison quelconque des paramétrages restant avec la variante sélectionnée est ensuite possible. La combinaison de plusieurs options s'effecue par insertion du signe plus (+).		
	<valeur> :	0	A la désactivation de l'oscillation, arrêt au point d'inversion suivant (paramétrage par défaut) **Remarque :** ceci n'est possible que par réinitialisation des valeurs 1 et 2.
		1	À la désactivation de l'oscillation, arrêt au point d'inversion de sens 1
		2	À la désactivation de l'oscillation, arrêt au point d'inversion de sens 2
		3	À la désactivation de l'oscillation, ne pas accoster de point d'inversion de sens, si aucune passe à lécher n'est programmée
		4	Après les passes à lécher, accoster la position de fin
		8	Si l'oscillation est interrompue par effacement de la distance restant à parcourir, le traitement des passes à lécher doit ensuite être terminé et, le cas échéant, la position finale accostée.
		16	Si l'oscillation est interrompue par effacement de la distance restant à parcourir, la position d'inversion correspondante doit être accostée comme dans le cas de la désactivation.
		32	L'avance modifiée ne prend effet qu'à partir du prochain point d'inversion de sens
		64	FA est égal à 0, FA = 0 : le forçage du déplacement est activé FA est différent de 0, FA <> 0 : la correction de vitesse est activée
		128	Dans le cas d'un axe rotatif DC (course minimale)
		256	Passe à lécher exécutée sous forme de course double (standard). 1=passe à lécher exécutée sous forme de course simple.
		512	Accoster d'abord la position de départ
OSNSC :	Définir le nombre de passes à lécher		

OSE :	Définir la position finale à accoster (dans le SCP) après désactivation de l'oscillation **Remarque :** la programmation de "OSE" active de façon implicite l'option 4 pour "OSCTRL".
OSB :	Définir la position de départ à accoster (dans le SCP) avant activation de l'oscillation La position de départ est accostée avant le point d'inversion de sens 1. Si la position de départ coïncide avec la position d'inversion 1, la position d'inversion 2 est ensuite accostée. Lorsque la position de départ est atteinte, aucun temps d'arrêt n'a d'effet, même lorsque la position de départ coïncide avec la position d'inversion 1, par contre un arrêt précis fin est attendu. La condition de précision réglée est respectée. **Remarque :** afin que la position de départ puisse être accostée, le bit 9 doit être mis à 1 dans la donnée de réglage SD43770 $SA_OSCILL_CTRL_MASK.

Exemples

Exemple 1 : l'axe d'oscillation doit osciller entre deux points d'inversion

L'axe d'oscillation Z doit osciller entre la position 10 et la position 100. Le point d'inversion de sens 1 doit être accosté avec un arrêt précis fin, le point d'inversion de sens 2 avec un arrêt précis grossier. L'avance de l'axe d'oscillation doit être égale à 250. A la fin de l'usinage, 3 passes à lécher doivent être exécutées et l'axe d'oscillation doit atteindre la position finale 200. L'avance pour l'axe de pénétration doit être égale à 1, la fin de la pénétration en direction X doit être atteinte à la position 15.

Code de programme	**Commentaire**
WAITP(X,Y,Z)	; Position de départ.
G0 X100 Y100 Z100	; Commutation en mode d'axe de positionnement.
WAITP(X,Z)	
OSP1[Z]=10 OSP2[Z]=100	; Point d'inversion de sens 1, point d'inversion de sens 2.
OSE[Z]=200	; Position finale.
OST1[Z]=0 OST2[Z]=-1	; Temps d'arrêt sur U1 : Arrêt précis fin
	; Temps d'arrêt sur U2 : arrêt précis grossier
FA[Z]=250 FA[X]=1	; Avance axe d'oscillation, avance axe de pénétration.
OSCTRL[Z]=(4,0)	; Options de réglage.
OSNSC[Z]=3	; 3 passes à lécher.
OS[Z]=1	; Démarrer l'oscillation.
WHEN $A_IN[3]==TRUE DO DELDTG(X)	; Effacement de la distance restant à parcourir.
POS[X]=15	; Position de départ axe X
POS[X]=50	; Position finale axe X.
OS[Z]=0	; Arrêter l'oscillation.
M30	

Remarque

La séquence d'instructions "OSP1[Z]=..." à "OSNCS[Z]=..." peut également être programmée dans un bloc.

Exemple 2 : oscillation avec modification en ligne de la position d'inversion

Les données de réglage nécessaires pour l'oscillation asynchrone peuvent être réglées dans le programme pièce.

Si les données de réglage sont écrites directement dans le programme pièce, la modification prend effet dès le prétraitement. Le comportement synchrone peut être obtenu par un arrêt du prétraitement (STOPRE).

Code de programme	Commentaire
$SA_OSCILL_REVERSE_POS1[Z]=-10	
$SA_OSCILL_REVERSE_POS2[Z]=10	
G0 X0 Z0	
WAITP(Z)	
ID=1 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS1[Z] DO $AA_OVR[X]=0	; Arrêt de l'axe de pénétration lorsque la valeur réelle de l'axe d'oscillation a dépassé le point d'inversion.
ID=2 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[X]=0	
OS[Z]=1 FA[X]=1000 POS[X]=40	; Activer l'oscillation.
OS[Z]=0	; Désactiver l'oscillation.
M30	

Informations complémentaires

Axe d'oscillation

Pour les axes d'oscillation, il est convenu que :

- chaque axe peut être utilisé comme axe d'oscillation
- Plusieurs axes d'oscillation peuvent être actifs simultanément (au maximum : le nombre des axes de positionnement).
- Quelle que soit la fonction G momentanément active dans le programme, l'interpolation linéaire G1 est toujours active pour l'axe d'oscillation.

L'axe d'oscillation peut être :

- un axe d'entrée pour la transformation dynamique
- un axe directeur pour les axes Gantry et conjugués
- déplacé :
  - Sans limitation des à-coups "BRISK"
    ou
  - Avec limitation des à-coups "SOFT"
    ou
  - avec une ligne caractéristique d'accélération coudée (comme les axes de positionnement)

Points d'inversion de sens

Pour déterminer les points d'inversion de l'oscillation, il convient de tenir compte des décalages actuels ci-après :

- Déclaration en absolu
  "OSP1[Z]=<valeur>"
  Position du point d'inversion = Somme des décalages + valeur programmée
- Déclaration en relatif
  "OSP1[Z]=IC(<valeur>)"
  Position du point d'inversion = point d'inversion 1 + valeur programmée

Exemple :

Code de programme
`N10 OSP1[Z]=100 OSP2[Z]=110`
`...`
`N40 OSP1[Z]= IC(3)`

WAITP

Pour faire osciller un axe géométrique, il faut débloquer ce dernier pour l'oscillation avec "WAITP".

Lorsque l'oscillation est terminée, "WAITP" l'axe d'oscillation est à nouveau défini comme axe de positionnement et peut à nouveau être utilisé normalement.

Oscillation avec actions synchrones au déplacement et temps d'arrêt

Après écoulement des temps d'arrêt réglés, le changement de bloc interne a lieu pendant l'oscillation (visible aux nouvelles distances restant à parcourir des axes). La fonction de désactivation est vérifiée lors du changement de bloc. Elle est déterminée après le réglage de la séquence de déplacements (`OSCTRL`). *Ce comportement temporel peut être modifié par la correction de l'avance.*

Il est possible qu'une course d'oscillation soit encore effectuée ensuite avant le démarrage des passes à lécher ou l'accostage de la position finale. *On a l'impression que le comportement à la désactivation est modifié. Ceci n'est toutefois pas le cas.*

3.18.2 Oscillation commandée par des actions synchrones (OSCILL)

Avec ce type d'oscillation, le mouvement de pénétration n'est autorisé qu'aux points d'inversion de sens ou dans des zones d'inversion de sens bien définies.

Selon les besoins, l'oscillation peut

- soit être poursuivie
- soit être arrêtée jusqu'à ce que la pénétration soit entièrement effectuée.

Syntaxe

1. **Fixer les paramètres pour l'oscillation**
2. **Définir les actions synchrones au déplacement**
3. **Affecter les axes, fixer la profondeur de pénétration**

Signification

`OSP1[<Axe d'oscillation>]=`	Position du point d'inversion de sens 1
`OSP2[<Axe d'oscillation>]=`	Position du point d'inversion de sens 2
`OST1[<Axe d'oscillation>]=`	Temps d'arrêt au point d'inversion de sens 1 en secondes
`OST2[<Axe d'oscillation>]=`	Temps d'arrêt au point d'inversion de sens 2 en secondes
`FA[<Axe d'oscillation>]=`	Avance de l'axe d'oscillation
`OSCTRL[<Axe d'oscillation>]=`	Options d'activation ou de désactivation
`OSNSC[<Axe d'oscillation>]=`	Nombre de passes à lécher
`OSE[<Axe d'oscillation>]=`	Position finale
`WAITP(<Axe d'oscillation>)`	Libération de l'axe pour l'oscillation

Affectation des axes, profondeur de pénétration

```
OSCILL[<Axe d'oscillation>]=(<Axe de pénétration 1>,<Axe de
pénétration 2>,<Axe de pénétration 3>)
POSP[<Axe de pénétration>]=(<Position de fin>,<Longueur
partielle>,<Mode>)
```

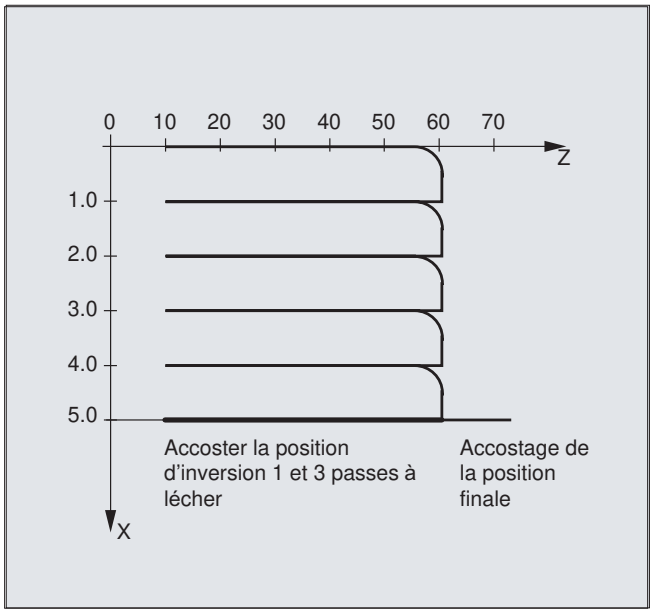
`OSCILL` :	Affecter l'axe ou les axes de pénétration à l'axe d'oscillation
`POSP` :	Définir la pénétration totale et partielle (voir chapitre Gestion des fichiers et programmes)
`Position de fin` :	Position de fin pour l'axe de pénétration, quand toutes les pénétrations partielles ont été effectuées.
`Longueur partielle` :	Valeur de la pénétration partielle au point d'inversion de sens/dans la zone d'inversion de sens
`Mode` :	Division de la pénétration totale en pénétrations partielles = deux dernières pénétrations partielles identiques (préréglage) ; = toutes les pénétrations partielles identiques

Actions synchrones au déplacement

WHEN… … DO	Quand..., alors...
WHENEVER … DO	À chaque fois que..., alors...

Exemple

Il ne doit pas y avoir de pénétration au point d'inversion de sens 1. Au point d'inversion de sens 2, la pénétration doit déjà commencer à la distance ii2 du point d'inversion de sens 2 et l'axe d'oscillation ne doit pas attendre la fin de la pénétration partielle. L'axe Z est axe d'oscillation et l'axe X axe de pénétration.



1. Paramètres pour l'oscillation

Code de programme	Commentaire
DEF INT ii2	; Définir la variable pour la zone d'inversion de sens 2
OSP1[Z]=10 OSP2[Z]=60	; Définir les points d'inversion de sens 1 et 2
OST1[Z]=0 OST2[Z]=0	; Point d'inversion 1 : Arrêt précis fin
	Point d'inversion 2 : Arrêt précis fin
FA[Z]=150 FA[X]=0.5	; Avance de l'axe d'oscillation Z, avance de l'axe de pénétration X
OSCTRL[Z]=(2+8+16,1)	; Désactiver l'oscillation au point d'inversion de sens 2 ; après effacement de la distance restant à parcourir, passes à lécher et accostage de la position de fin ; après effacement de la distance restant à parcourir, accostage de la position d'inversion de sens correspondante
OSNC[Z]=3	; Passes à lécher
OSE[Z]=70	; Position de fin = 70
ii2=2	; Régler la zone d'inversion de sens
WAITP(Z)	; Autorise l'oscillation pour l'axe Z

2. Action synchrone au déplacement

Code de programme	Commentaire
WHENEVER $AA_IM[Z]<$SA_OSCILL_REVERSE_POS2[Z] DO -> $AA_OVR[X]=0 $AC_MARKER[0]=0	; À chaque fois que la position courante de l'axe d'oscillation Z est inférieure au début de la zone d'inversion de sens 2 dans le SCM, mettre la correction axiale de l'axe de pénétration X à 0 % et le mémento avec l'indice 0 à la valeur 0.
WHENEVER $AA_IM[Z]>=$SA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[Z]=0	; À chaque fois que la position courante de l'axe d'oscillation Z est supérieure à la position d'inversion de sens 2 dans le SCM, mettre la correction axiale de l'axe d'oscillation Z à 0 %.
WHENEVER $AA_DTEPW[X] == 0 DO $AC_MARKER[0]=1	; À chaque fois que la distance restant à parcourir pour l'approche partielle est égale à 0, mettre le mémento avec l'indice 0 à la valeur 1.
WHENEVER $AC_MARKER[0]==1 DO $AA_OVR[X]=0 $AA_OVR[Z]=100	; À chaque fois que le mémento avec l'indice 0 est égal à 1, mettre la correction axiale de l'axe d'approche X à 0 %. Ceci permet d'empêcher une approche prématurée (axe d'oscillation Z n'a pas encore quitté la zone d'inversion 2, l'axe d'approche X est toutefois prêt pour une nouvelle approche). Modifier la correction axiale de l'axe d'oscillation Z de 0 % (action de la 2e action synchrone) de nouveau à la valeur 100 % pour le déplacement.

-> doit être programmé dans un bloc

3. Démarrage de l'oscillation

Code de programme	Commentaire
OSCILL[Z]=(X) POSP[X]=(5,1,1)	; Démarrage des axes L'axe X est affecté comme axe de pénétration à l'axe d'oscillation Z. L'axe X doit rejoindre la position de fin 5 par incréments de 1.
M30	; Fin du programme

plus d'informations

1. **Fixer les paramètres pour l'oscillation**
   Les paramètres pour l'oscillation sont à fixer dans le programme avant le bloc de déplacement qui contient d'une part la correspondance entre l'axe de pénétration et l'axe d'oscillation et d'autre part l'instruction avec les paramètres de pénétration (voir "Oscillation asynchrone").
2. **Définir les actions synchrones au déplacement**
   Par le biais des actions synchrones, il est possible de :
   Inhiber le mouvement de pénétration, jusqu'à ce que l'axe d'oscillation se trouve à l'intérieur d'une zone d'inversion
   (ii1, ii2) ou à un point d'inversion (U1, U2).
   Arrêter l'oscillation pendant la pénétration au point d'inversion de sens.
   Redémarrer l'oscillation lorsque la pénétration partielle est **terminée**.
   Définir le lancement de la pénétration partielle suivante .
3. **Affecter les axes d'oscillation et de pénétration** et définir **la pénétration totale et partielle**.

Fixer les paramètres pour l'oscillation

Affectation des axes d'oscillation et de pénétration : OSCILL

```
OSCILL[<Axe d'oscillation>]=(<Axe de pénétration1>,<Axe de
pénétration2>,<Axe de pénétration3>)
```

L'instruction "OSCILL" permet d'effectuer les affectations d'axe et le démarrage de l'oscillation.

Trois axes de pénétration peuvent être affectés au maximum à un axe d'oscillation.

Remarque

Avant le démarrage de l'oscillation, les conditions synchrones pour le comportement des axes doivent être définies.

Définition des pénétrations : POSP

```
POSP[<Axe de pénétration>]=(<Position de fin>,<Longueur
partielle>,<Mode>)
```

Avec l'instruction "POSP", on fournit les indications suivantes à la commande :

- la pénétration totale (en se basant sur la position de fin)
- la valeur de la pénétration partielle au point d'inversion de sens ou dans la zone d'inversion de sens
- le comportement lorsque la position de fin est atteinte (avec le mode)

Mode = 0	Pour les deux dernières pénétrations partielles, la distance restant à parcourir jusqu'à la destination est divisée en deux incréments identiques (préréglage).
Mode = 1	Toutes les pénétrations partielles sont identiques. Elles sont calculées à partir de la pénétration totale.

Définir les actions synchrones au déplacement

De façon tout à fait générale, on utilise pour l'oscillation les actions synchrones au déplacement énumérées ci-après.

Des exemples sont donnés qui pourront vous servir de base pour réaliser vos propres programmes d'oscillation.

Remarque

Dans certains cas particuliers, les conditions synchrones peuvent également être programmées différemment.

Mots-clés

`WHEN ... DO ...`	Quand..., alors...
`WHENEVER ... DO`	À chaque fois que..., alors...

Fonctions

Avec les instructions décrites en détail ci-après, vous pouvez réaliser les fonctions suivantes :

1. Pénétration au point d'inversion de sens.
2. Pénétration dans la zone d'inversion de sens.
3. Pénétration aux deux points d'inversion de sens.
4. Arrêt de l'oscillation au point d'inversion de sens.
5. Redémarrage de l'oscillation.
6. Inhibition puis démarrage de la pénétration partielle suivante.

Toutes les actions synchrones mentionnées ici à titre d'exemple sont basées sur les hypothèses suivantes :

- point d'inversion de sens 1 < point d'inversion de sens 2
- Z = axe d'oscillation
- X = axe de pénétration

Remarque

Pour plus d'explications, voir le chapitre Actions synchrones au déplacement.

Affecter les axes d'oscillation et de pénétration et définir la pénétration totale et partielle.

Pénétration dans la zone d'inversion de sens

Le mouvement de pénétration doit débuter à l'intérieur d'une zone d'inversion de sens, avant que le point d'inversion de sens ne soit atteint.

Ces actions synchrones inhibent le mouvement de pénétration jusqu'à ce que l'axe d'oscillation se trouve dans une zone d'inversion de sens.

Avec les hypothèses formulées (voir ci-dessus), les instructions sont les suivantes :

Zone d'inversion de sens 1 :

```
WHENEVER $AA_IM[Z]>$SA_OSCILL_RESERVE_POS1[Z]+ii1 DO $AA_OVR[X] = 0
```

À chaque fois que la position courante de l'axe d'oscillation est supérieure au début de la zone d'inversion de sens 1 dans le SCM, mettre la correction axiale de l'axe de pénétration à 0 %.

Zone d'inversion de sens 2 :

```
WHENEVER $AA_IM[Z]<$SA_OSCILL_RESERVE_POS2[Z]+ii2 DO $AA_OVR[X] = 0
```

À chaque fois que la position courante de l'axe d'oscillation est inférieure au début de la zone d'inversion de sens 2 dans le SC, mettre la correction axiale de l'axe de pénétration à 0 %.

Pénétration au point d'inversion de sens

Tant que l'axe d'oscillation n'a pas atteint le point d'inversion, l'axe de pénétration n'effectue aucun déplacement.

Avec les hypothèses formulées (voir ci-dessus), les instructions sont les suivantes :

Zone d'inversion de sens 1 :

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

À chaque fois que la position courante de l'axe d'oscillation Z est supérieure ou inférieure à la position du point d'inversion de sens 1 dans le SCM, mettre la correction axiale de l'axe de pénétration X à 0 % et la correction axiale de l'axe d'oscillation Z à 100 %.

Zone d'inversion de sens 2 :

Pour le point d'inversion 2 :

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

À chaque fois que la position courante de l'axe d'oscillation Zu est supérieure ou inférieure à la position du point d'inversion de sens 2 dans le SCM, mettre la correction axiale de l'axe de pénétration X à 0 % et la correction axiale de l'axe d'oscillation Z à 100 %.

Arrêt de l'oscillation au point d'inversion de sens

L'axe d'oscillation s'arrête au point d'inversion de sens, le mouvement de pénétration commence au même moment. L'oscillation est poursuivie lorsque la pénétration est entièrement effectuée.

Cette action synchrone peut également être utilisée pour redémarrer le mouvement de pénétration si celui-ci a été arrêté par une action synchrone précédente encore active.

Avec les hypothèses formulées (voir ci-dessus), les instructions sont les suivantes :

Zone d'inversion de sens 1 :

```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

À chaque fois que la position courante de l'axe d'oscillation est égale à la position d'inversion de sens 1 dans le SCM, mettre la correction axiale de l'axe d'oscillation à 0 % et la correction axiale de l'axe de pénétration à 100 %.

Zone d'inversion de sens 2 :
```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```
À chaque fois que la position courante de l'axe d'oscillation Zu est égale à la position d'inversion de sens 2 dans le SCM, mettre la correction axiale de l'axe d'oscillation X à 0 % et la correction axiale de l'axe de pénétration à 100 %.

Exploitation en ligne du point d'inversion de sens

Si une variable d'exécution marquée par `$$` figure du côté droit de la comparaison, les deux variables seront exploitées en continu à la période d'appel de l'interpolateur et comparées entre elles.

Remarque

Plus d'informations à ce sujet, voir chapitre "Actions synchrones au déplacement".

Redémarrage de l'oscillation

Cette action synchrone sert à reprendre le mouvement d'oscillation, quand la pénétration partielle est terminée.

Avec les hypothèses formulées (voir ci-dessus), les instructions sont les suivantes :

```
WHENEVER $AA_DTEPW[X]==0 DO $AA_OVR[Z]= 100
```
À chaque fois que la distance restant à parcourir pour la pénétration partielle de l'axe de pénétration X est égale à zéro dans le SCM, mettre la correction axiale de l'axe d'oscillation à 100 %.

Pénétration partielle suivante

Lorsque la pénétration a été effectuée, il convient d'éviter que la pénétration partielle suivante ne démarre trop tôt.

On utilise pour ce faire un mémento spécifique au canal (`$AC_MARKER[Index]`) qui est mis à 1 à la fin de la pénétration partielle (distance partielle restant à parcourir 0) et qui est effacé à la sortie de la zone d'inversion de sens. Une action synchrone empêche ensuite le mouvement de pénétration suivant.

Selon les hypothèses formulées (voir ci-dessus), les instructions sont les suivantes pour le point d'inversion de sens 1 par exemple :

1. Définir les marques :
```
WHENEVER $AA_DTEPW[X]==0 DO $AC_MARKER[1] = 1
```
À chaque fois que la distance restant à parcourir pour la pénétration partielle de l'axe de pénétration X est égale à zéro dans le SCM, mettre le mémento avec l'indice 1 à la valeur 1.

2. Effacer les marques
```
WHENEVER $AA_IM[Z]<> $SA_OSCILL_RESERVE_POS1[Z] DO $AC_MARKER[1] = 0
```
Chaque fois que la position actuelle de l'axe d'oscillation Z est supérieure ou inférieure à la position du point d'inversion de sens 1 dans le SCM, mettre le mémento 1 à 0.

3. Inhiber la pénétration

```
WHENEVER $AC_MARKER[1]==1 DO $AA_OVR[X] = 0
```

À chaque fois que le mémento 1 est égal, mettre la correction axiale de l'axe de pénétration X à 0 %.

3.19 Poinçonnage et grignotage

3.19.1 Activation/désactivation

3.19.1.1 Activation/désactivation du poinçonnage et du grignotage (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC)

Activation/désactivation du poinçonnage ou du grignotage

PON et SON permettent d'activer la fonction de poinçonnage ou de grignotage. SPOF met fin à toutes les fonctions spécifiques au poinçonnage et au grignotage. Les instructions modales PON et SON s'excluent mutuellement, c'est-à-dire que PON désactive SON et vice versa.

Poinçonnage/grignotage à la période d'appel de l'interpolateur

Avec SONS et PONS, vous activez aussi respectivement la fonction poinçonnage et la fonction grignotage.

Contrairement au pilotage du poinçon au niveau de l'interpolation actif avec SON/PON, le pilotage du poinçon s'effectue ici par le biais de signaux au niveau servo. Ceci permet de travailler avec une cadence de frappe plus rapide et donc une meilleure productivité.

Pendant le traitement des signaux à la période d'appel de l'interpolateur, toutes les fonctions susceptibles de modifier la position des axes de grignotage ou de poinçonnage sont verrouillées (par ex. déplacement par manivelle électronique, modifications de frames via AP, fonctions de mesure).

Poinçonnage avec temporisation

PDELAYON temporise l'application du coup de poinçon. Cette instruction modale comporte une fonctionnalité préparatoire et se place de ce fait, en règle générale, avant PON. Après PDELAYOF, le poinçonnage se poursuit de manière normale.

Remarque

La définition du temps de retard s'effectue dans la donnée de réglage SD42400 $SC_PUNCH_DWELLTIME.

Accélération asservie à la course

PUNCHACC permet de paramétrer une caractéristique d'accélération, définissant différentes accélérations selon la distance entre les trous.

Deuxième interface de poinçonnage

Pour les machines utilisant en alternance une deuxième interface de poinçonnage (deuxième unité de poinçonnage ou support similaire), il est possible de commuter sur une deuxième paire d'entrées et de sorties TOR rapides de la commande (paire d'E/S). La fonctionnalité de poinçonnage/grignotage complète est applicable avec chacune des deux interfaces de

poinçonnage. La commutation entre la première et la deuxième interface de poinçonnage s'effectue avec les instructions SPIF1 et SPIF2.

Remarque

Condition : une deuxième paire d'E/S doit être définie dans un paramètre machine pour la fonctionnalité de poinçonnage (→ voir les indications du constructeur de machine !).

Syntaxe

```
PON G... X... Y... Z...
SON G... X... Y... Z...
SONS G... X... Y... Z...
PONS G... X... Y... Z...
PDELAYON
PDELAYOF
PUNCHACC(<Smini>,<Amini>,<Smaxi>,<Amaxi>)
SPIF1/SPIF2
SPOF
```

Signification

PON :	Activer le poinçonnage	
SON :	Activer le grignotage	
PONS :	Activer le poinçonnage à la période d'appel de l'interpolateur	
SONS :	Activer le grignotage à la période d'appel de l'interpolateur	
SPOF :	Désactiver le poinçonnage/grignotage	
PDELAYON :	Activer le poinçonnage avec temporisation	
PDELAYOF :	Désactiver le poinçonnage avec temporisation	
PUNCHACC :	Activation de l'accélération en fonction de la distance Paramètres :	
	<Smini>	Distance minimale entre les trous
	<Amini>	Accélération initiale <Amini> peut être supérieur à <Amaxi>.
	<Smaxi>	Distance maximale entre les trous
	<Amaxi>	Accélération finale <Amax> peut être supérieur à <Amin>.
SPIF1 :	Activer la **première** interface de poinçonnage Le pilotage du poinçon s'effectue avec la première paire d'E/S rapides.	
SPIF2 :	Activer la **deuxième** interface de poinçonnage Le pilotage du poinçon s'effectue avec la deuxième paire d'E/S rapides.	
	Remarque : après un RESET ou un démarrage de la commande, la première interface de poinçonnage est toujours active. Si une seule interface de poinçonnage est utilisée, il n'est pas nécessaire de la programmer.	

Exemples

Exemple 1 : activer le grignotage

Code de programme	Commentaire
...	
N70 X50 SPOF	; Positionnement sans déclenchement de poinçonnage.
N80 X100 SON	; Activer le grignotage, déclenchement d'un coup avant le déplacement (X=50) et à la fin du déplacement programmé (X=100).
...	

Exemple 2 : poinçonnage avec temporisation

Code de programme	Commentaire
...	
N170 PDELAYON X100 SPOF	; Positionnement sans déclenchement de poinçonnage, activation du déclenchement de poinçonnage temporisé.
N180 X800 PON	; Activer le poinçonnage. Lorsque la position finale est atteinte, un coup de poinçon est déclenché avec une temporisation.
N190 PDELAYOF X700	; Désactiver le poinçonnage avec une temporisation, déclenchement normal du poinçonnage à la fin du déplacement programmé.
...	

Exemple 3 : poinçonnage avec deux interfaces de poinçonnage

Code de programme	Commentaire
...	
N170 SPIF1 X100 PON	; A la fin du bloc, le déclenchement d'un coup est réalisé sur la première sortie rapide. Une surveillance du signal "Coup actif" a lieu sur la première entrée.
N180 X800 SPIF2	; Le deuxième déclenchement d'un coup est réalisé sur la deuxième sortie rapide. Une surveillance du signal "Coup actif" a lieu sur la deuxième entrée.
N190 SPIF1 X700	; La commande du coup s'effectue avec la première interface pour tous les autres coups.
...	

Informations complémentaires

Poinçonnage et grignotage à la période d'appel de l'interpolateur (PONS/SONS)

Le poinçonnage et le grignotage à la période d'appel de l'interpolateur ne sont pas possibles simultanément dans plusieurs canaux. L'activation de PONS ou SONS n'est possible que dans un seul canal à la fois.

Accélération en fonction de la course (PUNCHACC)

Exemple :

```
PUNCHACC(2,50,10,100)
```

Distances entre les trous inférieures à 2mm :

Le déplacement a lieu avec une accélération égale à 50% de l'accélération maximale.

Distances entre les trous comprises entre 2mm et 10mm :

L'accélération augmente proportionnellement à la distance, jusqu'à 100%.

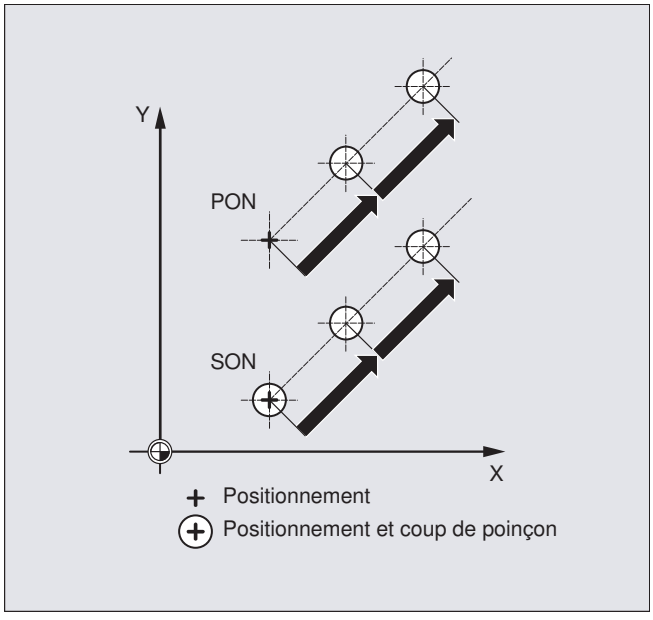
Distances entre les trous supérieures à 10mm :

Déplacement avec une accélération égale à 100%.

Déclenchement du premier coup de poinçon

Selon qu'il s'agit du poinçonnage ou du grignotage, le déclenchement du premier coup de poinçon, après activation de la fonction, intervient à des instants différents :

- `PON/PONS` :
  - Tous les coups de poinçon - y compris celui du premier bloc après l'activation - sont exécutés en fin de bloc.
- `SON/SONS` :
  - Le premier coup de poinçon après l'activation du grignotage, est toujours exécuté en début de bloc.
  - Les autres coups de poinçon sont toujours libérés en fin de bloc.



Poinçonnage et grignotage sur place

Un coup de poinçon ne peut être déclenché que si le bloc contient une instruction de déplacement pour les axes de poinçonnage ou de grignotage (axes du plan actif).

Afin de pouvoir néanmoins déclencher un coup de poinçon, tout en restant au même endroit, l'un des axes de poinçonnage/grignotage est programmé avec un déplacement égal à 0.

Utilisation d'outils indexables

Remarque

Pour positionner des outils indexables de façon tangentielle à la trajectoire programmée, utilisez la fonction Positionnement tangentiel.

Utilisation d'instructions M

Grâce à la macroprogrammation, il est toujours possible d'utiliser des fonctions M spéciales au lieu des instructions en langage de programmation (compatibilité). Les correspondances avec les systèmes plus anciens sont les suivantes :

```
M20, M23   ≙  SPOF
M22        ≙  SON
M25        ≙  PON
M26        ≙  PDELAYON
```

Exemple de fichier macro :

Code de programme	**Commentaire**
`DEFINE M25 AS PON`	`; Activation du poinçonnage`
`DEFINE M125 AS PONS`	`; Activation du poinçonnage à la période d'appel de l'interpolateur`
`DEFINE M22 AS SON`	`; Activation du grignotage`
`DEFINE M122 AS SONS`	`; Activation du grignotage à la période d'appel de l'interpolateur`
`DEFINE M26 AS PDELAYON`	`; Activation du poinçonnage avec temporisation`
`DEFINE M20 AS SPOF`	`; Désactivation du poinçonnage, du grignotage`
`DEFINE M23 AS SPOF`	`; Désactivation du poinçonnage, du grignotage`

Exemple de programmation :

Code de programme	**Commentaire**
`...`	
`N100 X100 M20`	`; Positionnement sans déclenchement de poinçonnage.`
`N110 X120 M22`	`; Activer le grignotage, déclenchement d'un coup avant et après le déplacement.`
`N120 X150 Y150 M25`	`; Activer le poinçonnage, déclenchement d'un coup à la fin du déplacement.`
`...`	

3.19.2 Segmentation automatique des déplacements

Subdivision en distances partielles

Quand le poinçonnage ou le grignotage est activé, SPP et SPN provoquent la segmentation en un nombre donné de distances partielles de même longueur (segmentation équidistante des déplacements) de l'ensemble de la distance de déplacement programmée pour les axes à interpolation. En interne, à chaque distance partielle correspond un bloc.

Nombre de coups de poinçon

En poinçonnage, le premier coup de poinçon intervient au point final de la première distance partielle ; par contre, en grignotage, il intervient au point de départ de la première distance partielle. Sur la distance globale parcourue, les valeurs sont les suivantes :

Poinçonnage : nombre de coups = nombre de distances partielles

Grignotage : nombre de coups = nombre de distances partielles + 1

Fonctions auxiliaires

Les fonctions auxiliaires sont exécutées dans le premier des blocs générés.

Syntaxe

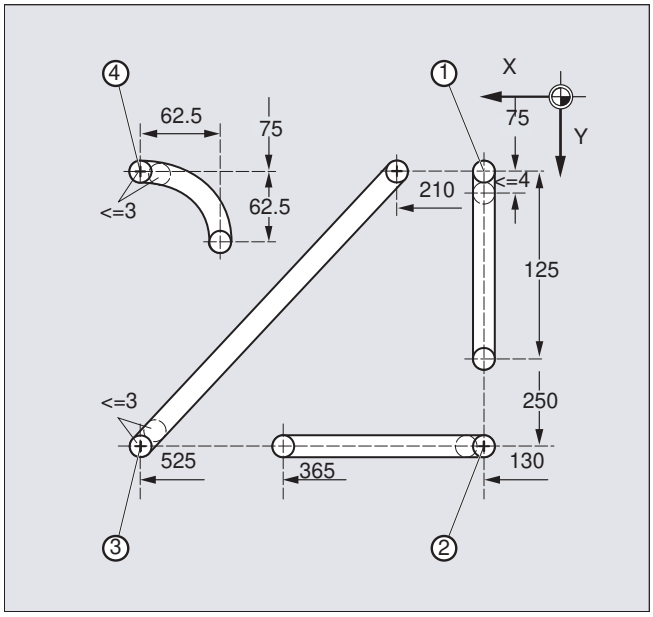
```
SPP=
SPN=
```

Signification

`SPP` :	Taille de la distance partielle (distance maximale entre coups de poinçon consécutifs) ; fonction à effet modal
`SPN` :	Nombre de distances partielles par bloc ; fonction à effet modal

Exemple 1

Les parcours de grignotage doivent être décomposés automatiquement en distances partielles identiques.

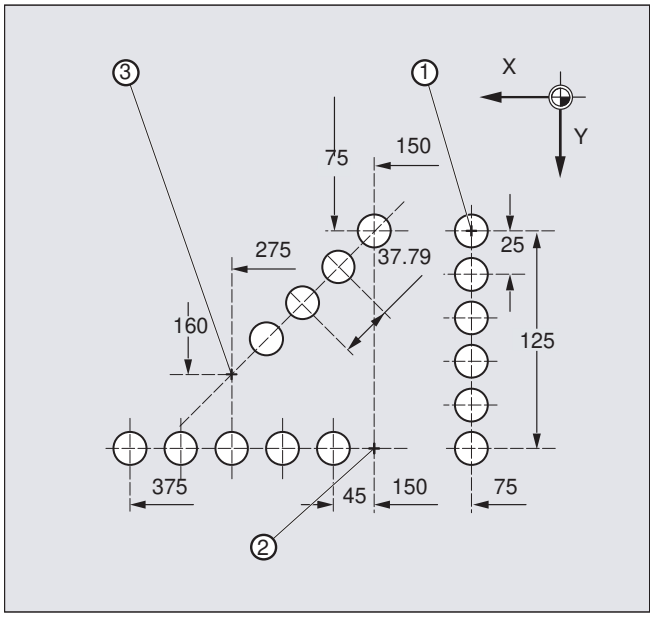


Code de programme	**Commentaire**
N100 G90 X130 Y75 F60 SPOF	; Positionnement sur le point de départ 1
N110 G91 Y125 SPP=4 SON	; Activation du grignotage ; longueur maximale de distance partielle pour segmentation automatique des déplacements : 4 mm
N120 G90 Y250 SPOF	; Désactiver le grignotage ; positionnement sur le point de départ 2
N130 X365 SON	; Activation du grignotage ; longueur maximale de distance partielle pour segmentation automatique des déplacements : 4 mm
N140 X525 SPOF	; Désactiver le grignotage ; positionnement sur le point de départ 3
N150 X210 Y75 SPP=3 SON	; Activation du grignotage ; longueur maximale de distance partielle pour segmentation automatique des déplacements : 3 mm
N160 X525 SPOF	; Désactiver le grignotage ; positionnement sur le point de départ 4

Code de programme	Commentaire
N170 G02 X-62.5 Y62.5 I J62.5 SPP=3 SON	; Activation du grignotage ; longueur maximale de distance partielle pour segmentation automatique des déplacements : 3 mm
N180 G00 G90 Y300 SPOF	; Désactiver le grignotage

Exemple 2

On souhaite une segmentation automatique des déplacements pour chaque rangée de trous. Pour la segmentation, la longueur maximale de distance partielle (valeur de SPP) sera programmée.



Code de programme	Commentaire
N100 G90 X75 Y75 F60 PON	; Positionnement sur le point de départ 1 ; activation du poinçonnage, poinçonner un trou unique
N110 G91 Y125 SPP=25	; Longueur maximale de distance partielle pour segmentation automatique des déplacements : 25 mm
N120 G90 X150 SPOF	; Désactiver le poinçonnage ; positionnement sur le point de départ 2
N130 X375 SPP=45 PON	; Activation du poinçonnage ; longueur maximale de distance partielle pour segmentation automatique des déplacements : 45 mm
N140 X275 Y160 SPOF	; Désactiver le poinçonnage ; positionnement sur le point de départ 3
N150 X150 Y75 SPP=40 PON	; Activer le poinçonnage ; au lieu de la longueur de distance partielle programmée de 40 mm, la longueur de distance partielle calculée de 37,79 mm est utilisée.
N160 G00 Y300 SPOF	; Désactiver le poinçonnage ; positionnement

3.19.2.1 Segmentation des déplacements pour axes à interpolation

Longueur de la distance partielle SPP

Avec `SPP`, vous définissez la distance maximale entre coups de poinçon consécutifs et également la longueur maximale des distances partielles, sur lesquelles la distance totale de déplacement doit être répartie. La désactivation de l'instruction se fait avec `SPOF` ou `SPP=0`.

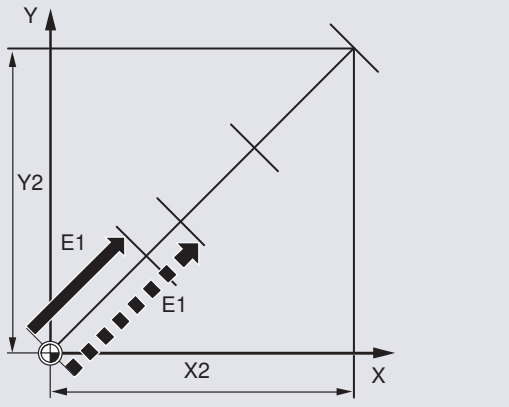
Exemple :

```
N10 SON X0 Y0
N20 SPP=2 X10
```

La distance totale de déplacement de 10 mm est divisée en 5 distances partielles de 2 mm chacune (`SPP=2`).

Remarque

La segmentation des déplacements avec `SPP` donne toujours lieu à des déplacements équidistants : toutes les distances partielles ont la même longueur. Autrement dit, la grandeur de la distance partielle programmée (valeur de `SPP`) n'est valable que si, et seulement si, le quotient de la distance totale de déplacement par la valeur de SPP est un entier. Si ce n'est pas le cas, la taille de la distance partielle est réduite, de façon interne, de manière à que le quotient soit un entier.



X2/Y2 Distance à parcourir programmée (bloc de grignotage / poinçonnage)
E1 Distance partielle programmée
E1' Distance partielle automatiquement arrondie

Exemple :

```
N10 G1 G91 SON X10 Y10
N20 SPP=3.5 X15 Y15
```

Avec une distance totale de déplacement de 15 mm et une distance partielle de 3,5 mm, on obtient un quotient qui n'est pas un nombre entier (4,28). Une réduction de la valeur de SPP a

donc lieu jusqu'à l'obtention d'un quotient entier. Il en résulte une taille de distance partielle de 3 mm.

Nombre de distances partielles SPN

Avec `SPN` , vous précisez le nombre de distances partielles à générer sur l'ensemble de la distance de déplacement. La longueur des distances partielles est calculée automatiquement. `SPN` ayant un effet non modal, il convient d'activer auparavant le poinçonnage ou le grignotage avec `PON` ou `SON`.

3.19.2.2 Segmentation du déplacement dans le cas d'axes individuels

Si, en plus des axes d'interpolation, des axes individuels sont aussi configurés comme axes de poinçonnage/grignotage, ceux-ci sont aussi soumis à la segmentation automatique des déplacements.

Comportement d'un axe individuel avec SPP

La longueur programmée de la distance partielle (`SPP`) se rapporte en premier lieu aux axes d'interpolation. C'est la raison pour laquelle la valeur SPP est ignorée dans un bloc où sont programmés le déplacement d'axe individuel et la valeur SPP, mais aucun axe d'interpolation.

Si, dans un bloc, des axes individuels aussi bien que des axes d'interpolation sont programmés, le comportement de l'axe individuel dépend du réglage du paramètre machine concerné.

1. Réglage par défaut
   Le déplacement de l'axe individuel est réparti de façon égale sur les blocs intermédiaires générés par `SPP`.

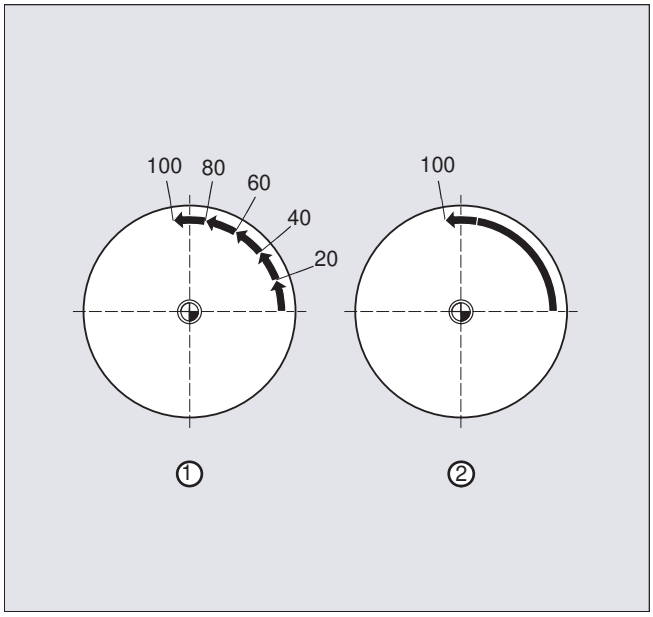
Exemple :

```
N10 G1 SON X10 A0

N20 SPP=3 X25 A100
```

En fonction de la distance entre coups de poinçon de 3 mm, le déplacement global de l'axe X (axe d'interpolation) de 15 mm donne lieu à 5 blocs.

L'axe A pivote de ce fait de 20° par bloc.



1. Axe individuel sans segmentation des déplacements
   L'axe individuel parcourt son déplacement global dans le premier des blocs générés.
2. Segmentation variable des déplacements
   Le comportement de l'axe individuel dépend de l'interpolation des axes d'interpolation :

- Interpolation circulaire : Segmentation des déplacements
- Interpolation linéaire : pas de segmentation des déplacements

Comportement avec SPN

Le nombre de distances partielles programmé reste valable, même si un axe d'interpolation n'a pas été programmé en même temps.

Condition : L'axe individuel est défini comme axe de poinçonnage/grignotage.

3.20 Rectification

3.20.1 Activer/désactiver la surveillance d'outil spécifique à la rectification (TMON, TMOF)

Les procédures TMON(...) et TMOF(...) prédéfinies permettent d'activer ou de désactiver la surveillance d'outil spécifique à la rectification (surveillance de la géométrie et de la vitesse).

Condition

Les paramètres $TC_TPG1 à $TC_TPG9 spécifiques à un outil doivent être définis.

Syntaxe

```
TMON(<TNr>)
...
TMOF(<TNr>)
```

Signification

`TMON(...)` :	Activer la surveillance d'outil spécifique à la rectification L'instruction doit être programmée dans le canal dans lequel la surveillance d'outil spécifique à la rectification doit être activée.
`TMOF(...)` :	Désactiver la surveillance d'outil spécifique à la rectification L'instruction doit être programmée dans le canal dans lequel la surveillance d'outil spécifique à la rectification doit être désactivée.
`<TNr>` :	Numéro T **Remarque :** Requis uniquement lorsque la surveillance doit être activée ou désactivée pour une meule non active à la place de l'outil actif mis en œuvre.
`TMOF(0)` :	désactiver la surveillance pour tous les outils

3.21 Arrêt étendu et retrait (ESR)

La fonction "Arrêt étendu et retrait", désignée par le sigle ESR dans la suite de ce document, permet de réagir avec souplesse aux situations d'erreur en fonction du process :

- **Arrêt étendu**
  Dans la mesure où le cas d'erreur spécifique le permet, tous les axes débloqués pour l'arrêt étendu sont immobilisés de manière ordonnée.
- **Retrait**
  L'outil se trouvant en cours d'intervention est retiré de la pièce le plus rapidement possible.
- **Fonctionnement en génératrice (fonction SINAMICS "Régulation Vdc")**
  En cas de dépassement par la bas d'une valeur paramétrée de la tension de circuit intermédiaire, par ex. en cas de panne de réseau, l'énergie électrique nécessaire au retrait est générée par réinjection de l'énergie de freinage d'un entraînement prévu à cet effet (fonctionnement en génératrice).

Causes de déclenchement

Causes générales (externes à la CN / globales ou spécifiques à un GMFC/canal)

- Entrées TOR (par ex. d'un module NCU) ou mémoire image lisible, interne à la commande, des sorties TOR ($A_IN, $A_OUT)
- Etat du canal ($AC_STAT)
- Signaux VDI ($A_DBB)
- Signalisations groupées d'un certain nombre d'alarmes ($AC_ALARM_STAT)

Causes axiales

- Seuil de retrait d'urgence de l'axe asservi (synchronisme du couplage électronique, $VC_EG_SYNCDIFF[<axe asservi>])
- Entraînement : seuil d'alarme du circuit intermédiaire (menace de sous-tension imminente), $AA_ESR_STAT[<axe>]
- Entraînement : seuil de vitesse minimale en génératrice (plus d'énergie de rotation réinjectable disponible), $AA_ESR_STAT[<axe>]

Logique combinatoire des actions synchrones statiques : Combinaison de causes/réactions

Les possibilités flexibles de combinaison offertes par les actions synchrones statiques sont utilisées pour déclencher assez rapidement certaines réactions en fonction des différentes causes.

La combinaison de toutes les causes possibles à l'aide d'actions synchrones statiques incombe à l'utilisateur. Il peut évaluer les variables système des causes soit dans leur ensemble, soit de manière sélective à l'aide de masques de bits, puis leur associer les réactions de son choix. Les actions synchrones statiques sont effectives dans tous les modes de fonctionnement.

Activation

Déblocage des fonctions

Les fonctions de mode génératrice, d'arrêt et de retrait sont débloquées par mise à un du signal de commande correspondant $AA_ESR_ENABLE. Ce signal de commande peut être modifié par des actions synchrones.

Déclenchement des fonctions

La fonction AER est déclenchée simultanément pour l'ensemble des axes concernés par mise à un de la variable système $AC_ESR_TRIGGER.

Le fonctionnement en génératrice est "automatiquement" activé dans l'entraînement dès la détection d'une menace de sous-tension imminente du circuit intermédiaire.

L'arrêt et/ou le retrait contrôlés par le variateur sont activés dès la détection d'une panne de communication (entre la CN et l'entraînement) et d'une sous-tension du circuit intermédiaire dans l'entraînement (sous réserve que la configuration et les paramètres de déblocage le permettent).

L'arrêt et/ou le retrait contrôlés par le variateur peuvent également être déclenchés depuis la CN par mise à un du signal de commande correspondant $AN_ESR_TRIGGER (ordre de diffusion à tous les entraînements).

Informations complémentaires

Description fonctionnelle Axes et broches

3.21.1 ESR assisté par CN

3.21.1.1 Retrait assisté par CN (POLF, POLFA, POLFMASK, POLFMLIN)

Pour un retrait assisté par CN, certaines conditions initiales sont requises (voir "Retrait assisté par CN (POLF, POLFA, POLFMASK, POLFMLIN) (Page 1010)"). Lorsque ces conditions sont remplies, le relèvement rapide (LIFTFAST) est activé pour le ou les axes de retrait configurés dans le canal par mise à un de la variable système $AC_ESR_TRIGGER (ou $AA_ESR_TRIGGER pour les axes individuels).

Syntaxe

```
POLF(<axe>)=<position>
POLFA(<axe>,<type>,<position>)
POLFMASK(<axe_1>,<axe_2>...)
POLFMLIN(<axe_1>,<axe_2>...)
```

Pour POLFA, les formes abrégées suivantes sont autorisées :

```
POLFA(<axe>,<type>) ; forme abrégée pour retrait d'axe individuel
POLFA(axe,0/1/2) ; désactivation ou activation rapide
POLFA(axe,0,$AA_POLFA[axe]) ; provoque un arrêt du prétraitement des
blocs
```

```
POLFA(axe,0) ; ne provoque pas d'arrêt du prétraitement des blocs
```

Signification

POLF :	Adresse d'indication de la position de destination de l'axe de retrait POLF est une fonction modale.			
	<axe> :	Nom de l'axe géométrique ou de l'axe canal/machine qui se retire.		
	<position> :	Position de retrait		
		Type :	REAL	
		Pour les axes géométriques, le système applicable est SCP, sinon SCM. Lorsque les descripteurs d'axe géométrique et d'axe canal/machine sont identiques, le retrait a lieu dans le SCP.		
POLFA :	Appel de sous-programme prédéfini pour l'indication de la position de retrait des axes individuels			
	<axe> :	Descripteur d'axe de canal		
	<type> :	Mode d'indication de position		
		Type :	INT	
		Valeur :	0 :	Identification de la valeur de position comme non valable
			1 :	La valeur de position est absolue
			2 :	La valeur de position est relative (distance)
		Remarque : Si un axe n'est pas un axe individuel ou bien si le type est absent ou nul, une alarme correspondante est générée.		
	<position> :	Position de retrait (cf. supra)		
		Remarque : La valeur de position est également appliquée avec type=0. Toutefois, cette valeur est alors identifiée comme étant non valable et doit être reprogrammée pour le retrait.		
POLFMASK :	Appel de sous-programme prédéfini pour la sélection des axes qui doivent être retirés **indépendamment les uns des autres** après le déclenchement du relèvement rapide.			
	<axe_1>... :	Noms des axes à déplacer sur leurs positions définies avec POLF lors du relèvement rapide. Tous les axes indiqués doivent se trouver dans le même système de coordonnées.		
	L'instruction POLFMASK() sans indication d'un axe désactive le relèvement rapide pour tous les axes qui ont été retirés indépendamment les uns des autres.			
POLFMLIN :	Appel de sous-programme prédéfini pour la sélection des axes qui doivent être retirés **avec corrélation linéaire** après le déclenchement du relèvement rapide.			
	<axe_1>,... :	cf. supra		
	L'instruction POLFMLIN() sans indication d'un axe désactive le relèvement rapide pour tous les axes qui ont été retirés avec corrélation linéaire.			

Remarque

Avant que le relèvement rapide sur une position fixe puisse être débloqué avec POLFMASK ou POLFMLIN, une position doit avoir été programmée avec POLF pour les axes sélectionnés.

Remarque

Si des axes sont débloqués successivement avec POLFMASK, POLFMLIN ou POLFMLIN, POLFMASK, la dernière définition est toujours valable pour l'axe correspondant.

Remarque

Les positions programmées avec POLF et l'activation avec POLFMASK ou POLFMLIN sont supprimées au démarrage du programme pièce. Cela signifie que l'utilisateur doit reprogrammer les valeurs de POLF et les axes sélectionnés dans POLFMASK ou POLFMLIN pour chaque programme pièce.

Remarque

Si seul le type est modifié lorsque la forme abrégée POLFA est utilisée, l'utilisateur doit veiller à ce que soit la position de retrait soit la distance de retrait contienne une valeur judicieuse. Plus particulièrement, la position et la distance de retrait doivent être redéfinies après un POWER ON.

Exemple

Retrait d'un axe individuel :

Code de programme	Commentaire
PM37500 $MA_ESR_REACTION[AX1]=21	; Retrait assisté par CN.
...	
$AA_ESR_ENABLE[AX1]=1	
POLFA(AX1,1,20.0)	; La position de retrait axiale 20.0 (absolue) est affectée à AX1.
$AA_ESR_TRIGGER[AX1]=1	; Le retrait commence ici.

Informations complémentaires

Conditions requises pour le retrait assisté par CN

- Un axe de retrait est configuré dans le canal pour le retrait assisté par CN :
  PM37500 $MA_ESR_REACTION = 21
- L'ESR doit être débloqué pour cet axe :
  $AA_ESR_ENABLE = 1
- Des temporisations sont définies :
  PM21380 $MC_ESR_DELAY_TIME1
  PM21381 $MC_ESR_DELAY_TIME2
- Les positions de retrait spécifiques à l'axe sont programmées au moyen de POLF dans le programme pièce.

- Les axes sont sélectionnés pour le retrait assisté par CN au moyen de POLFMASK/POLFMLIN.
- Les signaux de déblocage pour le mouvement de retrait doivent être mis à 1 et y rester.

Démarrage et libération du retrait assisté par CN

Lorsque la variable système $AC_ESR_TRIGGER est mise à 1 et qu'un axe de retrait est configuré dans ce canal (à savoir PM37500 $MA_ESR_REACTION = 21), $AA_ESR_ENABLE = 1 étant défini pour cet axe, le relèvement rapide (LIFTFAST) est activé dans ce canal.

Le mouvement de relèvement configuré au moyen de POLF (ou LFPOS) pour les axes sélectionnés au moyen de POLFMASK ou POLFMLIN remplace le déplacement défini dans le programme pièce pour ces axes.

La temporisation maximale disponible pour le retrait est la somme des temps PM21380 $MC_ESR_DELAY_TIME1 et PM21381 $MC_ESR_DELAY_TIME2. Après l'écoulement de ce laps de temps, un freinage rapide est déclenché également pour l'axe de retrait puis la fonction poursuite est déclenchée.

Remarque

L'extension du mouvement de retrait (c'est-à-dire LIFTFAST/LFPOS déclenché par $AC_ESR_TRIGGER) **ne peut pas être interrompue** et ne peut être terminée de manière anticipée que par un arrêt d'urgence.

Remarque

Le retrait déclenché par $AC_ESR_TRIGGER est verrouillé pour empêcher tout retrait multiple.

Retrait d'axe individuel

Pour un retrait d'axe individuel au moyen de POLFA, la position de retrait de l'axe doit être programmée et les conditions suivantes respectées :

- $AA_ESR_ENABLE = 1
- L'<axe> doit être un axe individuel à l'instant du déclenchement ($A**A**_ESR_TRIGGER = 1).
- Le <type> doit être 1 ou 2.

Sens du retrait pour le relèvement rapide

Le frame valable au moment de l'activation du relèvement rapide est pris en compte.

Remarque

Les frames avec rotation influencent également, via POLF, le sens dans lequel le relèvement a lieu.

Permutation d'axe

Les axes de retrait doivent toujours être affectés à un canal CN et ne doivent pas être échangés entre les canaux. Si une tentative de faire passer un axe de retrait dans un autre canal a lieu, une alarme est générée. Ce n'est qu'une fois que l'axe a été à nouveau désactivé avec $AA_ESR_ENABLE[AX] = 0 que l'axe peut être permuté dans un nouveau canal. Une fois la

permutation effectuée, les axes peuvent de nouveau être assignés avec $AA_ESR_ENABLE[AX] = 1.

Axes neutres

Des axes neutres ne peuvent pas exécuter d'ESR assisté par CN.

3.21.1.2 Arrêt assisté par CN

La mise à un de la variable système $AC_ESR_TRIGGER (ou $AA_ESR_TRIGGER pour les axes individuels) active l'arrêt assisté par CN pour le ou les axes d'immobilisation configurés dans le canal.

Conditions

- Un axe d'immobilisation est configuré dans le canal pour l'arrêt assisté par CN :
  PM37500 $MA_ESR_REACTION = 22
- L'ESR doit être débloqué pour cet axe :
  $AA_ESR_ENABLE = 1
- Des temporisations sont définies :
  PM21380 $MC_ESR_DELAY_TIME1 (temporisation axes AER)
  PM21381 $MC_ESR_DELAY_TIME2 (temps AER pour freinage avec interpolation)

Déroulement

Pendant le laps de temps du paramètre machine PM21380, l'interpolation de l'axe continue tranquillement suivant la programmation. Après écoulement du laps de temps du paramètre machine PM21380, le freinage piloté par interpolation (rampe d'arrêt) est amorcé. Pour le freinage piloté avec interpolation, le délai maximal disponible est celui spécifié dans PM21381. Après écoulement de ce délai, le freinage rapide est amorcé suivi de l'asservissement.

Exemple

Immobilisation d'un axe individuel :

```
Code de programme                                   Commentaire
PM37500 $MC_ESR_REACTION[AX1]=22                    ; Arrêt assisté par CN.
PM21380 $MC_ESR_DELAY_TIME1[AX1]=0.3
PM21381 $MC_ESR_DELAY_TIME2[AX1]=0.06
...
$AA_ESR_ENABLE[AX1]=1
$AA_ESR_TRIGGER[AX1]=1                              ; L'immobilisation commence ici.
```

3.21.2 ESR contrôlé par le variateur

3.21.2.1 Configuration d'un arrêt contrôlé par le variateur (ESRS)

La fonction `ESRS(...)` permet de configurer les paramètres d'entraînement pour l'"arrêt" de la fonction AER contrôlée par le variateur.

Syntaxe

```
ESRS(<axe_1>,<temps_arrêt_1>[,...,<axe_n>,<temps_arrêt_n>])
```

Signification

`ESRS(...)` :	Fonction d'écriture des paramètres d'entraînement pour la fonction ESR "Arrêt" La fonction : • doit figurer seule dans le bloc, • déclenche un arrêt du prétraitement des blocs, • ne peut pas être utilisée dans des actions synchrones.	
`<axe_1>`, `...`, `<axe_n>` :	Axe pour lequel l'arrêt contrôlé par le variateur doit être configuré. Le paramètre d'entraînement p0888 (configuration) est écrit dans l'entraînement pour cet axe : p0888 = 1	
	Type :	AXIS
	Plage de valeurs :	Descripteur d'axe de canal
`<temps_arrêt_1>`, `...`, `<temps_arrêt_n>` :	Durée pendant laquelle l'entraînement poursuit un fonctionnement constant à la vitesse de consigne actuelle après l'apparition d'un défaut. Le paramètre d'entraînement p0892 (temporisation) est écrit dans l'entraînement pour l'axe spécifié: p0892 = <temps d'arrêt>	
	Unité :	s
	Type :	REAL
	Plage de valeurs :	0.00 - 20.00
5 axes au maximum peuvent être configurés dans un appel de fonction ; n = 5		

3.21.2.2 Configuration d'un retrait contrôlé par le variateur (ESRR)

La fonction `ESRR(...)` permet de configurer les paramètres d'entraînement pour le "retrait" de la fonction AER contrôlée par le variateur.

Syntaxe

```
ESRR(<axe_1>,<distance_retrait_1>,<vitesse_retrait_1>[,...,<axe_n>,<
distance_retrait_n>,<vitesse_retrait_n>])
```

Signification

`ESRR(...)` :	Fonction d'écriture des paramètres d'entraînement pour la fonction ESR "Retrait" La fonction : • doit figurer seule dans le bloc, • déclenche un arrêt du prétraitement des blocs, • ne peut pas être utilisée dans des actions synchrones.	

`<axe_1>,`
`...,`
`<axe_n>` :
	axe pour lequel le retrait contrôlé par le variateur doit être configuré.
	Le paramètre d'entraînement p0888 (configuration) est écrit dans l'entraînement pour cet axe :
	p0888 = 2	

	Type :	AXIS
	Plage de valeurs :	Descripteur d'axe de canal

`<distance_retrait_1>,`
`...,`
`<distance_retrait_n>` :
	La distance de retrait est convertie en vitesse de retrait pour l'entraînement. La valeur est écrite dans le paramètre d'entraînement p0893 (vitesse) pour l'axe spécifié :
	p0893 = (<distance_retrait_n> convertie en vitesse de retrait)	

	Unité :	mm/min, inch/min, degré/min (en fonction de l'unité de l'axe)
	Type :	REAL
	Plage de valeurs :	MIN - MAX

`<vitesse_`
`retrait_1>,`
`...,`
`<vitesse_`
`retrait_n>` :
	La vitesse de retrait est convertie en durée pour l'entraînement. La valeur est écrite dans le paramètre d'entraînement p0892 [s] pour l'axe spécifié :
	p0892 = <distance_retrait_n> / <vitesse_retrait _n>	

	Unité :	mm/min, inch/min, degré/min (en fonction de l'unité de l'axe)
	Type :	REAL
	Plage de valeurs :	0.00 - MAX
5 axes au maximum peuvent être configurés dans un appel de fonction ; n = 5		

3.22 Temps d'exécution du programme / Compteur de pièces

Des informations relatives au temps d'exécution du programme et au nombre de pièces sont disponibles pour aider l'opérateur de la machine-outil.

Ces informations peuvent être traitées comme variables système dans le programme CN et/ou AP. En même temps, elles sont disponibles pour l'affichage sur l'interface utilisateur.

3.22.1 Temps d'exécution de programme

Pour la surveillance de processus technologiques, la fonction "Temps d'exécution" met à disposition des compteurs chronométriques internes de la CN, qui peuvent être lus dans le programme pièce et dans les actions synchrones au moyen de variables système spécifiques à la CN et au canal.

Le déclencheur de la mesure de temps d'exécution ($AC_PROG_NET_TIME_TRIGGER) est la seule variable système de la fonction, qui est accessible en écriture. Il sert à opérer une mesure sélective de sections de programme. En d'autres termes, l'écriture du déclencheur dans le programme CN permet d'activer et de désactiver la mesure de temps.

Variable système	Signification	Activité
Spécifiques à la CN		
$AN_SETUP_TIME	Temps (en minutes) depuis le dernier démarrage de la commande avec les valeurs par défaut ("démarrage à froid") Remise à "0" automatique à chaque démarrage de la commande avec les valeurs par défaut.	• Activation permanente
$AN_POWERON_TIME	Temps en minutes depuis le dernier démarrage normal de la commande ("démarrage à chaud") Remise à "0" automatique à chaque démarrage normal de la commande.	
Spécifiques au canal		
$AC_OPERATING_TIME	Temps global en secondes pour l'exécution de programmes CN en mode automatique Remise à "0" automatique de la valeur à chaque démarrage de la commande.	• Activation par le paramètre machine MD27860 • Uniquement en mode AUTO
$AC_CYCLE_TIME	Temps d'exécution du programme CN sélectionné en secondes Remise à "0" automatique de la valeur au démarrage d'un nouveau programme CN.	
$AC_CUTTING_TIME	Temps d'usinage en secondes Le temps d'exécution des axes à interpolation (au moins l'un d'entre eux est actif) est totalisé sans l'activation du rapide dans tous les programmes CN, entre le départ CN et la fin du programme / RESET de la CN. La mesure est suspendue pendant tout arrêt temporisé. Remise à "0" automatique de la valeur à chaque démarrage de la commande avec les valeurs par défaut.	

Variable système	Signification		Activité
$AC_ACT_PROG_NET_TIME	Temps d'exécution actuel net en secondes du programme CN courant Remise à "0" automatique au démarrage d'un programme CN.		• Activation permanente • Uniquement en mode AUTO
$AC_OLD_PROG_NET_TIME	Temps d'exécution net du programme terminé correctement à l'instant avec M30, en secondes.		
$AC_OLD_PROG_NET_TIME_COUNT	Modifications sur $AC_OLD_PROG_NET_TIME Après un POWER ON, $AC_OLD_PROG_NET_TIME_COUNT est à "0". $AC_OLD_PROG_NET_TIME_COUNT est toujours incrémenté lorsque la commande a récrit $AC_OLD_PROG_NET_TIME.		
$AC_PROG_NET_TIME_TRIGGER	Déclencheur de la mesure du temps d'exécution :		• Uniquement en mode AUTO
	0	Etat neutre Le déclencheur n'est pas actif.	
	1	Arrêter Arrête la mesure et copie la valeur de $AC_ACT_PROG_NET_TIME dans $AC_OLD_PROG_NET_TIME. $AC_ACT_PROG_NET_TIME est mis à "0" puis continue.	
	2	Départ Lance la mesure et met à "0" $AC_ACT_PROG_NET_TIME. $AC_OLD_PROG_NET_TIME n'est pas modifié.	
	3	Arrêt Arrêt de la mesure. $AC_OLD_PROG_NET_TIME n'est pas modifié et $AC_ACT_PROG_NET_TIME reste constant jusqu'à la reprise.	
	4	Reprise Reprise de la mesure, c'est-à-dire qu'une mesure arrêtée précédemment reprend. $AC_ACT_PROG_NET_TIME se poursuit. $AC_OLD_PROG_NET_TIME n'est pas modifié.	
Un POWER ON remet toutes les variables système à "0".			

Remarque

Constructeur de machines

Les temporisations activables sont activées au moyen du paramètre machine MD27860 $MC_PROCESSTIMER_MODE.

Le comportement des mesures de temps actives pour certaines fonctions (par ex. GOTOS, correction = 0%, test d'avance actif, test de programme, ASUP, PROG_EVENT, ...) est configuré au moyen des paramètres machine MD27850 $MC_PROG_NET_TIMER_MODE et MD27860 $MC_PROCESSTIMER_MODE.

Informations complémentaires : Description fonctionnelle Fonctions de base

Remarque

Temps restant pour une pièce

Lorsqu'il s'agit de produire des pièces identiques dans la foulée, les valeurs de temporisation suivantes permettent de déterminer le temps restant requis pour une pièce :

- temps d'usinage de la dernière pièce produite (voir $AC_OLD_PROG_NET_TIME)
- temps d'usinage actuel (voir $AC_ACT_PROG_NET_TIME)

Le temps restant s'affiche sur l'interface utilisateur, en plus du temps d'usinage actuel.

Remarque

Utilisation de STOPRE

Les variables système $AC_OLD_PROG_NET_TIME et $AC_OLD_PROG_NET_TIME_COUNT n'entraînent pas d'arrêt implicite du prétraitement. En cas d'utilisation dans le programme pièce, ceci n'est pas critique lorsque la valeur des variables système provient de l'exécution précédente du programme. Par contre, si le déclencheur de la mesure du temps d'exécution ($AC_PROG_NET_TIME_TRIGGER) est écrit avec une fréquence élevée, modifiant ainsi fréquemment $AC_OLD_PROG_NET_TIME, il est recommandé d'utiliser un STOPRE explicite dans le programme pièce.

Autres conditions à prendre en compte

- **Recherche de bloc**
  Aucun temps d'exécution de programme n'est déterminé pendant une recherche de bloc.
- **REPOS**
  La durée d'un processus REPOS est ajoutée au temps d'usinage actuel ($AC_ACT_PROG_NET_TIME).

Exemples

Exemple 1 : Mesure du temps de "mySubProgrammA"

Code de programme

```
...
N50 DO $AC_PROG_NET_TIME_TRIGGER=2
N60 FOR ii= 0 TO 300
N70 mySubProgrammA
N80 DO $AC_PROG_NET_TIME_TRIGGER=1
N95 ENDFOR
N97 mySubProgrammB
N98 M30
```

Après le traitement de la ligne N80 par le programme, le temps d'exécution net de "mySubProgrammA" figure dans $AC_OLD_PROG_NET_TIME.

La valeur de $AC_OLD_PROG_NET_TIME :

- est conservée au-delà de M30,
- est actualisée après chaque exécution de boucle.

Exemple 2 : Mesure du temps de "mySubProgrammA" et de "mySubProgrammC"

Code de programme

```
...
N10 DO $AC_PROG_NET_TIME_TRIGGER=2
N20 mySubProgrammA
N30 DO $AC_PROG_NET_TIME_TRIGGER=3
N40 mySubProgrammB
N50 DO $AC_PROG_NET_TIME_TRIGGER=4
N60 mySubProgrammC
N70 DO $AC_PROG_NET_TIME_TRIGGER=1
N80 mySubProgrammD
N90 M30
```

3.22.2 Compteurs de pièces

La fonction "Compteurs de pièces" met à disposition différents compteurs, qu'il est en particulier possible d'utiliser en interne à la commande pour le comptage de pièces.

Les compteurs sont disponibles sous forme de variables système spécifiques à un canal avec accès en écriture et lecture dans la plage de valeurs de 0 à 999 999 999.

Variable système	Signification
$AC_REQUIRED_PARTS	Nombre de pièces à fabriquer (nombre de pièces requis) Ce compteur permet de définir le nombre de pièces qui, lorsqu'il est atteint, entraîne la remise à "0" du nombre de pièces réel ($AC_ACTUAL_PARTS).
$AC_TOTAL_PARTS	Nombre total de pièces fabriquées (nombre total réel de pièces) Ce compteur indique le nombre total de pièces fabriquées à partir du début du comptage. Remise à "0" automatique de la valeur uniquement en cas de démarrage de la commande avec les valeurs par défaut.
$AC_ACTUAL_PARTS	Nombre de pièces fabriquées (nombre réel de pièces) Ce compteur enregistre le nombre total de pièces fabriquées depuis le début du comptage. Le compteur est automatiquement remis à "0" (à condition que $AC_REQUIRED_PARTS > 0), lorsque le nombre de pièces requis ($AC_REQUIRED_PARTS) est atteint.
$AC_SPECIAL_PARTS	Nombre de pièces comptées par l'utilisateur Ce compteur permet à l'utilisateur de définir son propre mode de comptage des pièces. Il est possible de définir l'émission d'une alarme lorsque le nombre de pièce requis est atteint ($AC_REQUIRED_PARTS). L'utilisateur doit assurer lui-même la remise à zéro du compteur.

Remarque

Chaque compteur de pièces est préréglé à sa valeur par défaut "0" au démarrage de la commande et son contenu peut être lu/écrit indépendamment de son mode d'activation.

Remarque

Des paramètres machine spécifiques au canal permettent de régler le mode d'activation des compteurs, l'instant de leur remise à zéro et l'algorithme de comptage.

Remarque

Comptage de pièces avec fonction M définie par l'utilisateur

Des paramètres machine permettent de définir le déclenchement des impulsions de comptage pour les différents compteurs de pièces au moyen de fonctions M définies par l'utilisateur à la place du déclenchement à la fin du programme `M2`/`M30`.

3.23 Simulation de programme

3.23.1 Fonction

Dans la simulation de programme, le programme pièce actuel est calculé dans son intégralité et le résultat représenté sous forme graphique dans l'interface utilisateur. Le résultat de la programmation peut ainsi être contrôlé sans déplacer les axes machine. Ceci permet de détecter à l'avance les phases d'usinage dont la programmation est incorrecte et d'éviter des défauts d'usinage sur la pièce.

CN de simulation

La simulation utilise une instance CN propre (CN de simulation). C'est pourquoi une calibration de la CN réelle par rapport à la CN de simulation doit être effectuée avant un démarrage de la simulation. Lors de cette calibration, entre autres, tous les paramètres machine actifs sont lus depuis la CN et chargés dans la CN de simulation. Les paramètres machine de cycle et CN font partie des paramètres machine actifs.

3.23.2 Simulation de programme associée aux cycles de compilation

Certains cycles de compilation (CC) sont pris en charge lors de la simulation de programme. Les paramètres machine des cycles de compilation pris en charge sont calibrés une seule fois après le démarrage de la commande. Une calibration au "démarrage de la simulation" n'a pas lieu !

Remarque

Les instructions de langage et les paramètres machine spécifiques aux CC ne peuvent **pas** être utilisés dans les programmes pièce pour les CC non pris en charge (voir le paragraphe "Instructions CC dans le programme pièce").

Dans certains cas, les mouvements spéciaux des CC pris en charge (transformations OEM) ne sont pas représentés correctement.

Instructions CC dans le programme pièce

Les instructions de langage du programme pièce provenant de cycles de compilation non pris en charge (`OMA1` ... `OMA5`, `OEMIPO1/2`, `G810` ... `G829`, procédures et fonctions utilisateur) entraînent, sans traitement individuel, la signalisation d'une alarme et l'interruption de la simulation.

Solution :

Traiter individuellement dans le programme pièce les éléments de langage spécifiques aux CC manquants (requête $P_SIM).

Exemple :

Code de programme	Commentaire
N1 G01 X200 F500	
IF (1==$P_SIM)	
N5 X300	; Inactif lors de la simulation CC.
ELSE	
N5 X300 OMA1=10	
ENDIF	

3.24 Autres fonctions

3.24.1 Activation des paramètres machine (NEWCONF)

L'instruction NEWCONF permet d'activer tous les paramètres machine. La fonction peut également être activée dans l'interface utilisateur IHM en actionnant la touche logicielle "Activer PM".

Lors de l'exécution de la fonction "NEWCONF", un arrêt implicite du prétraitement des blocs a lieu, c'est-à-dire que le déplacement avec interpolation est interrompu.

Syntaxe

```
NEWCONF
```

Signification

NEWCONF :	Instruction de prise d'effet de l'ensemble des paramètres machine du niveau de prise d'effet "NEW_CONFIG"

Exécution de NEWCONF pour tous les canaux depuis le programme pièce

Si des paramètres machine axiaux du programme pièce sont modifiés puis activés avec NEWCONF, alors l'instruction NEWCONF active uniquement les paramètres machine entraînant des modifications du canal du programme pièce.

Remarque

Afin d'activer toutes les modifications de façon sûre, l'instruction NEWCONF doit être exécutée dans chaque canal dans lequel doivent être calculés les axes ou fonctions concernés par les modifications des paramètres machine.

Avec NEWCONF, aucun paramètre machine axial n'est activé.

Un RESET axial doit être exécuté pour les axes contrôlés par l'AP.

Exemple

Réseau de trous : réaliser des trous taraudés en différentes positions

```
Code de programme                          Commentaire
N10 $MA_CONTOUR_TOL[AX]=1.0                ; Modifier un paramètre machine.
N20 NEWCONF                                ; Activer des paramètres machine.
...
```

3.24.2 Contrôler le langage CN existant (STRINGIS)

La fonction "STRINGIS(...)" permet de contrôler si la chaîne de caractères indiquée est disponible comme élément de langage de programmation CN dans l'étendue actuelle des langages.

Définition

```
INT STRINGIS(STRING <nom>)
```

Syntaxe

```
STRINGIS(<nom>)
```

Signification

`STRINGIS` :	Fonction avec valeur de retour
`<nom>` :	Nom de l'élément à contrôler du langage de programmation CN
Valeur de retour :	Le format de la valeur de retour est yxx (décimal).

Eléments du langage de programmation CN

Les éléments suivants du langage de programmation CN peuvent être contrôlés :

- instructions G de tous les groupes de fonctions G existants, par exemple "G0", "INVCW", "POLY", "ROT", "KONT", "SOFT", "CUT2D", "CDON", "RMBBL", "SPATH"
- Adresses DIN ou CN, comme "ADIS", "RNDM", "SPN", "SR", "MEAS"
- Fonctions, par exemple "TANG(...)" ou "GETMDACT"
- Procédures par ex. "SBLOF".
- Mots-clés, par exemple "ACN", "DEFINE" ou "SETMS"
- Données système par ex. paramètres machine $M... , données de réglage $S... ou données optionnelles $O...
- Variables système $A... , $V... , $P...
- Paramètres de calcul R...
- Noms de cycles activés
- Variables GUD et LUD
- Noms de macros
- Noms d'étiquettes

Valeur de retour

La valeur de retour est uniquement significative dans les 3 premiers chiffres en notation décimale. Le format de la valeur de retour est yxx, y étant alors = information de base et xx = information détaillée.

Valeur de retour	Signification	
000	Le 'nom' de la chaîne de caractères n'est pas connu dans le présent système [1)]	
100	Le 'nom' de la chaîne de caractères est un élément du langage de programmation CN, il n'est toutefois actuellement pas programmable (option/fonction est inactive)	
2xx	Le 'nom' de la chaîne de caractères est un élément programmable du langage de programmation CN (option/fonction est active). L'information détaillée xx contient d'autres informations sur le type de l'élément :	
	xx	**Signification**
	01	Adresse DIN ou CN [2)]
	02	Instruction G (par ex. G04, INVCW)
	03	Fonction avec valeur de retour
	04	Fonction sans valeur de retour
	05	Mot-clé (par ex. DEFINE)
	06	Paramètre machine ($M...) , donnée de réglage ($S...) ou donnée optionnelle ($O...)
	07	Paramètre système, par ex. variable système ($...) ou paramètre de calcul (R...)
	08	Cycle (le cycle doit être chargé dans la CN et les programmes de cycle doivent être actifs [3)])
	09	Variable GUD (la variable GUD doit correspondre à celle définie dans les fichiers de définition GUD et les variables GUD doivent être activées)
	10	Nom de macro (le macro doit correspondre à celui défini dans les fichiers de définition Macro et les macros doivent être activés) [4)]
	11	Variable LUD du programme pièce actuel
	12	Instruction G ISO (mode de langage ISO doit être actif)
400	Le 'nom' de la chaîne de caractères est une adresse CN n'ayant pas été reconnue comme xx == 01 ou xx == 10 et n'étant pas G ni R [2)]	
y00	Pas d'affectation spécifique possible	

1) En fonction de la commande, seul un sous-ensemble des instructions de langage CN Siemens est éventuellement connu, par ex. SINUMERIK 802D sl. Sur ces commandes, la valeur 0 est retournée pour des chaînes de caractères n'étant pas principalement des instructions de langage CN Siemens. Ce comportement peut être modifié au moyen de PM10711 $MN_NC_LANGUAGE_CONFIGURATION. Dans le cas de PM10711 = 1, la valeur 100 est toujours retournée pour des instructions de langage CN Siemens.

2) Les lettre suivantes sont des adresses CN : A, B, C, E, I, J, K, Q, U, V, W, X, Y, Z. Ces adresses CN peuvent également être programmées avec une extension d'adresse. L'extension d'adresse peut être indiquée lors du contrôle avec STRINGIS. Exemple : 201 == STRINGIS("A1").
Les lettres : D, F, H, L, M, N, O, P, S, T sont des adresses CN ou des fonctions auxiliaires utilisées de manière définie par l'utilisateur. La valeur 400 est toujours retournée pour ces adresses. Exemple : 400 == STRINGIS("D"). Ces adresses CN ne peuvent pas être indiquées avec extension d'adresse lors du contrôle avec STRINGIS.
Exemple : 000 == STRINGIS("M02"), mais 400 == STRINGIS("M").

3) Les noms de paramètres de cycles ne peuvent pas être contrôlés avec STRINGIS.
4) Des adresses définies comme macro, par ex. G, H, M, L, sont identifiées comme macro

Exemples

Dans les exemples ci-dessous, il est supposé que les éléments de langage CN indiqués comme chaîne de caractères, sauf indication contraire, sont principalement programmables dans la commande.

1. La chaîne de caractères "T" est définie comme fonction auxiliaire :
   ```
   400 == STRINGIS("T")
   000 == STRINGIS ("T3")
   ```
2. La chaîne de caractères "X" est définie comme axe :
   ```
   201 == STRINGIS("X")
   201 == STRINGIS("X1")
   ```
3. La chaîne de caractères "A2" est définie comme adresse CN avec extension :
   ```
   201 == STRINGIS("A")
   201 == STRINGIS("A2")
   ```
4. La chaîne de caractères "INVCW" est définie comme instruction G nommée :
   ```
   202 == STRINGIS("INVCW")
   ```
5. La chaîne de caractères "$MC_GCODE_RESET_VALUES" est définie comme paramètre machine :
   ```
   206 == STRINGIS("$MC_GCODE_RESET_VALUES")
   ```
6. La chaîne de caractères "GETMDACT" est une fonction de langage CN :
   ```
   203 == STRINGIS("GETMDACT ")
   ```
7. La chaîne de caractères "DEFINE" est un mot-clé :
   ```
   205 == STRINGIS("DEFINE")
   ```
8. La chaîne de caractères "$TC_DP3" est un paramètre système (composantes de longueur d'outil) :
   ```
   207 == STRINGIS("$TC_DP3")
   ```
9. La chaîne de caractères "$TC_TP4" est un paramètre système (taille d'outil) :
   ```
   207 == STRINGIS("$TC_TP4")
   ```
10. La chaîne de caractères "$TC_MPP4" est un paramètre système (état de l'emplacement d'outil) :
    - La gestion de magasin d'outils est active : `207 == STRINGIS("$TC_MPP4") ;`
    - La gestion de magasin d'outils n'est pas active : `000 == STRINGIS("$TC_MPP4")`

    Voir également ci-dessous le paragraphe : Gestion de magasin d'outils.
11. La chaîne de caractères "MACHINERY_NAME" est définie comme variable GUD :
    ```
    209 == STRINGIS("MACHINERY_NAME")
    ```
12. La chaîne de caractères "LONGMACRO" est définie comme macro :
    ```
    210 == STRINGIS("LONGMACRO")
    ```
13. La chaîne de caractères "MYVAR" est définie comme variable LUD :
    ```
    211 == STRINGIS("MYVAR")
    ```
14. La chaîne de caractères "XYZ" n'est pas une instruction, variable GUD, nom de macro ni nom de cycle connu dans la CN :
    ```
    000 == STRINGIS("XYZ")
    ```

Gestion de magasin d'outils

Si la fonction de gestion d'outils n'est pas active, STRINGIS fournit pour les paramètres système de la gestion de magasin d'outils, indépendamment du paramètre machine

- PM10711 $MN_NC_LANGUAGE_CONFIGURATION

toujours la valeur 000.

Mode ISO

Si la fonction "Mode ISO" est active :

- PM18800 $MN_MM_EXTERN_LANGUAGE (activation de langages CN externes)
- PM10880 $MN_ MM_EXTERN_CNC_SYSTEM (système de commande devant être adapté)

TRINGIS contrôle la chaîne de caractères indiquée tout d'abord en tant qu'instruction G SINUMERIK. Si la chaîne de caractères n'est pas une instruction G SINUMERIK, elle sera par la suite contrôlée en tant qu'instruction G ISO.

Des commutations programmées (`G290` (mode SINUMERIK), `G291` (mode ISO)) n'ont aucun effet sur STRINGIS.

Exemple

Les importants paramètres machine pour la fonction STRINGIS(...) ont les valeurs suivantes :

- PM10711 $MN_NC_LANGUAGE_CONFIGURATION = 2 (seules les instructions de langage CN dont les options sont configurées sont considérées comme étant reconnues)
- PM19410 $ON_TRAFO_TYPE_MASK = 'H0' (option : transformations)
- PM10700 $MN_PREPROCESSING_LEVEL='H43' (conditionnement pour cycles actif)

Le programme exemplaire ci-dessous est exécuté sans message d'erreur :

Code de programme	**Commentaire**
`N1 R1=STRINGIS("TRACYL")`	`; R1 == 0, étant donné que TRACYL est identifiée comme ''étant non connue" en raison de l'absence de l'option de transformation`
`N2 IF STRINGIS("TRACYL") == 204`	
`N3  TRACYL(1,2,3)`	`; N3 est ignoré`
`N4 ELSE`	
`N5  G00`	`; et N5 est exécuté à la place`
`N6 ENDIF`	
`N7 M30`	

3.24.3 Appel interactif des fenêtres depuis le programme pièce (MMC)

Le sous-programme MMC(...) prédéfini permet d'afficher à partir d'un programme CN des boîtes de dialogue propres à l'utilisateur sur l'interface utilisateur.

La configuration des boîtes de dialogue est possible pour les types de boîte de dialogue suivants :

- Run MyScreens
- Easy XML
- XML utilisateur

Informations complémentaires :

- Manuel de programmation Run MyScreens
- Manuel de programmation Easy XML

Syntaxe

```
MMC("<ADDRESS>,<COMMAND>,<FILE>,<DIALOG>","<QUIT>")
```

Signification

`MMC(...)` :	Descripteur de sous-programme Les paramètres sont indiqués avec un code d'emplacement et séparés par des virgules au sein de deux chaînes de caractères, la chaîne de commande et la chaîne d'acquittement.	
Paramètres au sein de la chaîne de commande :		
`<ADDRESS>` :	Groupe fonctionnel dans lequel sont exécutées les boîtes de dialogue utilisateur qui ont été configurées	
	Fonction	**Groupes fonctionnels**
	Boîte de dialogue utilisateur "Run MyScreens"	`CYCLES`
	Boîte de dialogue utilisateur "Easy XML"	`CYCLES`
	XML utilisateur	`XML`
	Fenêtre contextuelle "Run MyScreens"	`POPUPDLG`
	Fenêtre contextuelle "Easy XML"	`POPUPDLG`
`<COMMAND>` :	Commande à exécuter	
	Fonction	**Commandes**
	Boîte de dialogue utilisateur "Run MyScreens"	`PICTURE_ON`, `PICTURE_OFF`
	Boîte de dialogue utilisateur "Easy XML"	`PICTURE_ON`, `PICTURE_OFF`
	XML utilisateur	`XML_ON`, `XML_OFF`
	Fenêtre contextuelle "Run MyScreens"	`PICTURE_ON`, `PICTURE_OFF`
	Fenêtre contextuelle "Easy XML"	`PICTURE_ON`, `PICTURE_OFF`

<FILE> :	Nom du fichier dans lequel la boîte de dialogue à afficher est programmée	
	Fonction	**Fichiers**
	Boîte de dialogue utilisateur "Run MyScreens"	<nom>.com
	Boîte de dialogue utilisateur "Easy XML"	<nom>.xml
	XML utilisateur	<nom>.xml
	Fenêtre contextuelle "Run MyScreens"	<nom>.com
	Fenêtre contextuelle "Easy XML"	<nom>.xml
	Fenêtre contextuelle "Easy XML" avec configuration directe dans le programme CN (voir exemple 2)	xmldial_emb.xml
<DIALOG> :	Nom de la boîte de dialogue à afficher	
	Fonction	**Nom de la boîte de dialogue**
	Toutes les fonctions à l'exception de Fenêtre contextuelle "Easy XML" avec configuration directe dans le programme CN	Nom configuré de la boîte de dialogue dans le fichier <FILE>
	Fenêtre contextuelle "Easy XML" avec configuration directe dans le programme CN (voir exemple 3)	main
Paramètres au sein de la chaîne d'acquittement :		
<QUIT> :	Type d'acquittement	
	N :	Pas d'acquittement. L'exécution du programme se poursuit après l'envoi de l'ordre. Il n'y a aucune signalisation en retour lorsque la commande n'a pas pu être exécutée avec succès. **Remarque** Le type d'acquittement "N" doit être utilisé lorsqu'une durée d'affichage (arrêt temporisé) est programmée dans le programme CN (voir Exemple 2 ci-dessous)
	A:	Acquittement asynchrone L'exécution du programme se poursuit après l'envoi de l'ordre. La valeur de retour est enregistrée dans une variable d'acquittement (variable GUD) spécifique à l'utilisateur définie dans le cadre de la configuration de la boîte de dialogue et peut être lue dans le programme CN.

Autres conditions à prendre en compte

- Les fichiers de définition *.com des boîtes de dialogue doivent être enregistrés dans le dossier "proj".
- Les fichiers de définition Easy XML *.xml des boîtes de dialogue doivent être enregistrés dans le dossier "appl".
  Si les fichiers de définition sont enregistrés dans un autre répertoire, le chemin d'accès doit être indiqué de façon indirecte en partant du répertoire "appl".
- Les boîtes de dialogue définies par l'utilisateur ne peuvent pas être affichées simultanément à partir de différents canaux.
- La fonctionnalité MMC n'est pas prise en charge dans la simulation.

Exemples

Exemple 1

Affichage d'une boîte de dialogue et réaction à l'action de l'utilisateur dans un programme CN.

```
; La variable d'acquittement QUIT a déjà été créée en tant que variable utilisateur globale (GUD)
; du type STRING lors de la configuration de la boîte de dialogue :
; DEF NCK STRING[20] QUIT
QUIT = "XXX"                                ; Initialiser la variable d'acquittement
G4 F5
MMC("CYCLES,PICTURE_ON,test.com,test1","A")  ; Afficher la boîte de dialogue
                                            ;  - Groupe fonctionnel : CYCLES
                                            ;  - État de l'image : PICTURE_ON (afficher)
                                            ;  - Fichier de boîte de dialogue : test.com
                                            ;  - Boîte de dialogue : test1
INPUT :                                     ; Attente de l'action de l'utilisateur
   STOPRE                                   ;    Arrêt du prétraitement des blocs
   IF MATCH (QUIT,"RUN") >= 0 GOTOF WORK    ;    Touche logicielle "RUN"
   IF MATCH (QUIT,"CHK") >= 0 GOTOF CHECK   ;    Touche logicielle "CHK"
GOTOB INPUT                                 ; => Attente

WORK :                                      ; Touche logicielle "RUN" actionnée
MSG("Poursuivre l'usinage -> Départ CN")    ; Générer signalisation
MMC("CYCLES,PICTURE_OFF","N")               ; Fermer la boîte de dialogue
M0                                          ; Attente du départ CN
GOTOF END                                   ; => Fin du programme

CHECK :                                     ; Touche logicielle "CHK" actionnée
MSG("Accoster la position -> Départ CN")    ; Générer signalisation
MMC("CYCLES,PICTURE_OFF","N")               ; Fermer la boîte de dialogue
M0                                          ; Attente du départ CN
GOTOF END                                   ; => Fin du programme

END :
...
```

Exemple 2

La durée d'affichage d'une boîte de dialogue est définie dans le programme CN, par exemple par le biais d'un arrêt temporisé.

```
F1000 G94
...
```

Code de programme	**Commentaire**
MMC("POPUPDLG,PICTURE_**ON**,xmldial_emb.xml,main","N")	; Afficher la boîte de dialogue
X200	
Z40	
MMC("POPUPDLG,PICTURE_**OFF**","N")	; Fermer la boîte de dialogue

Exemple 3

Intégration d'un script de fenêtre contextuelle dans un programme CN et de son utilisation.

Code de programme

```
PROC POPUP_TEST
; ----------------------------- Script -----------------------------
; <main_dialog entry="rpara_main">
;    <let name="xpos" />
;    <let name="ypos" />
;    <let name="field_name" type="string" />
;    <let name="num" />
;    <menu name="rpara_main">
;       <open_form name="rpara_form"/>
;       <softkey_back>
;          <close_form />
;       </softkey_back>
;    </menu>
;    <form name="rpara_form">
;       <init>
;          <caption>mask from NC part program</caption>
;          <let name="count" >0</let>
;          <op>
;             xpos = 120;
;             ypos = 34;
;             "nck/Channel/Parameter/R[10]" = 10;
;          </op>
;          <!-- load the number of controls -->
;          <op>
;             num = "nck/Channel/Parameter/R[10]";
;          </op>
;          <while>
;             <condition> count < num</condition>
;             <print name="field_name" text="edit%d">count</print>
;             <op>
;                ypos = ypos + 24;
;                count = count + 1;
;             </op>
;          </while>
;       </init>
;       <paint>
```

Code de programme

```
;          <op>
;             xpos =  8;
;             ypos = 36;
;             count = 0;
;          </op>
;          <while>
;             <condition>count < num</condition>
;             <print name="field_name" text="R-Parameter%d">count</print>
;             <text xpos = "$xpos" ypos = "$ypos" >$$$field_name</text>
;             <op>
;                ypos = ypos + 24;
;                count = count + 1;
;             </op>
;          </while>
;       </paint>
;    </form>
; </main_dialog>
; ========================= Partie du programme ============================
...
G94 F100
MMC("POPUPDLG,PICTURE_ON,xmldial_emb.xml,main","N")
G4 F4
X200
MMC("POPUPDLG,PICTURE_OFF","N")
G4 F2
X0
...
```

3.24.4 Process DataShare - Sortie sur un appareil/fichier externe (EXTOPEN, WRITE, EXTCLOSE)

L'écriture de données depuis un programme pièce sur un périphérique/fichier externe s'effectue en trois étapes :

1. Ouverture du périphérique/fichier externe
   L'instruction EXTOPEN permet d'ouvrir le périphérique/fichier externe pour le canal en vue de l'écriture.
2. Écriture de données
   La donnée en sortie peut être préparée avec les fonctions String du langage CN, par exemple SPRINT. L'écriture elle-même s'effectue à l'aide de l'instruction WRITE.
3. Fermeture du périphérique/fichier externe
   L'instruction EXTCLOSE, le fait d'atteindre la fin du programme (M30) ou la réinitialisation du canal permet de libérer de nouveau le périphérique/fichier externe occupé dans le canal.

Syntaxe

```
DEF INT <Result>
DEF STRING[<n>] <Output>
…
EXTOPEN(<Result>,<ExtDev>,<SyncMode>,<AccessMode>,<WriteMode>)
…
<Output>="Sortie données"
WRITE(<Result>,<ExtDev>,<Output>)
…
EXTCLOSE(<Result>,<ExtDev>)
```

Signification

EXTOPEN :	Procédure prédéfinie pour l'ouverture d'un périphérique/fichier externe		
<Result> :	**Paramètre 1 :** variable de résultat La réussite de l'opération peut être contrôlée dans le programme au moyen de la valeur de la variable de résultat, afin de poursuivre le traitement en conséquence.		
	Type :	INT	
	Valeurs :	0	Aucune erreur
		1	Impossible d'ouvrir le périphérique externe
		2	Le périphérique externe n'est pas configuré
		3	Périphérique externe configuré avec un chemin d'accès non valide
		4	Pas de droits d'accès pour le périphérique externe
		5	Mode d'utilisation : périphérique externe déjà affecté de façon "exclusive"
		6	Mode d'utilisation : périphérique externe déjà affecté en mode "shared"
		7	Longueur de fichier supérieure à LOCAL_DRIVE_MAX_FILESIZE
		8	Nombre maximum de périphériques externes dépassé
		9	Option pour LOCAL_DRIVE non activée
		11	Réservé
		12	Mode d'écriture : Indication en contradiction avec extdev.ini
		16	Chemin externe programmé non valide
		22	Périphérique externe non monté

<table>
<tr><td><ExtDev> :</td><td colspan="2">Paramètre 2 : descripteur symbolique pour le périphérique/fichier externe à ouvrir</td></tr>
<tr><td></td><td>Type :</td><td>STRING</td></tr>
<tr><td></td><td colspan="2">Le descripteur symbolique se compose des éléments suivants :
1. nom logique du périphérique
2. éventuellement suivi d'un chemin d'accès au fichier (suffixé avec "/").
Les noms logiques de périphériques suivants sont définis :</td></tr>
<tr><td></td><td>"LOCAL_DRIVE" :</td><td>Carte CompactFlash locale (prédéfinie)</td></tr>
<tr><td></td><td>"CYC_DRIVE" :</td><td>Indication de lecteur réservée pour l'utilisation dans les cycles SIEMENS (prédéfini)</td></tr>
<tr><td></td><td>"/dev/ext/1", ...
"/dev/ext/9" :</td><td>Lecteurs réseau disponibles
Remarque :
Configuration requise dans le fichier extdev.ini !</td></tr>
<tr><td></td><td>"/dev/cyc/1",
"/dev/cyc/2" :</td><td>Indications de lecteurs réservées pour l'utilisation dans les cycles SIEMENS
Remarque :
Configuration requise dans le fichier extdev.ini !</td></tr>
<tr><td></td><td colspan="2">Chemin d'accès :
• Pour "LOCAL_DRIVE" et "CYC_DRIVE", un chemin d'accès doit être indiqué, par exemple :
"LOCAL_DRIVE/my_dir/my_file.txt"
• Les noms logiques de périphériques "/dev/ext/1...9" et "/dev/cyc/1...2" peuvent être configurés pour :
– faire référence à un fichier, auquel cas il suffit d'indiquer le nom logique du périphérique, par exemple :
"/dev/ext/4"
– ou faire référence à un répertoire, auquel cas un chemin d'accès est nécessaire, par exemple :
"/dev/ext/5/my_dir/my_file.txt"</td></tr>
<tr><td></td><td colspan="2">Remarque :
Pour les noms logiques de périphériques "/dev/ext/1...9", "/dev/v24" et "/dev/cyc/1...2", aucune distinction n'est faite entre minuscules et majuscules, mais pour l'indication d'un chemin d'accès à un fichier, la distinction est significative. Pour "LOCAL_DRIVE" et "CYC_DRIVE", seules les majuscules sont autorisées.</td></tr>
</table>

<SyncMode> :	**Paramètre 3 :** Mode de traitement des instructions WRITE à destination de ce périphérique/fichier		
	Type :	STRING	
	Valeurs :	"SYN" :	Écriture synchrone L'exécution du programme est interrompue jusqu'à ce que le processus d'écriture soit terminé. Le succès de l'écriture synchrone peut être contrôlé par l'évaluation des variables d'erreur de l'instruction WRITE.
		"ASYN" :	Écriture asynchrone L'exécution du programme n'est pas interrompue par l'instruction WRITE. **Remarque :** La variable de résultat de l'instruction WRITE n'est pas décisive dans ce mode et a toujours la valeur 0 (aucune erreur). Dans ce mode, il n'y a pas de certitude que l'instruction WRITE ait réussi.
<AccessMode> :	**Paramètre 4 :** Mode d'utilisation de ce périphérique/fichier		
	Type :	STRING	
	Valeurs :	"SHARED" :	Le périphérique/fichier est requis dans le mode "Shared". D'autres canaux peuvent également utiliser le périphérique, c'est à dire s'ouvrir également dans ce mode.
		"EXCL" :	Le périphérique/fichier est utilisé exclusivement dans le canal. Aucun autre canal ne peut également utiliser le périphérique.
<WriteMode> :	**Paramètre 5 :** Mode d'écriture pour les instructions WRITE à destination de ce périphérique/fichier (facultatif)		
	Type :	STRING	
	Valeurs :	"APP" :	Ajout Le fichier reste intact quant à son contenu. Les requêtes d'écriture s'ajoutent à la fin.
		"OVR" :	Écraser Le contenu du fichier est effacé et recréé par les appels d'écriture suivants.
	Remarque : Ce paramètre ne permet pas d'écraser le mode d'écriture configuré dans le fichier extdev.ini. En cas de conflit, l'appel EXTOPEN est acquitté avec génération d'une erreur.		

WRITE :	Procédure prédéfinie pour l'écriture des données de sortie

EXTCLOSE :	Procédure prédéfinie pour la fermeture d'un périphérique/fichier externe ouvert		
<Result> :	**Paramètre 1 :** variable de résultat		
	Type :	INT	
	Valeurs :	0	Aucune erreur
		16	Chemin externe programmé non valide
		21	Erreur lors de la fermeture d'un périphérique externe
<ExtDev> :	**Paramètre 2 :** descripteur symbolique pour le périphérique/fichier externe à fermer : description, voir sous EXTOPEN. **Remarque :** Le descripteur doit être identique à celui indiqué dans l'appel EXTOPEN.		

Exemple

Code de programme

```
N10       DEF INT RESULT
N20       DEF BOOL EXTDEVICE
N30       DEF STRING[80] OUTPUT
N40       DEF INT PHASE
N50       EXTOPEN(RESULT,"LOCAL_DRIVE/my_file.txt","SYN","SHARED")
N60       IF RESULT > 0
N70          MSG("Erreur avec EXTOPEN :" << RESULT)
N80       ELSE
N90          EXTDEVICE=TRUE
N100      ENDIF
…
N200      PHASE=4
N210      IF EXTDEVICE
N220         OUTPUT=SPRINT("Fin de la phase : %D",PHASE)
N230         WRITE(RESULT,"LOCAL_DRIVE/my_file.txt",OUTPUT)
N240      ENDIF
…
```

Voir aussi

Opérations sur les chaînes de caractères (Page 457)

Ecriture d'un fichier (WRITE) (Page 578)

3.24.5 Alarmes (SETAL)

Les alarmes peuvent être définies dans un programme CN. Elles sont affichées dans une fenêtre particulière de l'interface utilisateur. A une alarme correspond obligatoirement une réaction de la commande qui dépend de la catégorie à laquelle appartient l'alarme.

Syntaxe

```
SETAL(<n°>[,<chaîne>])
```

Signification

SETAL(...) :	Procédure prédéfinie pour l'activation d'une alarme SETAL doit être programmé dans un bloc CN spécifique.		
<N°> :	Numéro d'alarme		
	Type de données :	INT	
	Plage de valeurs :	60000 ... 64999 (réservé)	Alarmes pour cycles SIEMENS
		65000 ... 69999	Alarmes pour cycles utilisateur
<chaîne> :	Chaîne de caractères (en option)		
	Type de données :	STRING	
	Lors de la programmation d'alarmes de cycle utilisateur, il est possible d'indiquer une chaîne de caractères comportant jusqu'à 4 paramètres. Dans ces paramètres, il est possible de définir des textes utilisateur variables. Des paramètres prédéfinis sont également disponibles :		
	%1	Eine ausschließlich von der PLC kontrollierte Achse kannnuméro de canal	
	%2	numéro de bloc, étiquette	
	%3	Indice de texte pour alarmes de cycles	
	%4	Paramètre d'alarme supplémentaire	

Remarque

Les textes d'alarme doivent être configurés dans l'interface utilisateur.

Remarque

Si les alarmes générées doivent s'afficher dans la langue activée pour l'interface utilisateur, l'utilisateur doit disposer d'informations sur la langue actuellement réglée sur l'IHM. Cette information peut être interrogée dans le programme pièce et dans les actions synchrones, au moyen de la variable système $AN_LANGUAGE_ON_HMI (Page 1396).

Exemple

Code de programme	Commentaire
...	
N100 SETAL (65000)	; Activer l'alarme n° 65000
...	

Plus d'informations

Réaction et acquittement d'alarme

Les alarmes de cycle utilisateur sont affectées à des plages de numéros qui se distinguent par la réaction et l'acquittement d'alarme :

Plage de numéros	Réaction d'alarme	Acquittement d'alarme
65000 - 65499	Affichage, verrouillage Départ CN	Reset
65500 - 65999	Affichage, verrouillage Départ CN (pas pour les ASUP lorsque le MD20194 est mis à 1)	Reset
66000 - 66999	Affichage, verrouillage Départ CN, immobilisation après exécution des blocs prédécodés	Reset
67000 - 67999	Affichage	Annuler
68000 - 68999	Affichage, verrouillage Départ CN, arrêt immédiat de l'interpolateur	Reset
69000 - 69999	Affichage, verrouillage Départ CN, arrêt à la fin de bloc suivante	Reset

3.24.6 Définition d'une pièce brute (WORKPIECE)

La commande doit connaître la forme et la taille d'une pièce brute pour pouvoir l'afficher dans la simulation graphique. L'utilisateur a donc la possibilité de définir des pièces brutes via l'interface utilisateur ou directement dans le programme CN. Les définitions de pièces brutes sont maintenues après un reset (fin de programme, canal, GMFC). Elles sont supprimées automatiquement au démarrage suivant de la commande.

Syntaxe

```
WORKPIECE("<WP>", "<RefP>", "<ZeroOffset>", "<Type>", <Par5>,
<Par6>, ..., <Par12>)
```

Signification

`WORKPIECE(...)` :	Procédure prédéfinie pour la définition d'une pièce brute	
	Arrêt du prétraitement des blocs :	oui
	Uniquement dans le bloc :	oui
Paramètres :		

1	"<WP>" :	Nom de la pièce (facultatif)		
		Type de données :	STRING	
		L'indication est nécessaire uniquement lorsqu'il peut y avoir plusieurs pièces dans un canal. Sans indication, "WORKP<n>" est automatiquement pris en compte, où <n> correspond au numéro du canal déclaratif.		
2	"<RefP>" :	Ablocage (facultatif ; uniquement pour les fraiseuses)		
		Type de données :	STRING	
		Plage de valeurs :	"Plateau"	Ablocage sur le plateau fixe
			"A"	Ablocage sur l'axe rotatif A
			"B"	Ablocage sur l'axe rotatif B
			"C"	Ablocage sur l'axe rotatif C
		Condition : Le plateau ou l'axe rotatif doit être débloqué pour l'ablocage de pièce brute via le paramètre machine correspondant (voir Manuel de mise en service SINUMERIK Operate).		
3	"<ZeroOffset>" :	Décalage d'origine réglable pour le placement de la pièce brute (non programmable) La sélection d'un décalage d'origine réglable pour le placement de la pièce brute est proposée uniquement lors de l'introduction de pièce brute via l'interface utilisateur. Au lieu de cela, en cas de définition directe de la pièce brute dans le programme pièce, la pièce brute se rapporte toujours au décalage d'origine momentanément en vigueur.		
4	"<Type>" :	Forme de la pièce brute		
		Type de données :	STRING	
		Plage de valeurs :	"CYLINDER" :	Cylindre
			"PIPE" :	Tube
			"RECTANGLE" :	Parallélépipède centré
			"BOX" :	Parallélépipède rectangulaire
			"N_CORNER" :	Polygone
5 ... 12	<Par5> ... <Par12> :	Paramètre de description de la forme de la pièce brute		
		Type de données :	REAL	
		Le nombre de paramètres nécessaires et leur signification dépendent de la forme de la pièce brute et de la valeur du paramètre de bit. Voir : • Tableau "Paramètre de description de la forme de la pièce brute" • Tableau "Paramètre de bit"		
WORPIECE() :		Un appel de WORKPIECE sans paramètre supprime toutes les définitions de pièces brutes.		
WORPIECE(<WP>) :		Un appel de WORKPIECE avec nom de pièce supprime uniquement cette définition de pièce brute.		

Tableau 3-6 Paramètre de description de la forme de la pièce brute

Forme de la pièce brute	**Paramètres**							
	<Par5>	**<Par6>**	**<Par7>**	**<Par8>**	**<Par9>**	**<Par10>**	**<Par11>**	**<Par12>**
Cylindre	Paramètre de bit Valeur réelle interprétée comme nombre entier codé sur bits. Les bits définissent la signification des paramètres suivants (voir tableau "Paramètre de bit").	Point de référence Z_0	Longueur Z_1	Cote d'usinage Z_B	Diamètre extérieur d_0	-	Rotation autour de l'axe rotatif	-
Tube		Point de référence Z_0	Longueur Z_1	Cote d'usinage Z_B	Diamètre extérieur d_0	Épaisseur de paroi (rel) / Diamètre intérieur d_1 (abs)	Rotation autour de l'axe rotatif	-
Parallélépipède centré		Point de référence Z_0	Longueur Z_1	Cote d'usinage Z_B	Largeur W	Longueur L	Rotation autour de l'axe rotatif	-
Parallélépipède rectangulaire		Point de référence Z_0	Longueur Z_1	Cote d'usinage Z_B	X_0	Y_0	X_1	Y_1
Polygone		Point de référence Z_0	Longueur Z_1	Cote d'usinage Z_B	Nombre de coins	Cote sur plats	Rotation autour de l'axe rotatif	-

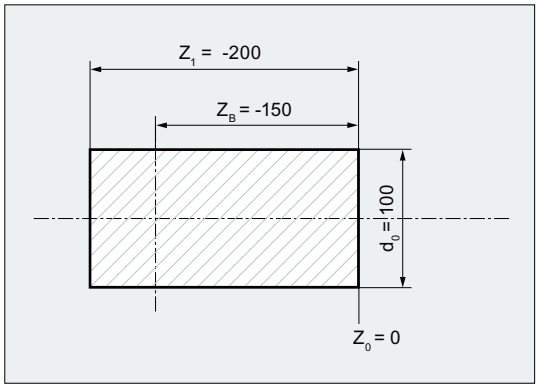
Tableau 3-7 Paramètre de bit

Bit	**Signification**	
4 (0x0010)	Parallélépipède rectangulaire : X_1	
	= 0	rel
	= 1	abs
5 (0x0020)	Parallélépipède rectangulaire : Y_1	
	= 0	rel
	= 1	abs
6 (0x0040)	Longueur Z_1 (cote finale)	
	= 0	rel
	= 1	abs
Bit 7 (0x0080)	Cote d'usinage Z_B	
	= 0	rel
	= 1	abs
Bit 8 (0x0100)	Tube : Épaisseur de paroi / Diamètre intérieur	
	= 0	rel
	= 1	abs
9 (0x0200)	Polygone	
	= 0	Cote sur plats
	= 1	Longueur d'arête

Bit	Signification	
12 (0x1000)	Ablocage pour tours	
	= 0	Broche principale
	= 1	Contre-broche
13 (0x2000)	Contre-broche	
	= 0	Avec fonction miroir
	= 1	Sans fonction miroir

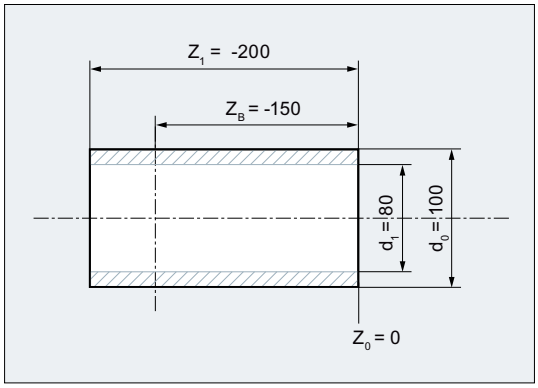
Exemples

Exemple 1 : Pièce brute cylindrique sur un tour



Code de programme	Commentaire
...	
WORKPIECE(,,,"CYLINDER",0,0,-200,-150,100)	; **Définition de la pièce brute :**
	; Forme de la pièce brute : Cylindre
	; Paramètre de bit=0(aucun bit activé) → Les valeurs de longueur et de cote d'usinage sont relatives, pièce brute sur la broche principale
	; Point de référence(Z0)=0
	; Longueur(Z1)=-200
	; Cote d'usinage(ZB)=-150
	; Diamètre extérieur(d0)=100
...	

Exemple 2 : Pièce brute tubulaire sur un tour



Code de programme	Commentaire
...	
WORKPIECE(,,,"PIPE",256,0,-200,-150,100,80)	**; Définition de la pièce brute :** ; Forme de la pièce brute : Tube ; Paramètre de bit=256(bit8=1) → Le diamètre intérieur est absolu, la longueur et la cote d'usinage sont relatives, pièce brute sur la broche principale ; Point de référence(Z0)=0 ; Longueur(Z1)=-200 ; Cote d'usinage(ZB)=-150 ; Diamètre extérieur(d0)=100 ; Diamètre intérieur(d1)=80
...	

3.24.7 Basculement du mode de langage (G290, G291)

La commande offre la possibilité de lire et d'exécuter des programmes pièce à partir de systèmes CNC externes. La condition est que le mode de langage CN (dialecte ISO) ait été défini lors de la mise en service.

Le mode en dialecte ISO peut être activé séparément pour chaque canal. Par exemple, le canal 1 peut fonctionner dans le mode en dialecte ISO tandis que le canal 2 est en mode SINUMERIK.

Le basculement entre le mode SINUMERIK et le mode en dialecte ISO s'effectue dans le programme CN via les instructions du groupe G 47. Le basculement n'influence pas l'outil actif, les corrections d'outil et les décalages d'origine.

Syntaxe

```
G291
...
G290
```

Signification

G290:	Activer mode de langage SINUMERIK	
	Uniquement dans le bloc :	oui
	Prise d'effet :	modale
G291:	Activer mode de langage ISO	
	Uniquement dans le bloc :	oui
	Prise d'effet :	modale

Conditions

Mode SINUMERIK

- Le préréglage des instructions G peut être défini pour chaque canal via un paramètre machine.
- Le mode SINUMERIK ne permet pas de programmation d'instructions en dialectes ISO.

Mode en dialecte ISO

- En tant que réglage de base de la commande, le mode en dialecte ISO peut être réglé par le biais de paramètres machine. Par défaut, la commande démarre ensuite dans le mode en dialecte ISO.
- Il est uniquement possible de programmer des instructions G du dialecte ISO. La programmation de fonctions SINUMERIK est impossible dans le mode en dialecte ISO.
- Une combinaison de dialecte ISO et de langage SINUMERIK est impossible dans un même bloc CN.
- Un basculement entre le dialecte ISO M (fraisage) et le dialecte ISO T (tournage) par le biais d'instructions G est impossible.
- L'appel de sous-programmes programmés en mode SINUMERIK est possible.
- Lorsque des fonctions SINUMERIK doivent être utilisées, il faut d'abord basculer en mode SINUMERIK (voir exemple).

Exemple

Compactage de blocs linéaires dans le mode en dialecte ISO

Code de programme	Commentaire
N5 G290	; Activer le mode de langage SINUMERIK.
N10 COMPCAD	; COMPCAD est une instruction du langage Siemens activant une fonction de compactage qui remplace des blocs linéaires consécutifs par des blocs de polynôme avec des longueurs de trajectoire aussi grandes que possible.
N15 G291	; Activer le mode de langage ISO.

Code de programme	Commentaire
N20 G01 X100 Y100 F1000	; Étant donné que COMPCAD a été activé en mode SINUMERIK, cette fonction peut également comprimer des blocs linéaires dans le mode en dialecte ISO.
...	

Informations complémentaires

Description fonctionnelle Langage ISO

3.25 Programmes de chariotage personnalisés

3.25.1 Fonctions additionnelles pour le chariotage

Pour le chariotage, on vous propose des cycles d'usinage prêts à l'usage. En plus, vous avez la possibilité de créer vos propres cycles de chariotage grâce aux fonctions élaborées suivantes :

- Création d'une table de contour (CONTPRON)
- Création d'une table de contour codée (CONTDCON)
- Désactivation de la préparation du contour (EXECUTE)
- Détermination de l'intersection entre deux éléments de contour (INTERSEC) (uniquement pour des tables créées avec CONTPRON)
- Exécution des éléments de contour d'une table bloc par bloc (EXECTAB) (uniquement pour des tables créées avec CONTPRON)
- Calcul de données de cercles (CALCDAT)

Remarque

Ces fonctions sont utilisables de façon universelle et pas seulement pour le chariotage.

Conditions

Avant tout appel des fonctions CONTPRON ou CONTDCON, il faut :

- accoster un point de départ permettant un usinage sans collision,
- désactiver la correction du rayon du tranchant avec `G40`.

3.25.2 Création d'une table de contour (CONTPRON)

La fonction `CONTPRON` permet d'activer la préparation du contour. Les blocs CN appelés par la suite ne sont pas exécutés, mais décomposés en déplacements élémentaires qui sont rangés dans la table de contour. A chaque élément de contour correspond une ligne de la table de contour à deux dimensions. Le nombre de détalonnages qui a été déterminé est indiqué en retour.

Syntaxe

Activer la préparation du contour :
```
CONTPRON(<table de contour>,<type d'usinage>,<détalonnages>,
<sens d'usinage>)
```

Désactiver la préparation du contour et revenir au mode de traitement normal :
```
EXECUTE(<ERREUR>)
```

Voir "Désactivation de la préparation du contour (EXECUTE) (Page 1059)"

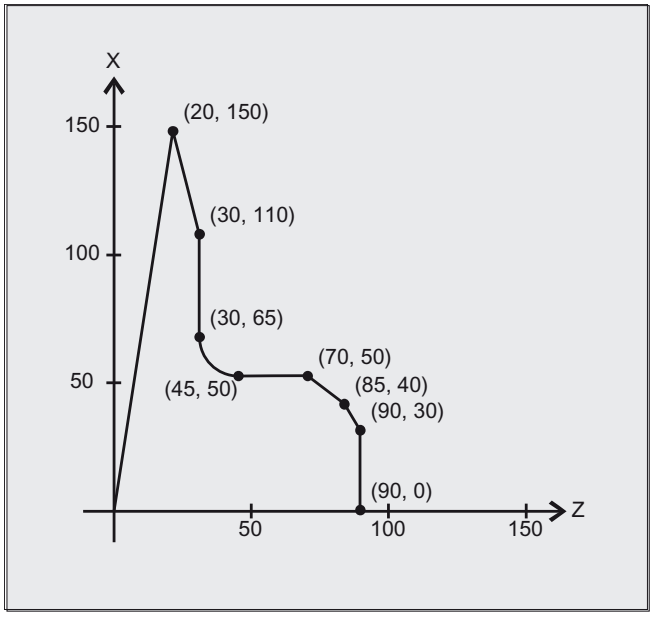
Signification

CONTPRON :	Procédure prédéfinie pour activer la préparation du contour pour la création d'une table de contour		
<Table de contour> :	Nom de la table de contour		
<Type d'usinage> :	Paramètre pour le type d'usinage		
	Type :	CHAR	
	Valeur :	"G" :	longitudinale : usinage intérieur
		"L" :	longitudinale : usinage extérieur
		"N" :	transversale : usinage intérieur
		"P" :	transversale : usinage extérieur
<Détalonnages> :	Variable de résultat pour le nombre d'éléments de détalonnage générés		
	Type :	INT	
<Sens d'usinage> :	Paramètre pour le sens d'usinage		
	Type :	INT	
	Valeur :	0	Préparation du contour en avant (valeur par défaut)
		1	Préparation du contour dans les deux sens

Exemple 1

Création d'une table de contour avec :

- le nom "TABC"
- 30 éléments de contour au maximum (arcs de cercle, segments de droite)
- une variable pour le nombre d'éléments de détalonnage générés
- une variable pour les messages d'erreur



Programme CN :

Code de programme	**Commentaire**
N10 DEF REAL TABC[30,11]	; Table de contour nommée TABC avec 30 éléments de contour au maximum. La valeur de paramètre 11 (nombre de colonnes de la table) est une grandeur fixe.
N20 DEF INT NBDETAL	; Variable pour le nombre d'éléments de détalonnage, nommée NBDETAL.
N30 DEF INT ERREUR	; Variable pour la signalisation d'erreur (0=aucune erreur, 1=erreur).
N40 G18	
N50 CONTPRON(TABC,"G",NBDETAL)	; Activer la préparation du contour.
N60 G1 X150 Z20	; N60 à N120 : Description du contour
N70 X110 Z30	
N80 X50 RND=15	
N90 Z70	
N100 X40 Z85	
N110 X30 Z90	
N120 X0	
N130 EXECUTE(ERREUR)	; Fin du remplissage de la table de contour et retour au mode de programme normal.
N140 …	; Suite du traitement de la table.

Table de contour TABC :

Indice Ligne	Colonne									
(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
7	7	11	0	0	20	150	0	82.40535663	0	0
0	2	11	20	150	30	110	-1111	104.0362435	0	0
1	3	11	30	110	30	65	0	90	0	0
2	4	13	30	65	45	50	0	180	45	65
3	5	11	45	50	70	50	0	0	0	0
4	6	11	70	50	85	40	0	146.3099325	0	0
5	7	11	85	40	90	30	0	116.5650512	0	0
6	0	11	90	30	90	0	0	90	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

Signification du contenu des colonnes :

(0) Index pointant l'élément de contour suivant (le numéro de ligne de la même table)
(1) Index pointant l'élément de contour précédent

(2) Codage du mode de contour pour le déplacement
Valeurs possibles pour X = abc
a = 10^2 G90 = 0 G91 = 1
b = 10^1 G70 = 0 G71 = 1
c = 10^0 G0 = 0 G1 = 1 G2 = 2 G3 = 3

(3), (4) Point de départ des éléments de contour
(3) = abscisse, (4) = ordonnée dans le plan courant

(5), (6) Point final des éléments de contour
(5) = abscisse, (6) = ordonnée dans le plan courant

(7) Indicateur maxi/mini : indique les maxima et minima locaux du contour

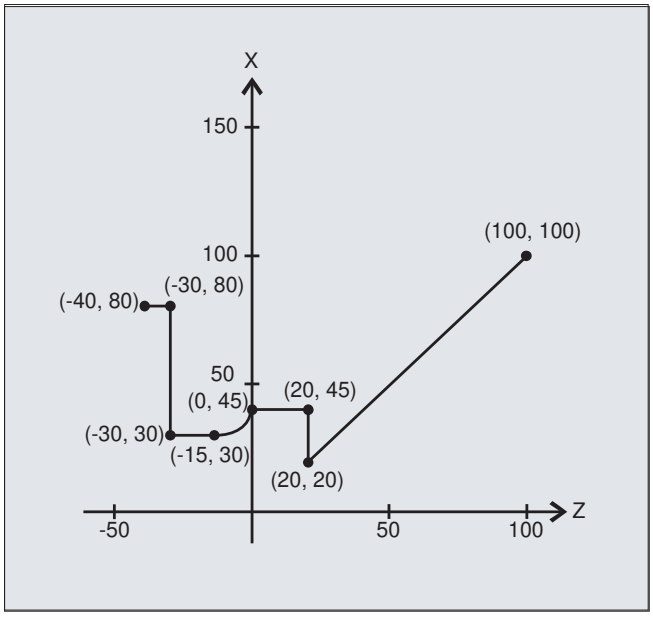
(8) Valeur maximale entre élément de contour et abscisse (dans le cas d'un usinage longitudinal) ou ordonnée (en surfaçage). L'angle dépend du type d'usinage programmé.

(9), (10) Coordonnées du centre de l'élément de contour, si c'est un arc de cercle..
(9) = abscisse, (10) = ordonnée

Exemple 2

Création d'une table de contour avec

- le nom TABC
- 92 éléments de contour au maximum (arcs de cercle, segments de droite)
- Mode d'usinage : cylindrage, usinage extérieur
- Préparation dans les deux sens



Programme CN :

Code de programme	**Commentaire**
N10 DEF REAL TABC[92,11]	; Table de contour nommée TABC avec 92 éléments de contour au maximum. La valeur de paramètre 11 est une grandeur fixe.
N20 DEF CHAR MU="L"	; Mode de fonctionnement pour CONTPRON : cylindrage, usinage extérieur
N30 DEF INT ED=0	; Nombre d'éléments de détalonnage=0
N40 DEF INT MODE=1	; Préparation dans les deux sens
N50 DEF INT ERR=0	; Signalisation d'erreur
...	
N100 G18 X100 Z100 F1000	
N105 CONTPRON(TABC,BT,HE,MODE)	; Activer la préparation du contour.
N110 G1 G90 Z20 X20	
N120 X45	
N130 Z0	
N140 G2 Z-15 X30 K=AC(-15) I=AC(45)	
N150 G1 Z-30	
N160 X80	
N170 Z-40	
N180 EXECUTE(ERR)	; Fin du remplissage de la table de contour et retour au mode de programme normal.
...	

Table de contour TABC :

Après la préparation du contour, le contour est disponible dans les deux sens.

Indice	Colonne										
Ligne	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
0	6[1)]	7[2)]	11	100	100	20	20	0	45	0	0
1	0[3)]	2	11	20	20	20	45	-3	90	0	0
2	1	3	11	20	45	0	45	0	0	0	0
3	2	4	12	0	45	-15	30	5	90	-15	45
4	3	5	11	-15	30	-30	30	0	0	0	0
5	4	7	11	-30	30	-30	45	-1111	90	0	0
6	7	0[4)]	11	-30	80	-40	80	0	0	0	0
7	5	6	11	-30	45	-30	80	0	90	0	0
8	1[5)]	2[6)]	0	0	0	0	0	0	0	0	0
	...										
83	84	0[7)]	11	20	45	20	80	0	90	0	0
84	90	83	11	20	20	20	45	-1111	90	0	0
85	0[8)]	86	11	-40	80	-30	80	0	0	0	0
86	85	87	11	-30	80	-30	30	88	90	0	0
87	86	88	11	-30	30	-15	30	0	0	0	0

88	87	89	13	-15	30	0	45	-90	90	-15	45
89	88	90	11	0	45	20	45	0	0	0	0
90	89	84	11	20	45	20	20	84	90	0	0
91	83[9)]	85[10)]	11	20	20	100	100	0	45	0	0

Signification du contenu des colonnes et des remarques concernant les lignes 0, 1, 6, 8, 83, 85 et 91

Les significations du contenu des colonnes données dans l'exemple 1 sont valables.

Toujours dans ligne 0 :

1) Précédent : ligne n contient la fin du contour en avant

2) Suivant : ligne n est la fin de la table de contour en avant

Une fois pour les éléments de contour en avant :

3) Précédent : début du contour (en avant)

4) Suivant : fin du contour (en avant)

Toujours dans ligne fin de table de contour (en avant) +1 :

5) Précédent : nombre de détalonnages en avant

6) Suivant : nombre de détalonnages en arrière

Une fois pour les éléments de contour en arrière :

7) Suivant : fin du contour (en arrière)

8) Précédent : début du contour (en arrière)

Toujours dans la dernière ligne de la table :

9) Précédent : ligne n est le début de la table de contour (en arrière)

10) Suivant : ligne n contient le début du contour (en arrière)

Informations complémentaires

Instructions de déplacements autorisées, système de coordonnées

Pour la programmation du contour, les instructions G suivantes sont autorisés :

- Groupe G 1 : `G0`, `G1`, `G2`, `G3`

Possibles aussi :

- Arrondi et chanfrein
- Programmation de cercle avec `CIP` et `CT`

Les fonctions spline, polynôme et filetage provoquent des erreurs.

Entre `CONTPRON` et `EXECUTE`, la modification du système de coordonnées par activation d'un frame est illicite. Ceci est également valable pour la permutation entre `G70` et `G71` ou entre `G700` et `G710`.

Pendant la préparation de la table de contour , une permutation d'axe avec `GEOAX` entraîne une alarme.

Eléments de détalonnage

La description des différents éléments de détalonnage peut être réalisée sous la forme d'un sous-programme ou de blocs isolés, au choix.

Chariotage indépendamment du sens de contour programmé

La préparation du contour avec CONTPRON a été étendue de sorte qu'après son appel, la table de contour est disponible quel que soit le sens programmé.

3.25.3 Création d'une table de contour codée (CONTDCON)

En cas d'activation de la préparation du contour avec CONTDCON, les blocs CN appelés par la suite sont enregistrés sous forme codée dans une table de contour de 6 colonnes pour simplifier la mémorisation. A chaque élément de contour correspond une ligne de la table de contour. Compte tenu des règles de codage indiquées ci-dessous, vous pouvez créer des programmes en code DIN à partir des lignes de la table, par exemple pour des cycles. Dans la ligne numéro 0 sont enregistrées les données du point de départ.

Syntaxe

Activer la préparation du contour :
`CONTDCON(<table de contour>,<sens d'usinage>)`

Désactiver la préparation du contour et revenir au mode de traitement normal :
`EXECUTE(<ERREUR>)`

Voir "Désactivation de la préparation du contour (EXECUTE) (Page 1059)"

Signification

CONTDCON :	Procédure prédéfinie pour activer la préparation du contour pour la création d'une table de contour codée		
<Table de contour> :	Nom de la table de contour		
<Sens d'usinage> :	Paramètre pour le sens d'usinage		
	Type :	INT	
	Valeur :	0	Préparation du contour suivant la séquence de blocs de contour (valeur par défaut)
		1	non admis

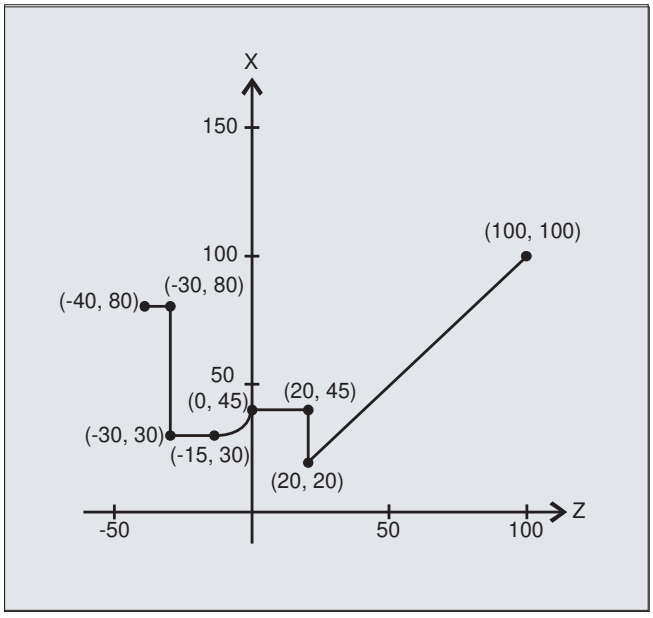
Remarque

Pour CONTDCON, les instructions G admises dans la section de programme à décoder sont plus nombreuses que pour la fonction CONTPRON. En outre, l'avance et le type d'avance sont enregistrés pour chaque élément de contour.

Exemple

Création d'une table de contour avec :

- le nom "TABC"
- les éléments de contour (arcs de cercle, segments de droite)
- Mode d'usinage : tournage
- Sens d'usinage : vers l'avant



Programme CN :

Code de programme	Commentaire
N10 DEF REAL TABC[9,6]	; Table de contour nommée TABC à 9 lignes. 8 blocs de contour sont donc possibles. La valeur 6 (nombre de colonnes de la table) est imposée.
N20 DEF INT MODE = 0	; Variable pour le sens d'usinage. Valeur par défaut 0 : uniquement dans le sens de contour programmé.
N30 DEF INT ERROR = 0	; Variable pour la signalisation d'erreur.
...	
N100 G18 G64 G90 G94 G710	
N101 G1 X100 Z100 F1000	
N105 CONTDCON (TABC, MODE)	; Appel de la préparation du contour (MODE peut être omis).
N110 G1 Z20 X20 F200	; Description du contour.
N120 G9 X45 F300	
N130 Z0 F400	
N140 G2 Z-15 X30 K=AC(-15) I=AC(45)F100	

Code de programme	**Commentaire**
N150 G64 Z-30 F600	
N160 X80 F700	
N170 Z-40 F800	
N180 EXECUTE(ERROR)	; Fin du remplissage de la table de contour et retour au mode de programme normal.
...	

Table de contour TABC :

	Indice de colonne					
	0	1	2	3	4	5
Indice ligne	Mode de contour	Point final abscisse	Point final ordonnée	Centre abscisse	Centre ordonnée	Avance
0	30	100	100	0	0	7
1	11031	20	20	0	0	200
2	111031	20	45	0	0	300
3	11031	0	45	0	0	400
4	11032	-15	30	-15	45	100
5	11031	-30	30	0	0	600
6	11031	-30	80	0	0	700
7	11031	-40	80	0	0	800
8	0	0	0	0	0	0

Signification du contenu des colonnes :

Ligne 0 : Codes du **point de départ :**

Colonne 0 : 10^0 (position des unités) : G0 = 0
10^1 (position des dizaines) : G70 = 0, G71 = 1, G700 = 2, G710 = 3
Colonne 1 : point de départ abscisse
Colonne 2 : point de départ ordonnée
Colonne 3-4 : 0
Colonne 5 : indice de ligne du dernier élément de contour dans la table

Lignes 1-n : Valeurs des **éléments de contour**

Colonne 0 : 10^0 (position des unités) : G0 = 0, G1 = 1, G2 = 2, G3 = 3
10^1 (position des dizaines) : G70 = 0, G71 = 1, G700 = 2, G710 = 3
10^2 (position des centaines) : G90 = 0, G91 = 1
10^3 (position des milliers) : G93 = 0, G94 = 1, G95 = 2, G96 = 3
10^4 (position des dizaines de milliers) : G60 = 0, G44 = 1, G641 = 2, G642 = 3
10^5 (position des centaines de milliers) : G9 = 1
Colonne 1 : Point final abscisse
Colonne 2 : Point final ordonnée
Colonne 3 : centre abscisse en interpolation circulaire

Colonne 4 : centre ordonnée en interpolation circulaire
Colonne 5 : Avance

Informations complémentaires

Instructions de déplacements autorisées, système de coordonnées

Pour la programmation du contour, les groupes G et les instructions G suivants sont autorisés :

Groupe G 1 : G0, G1, G2, G3
groupe G 10 : G60, G64, G641, G642
groupe G 11 : G9
groupe G 13 : G70, G71, G700, G710
groupe G 14 : G90, G91
groupe G 15 : G93, G94, G95, G96, G961

Possibles aussi :

- Arrondi et chanfrein
- Programmation de cercle avec CIP et CT

Les fonctions spline, polynôme et filetage provoquent des erreurs.

Entre CONTDCON et EXECUTE, la modification du système de coordonnées par activation d'un frame est illicite. Ceci est également valable pour la permutation entre G70 et G71 ou entre G700 et G710.

Pendant la préparation de la table de contour , une permutation d'axe avec GEOAX entraîne une alarme.

Sens d'usinage

La table de contour générée avec CONTDCON est prévue pour le chariotage dans le sens programmé du contour.

3.25.4 Détermination de l'intersection entre deux éléments de contour (INTERSEC)

INTERSEC détermine le point d'intersection de deux éléments de contour normés de la table de contour créée avec CONTPRON.

Syntaxe

```
<Etat>=INTERSEC(<table de contour_1>[<élément de contour_1>],
<table de contour_2>[<élément de contour_2>],<intersection>,<type
d'usinage>)
```

Signification

INTERSEC :	Fonction prédéfinie pour déterminer l'intersection de deux éléments de contour issus de tables de contour générées avec CONTPRON		
<état> :	Variable de l'état de l'intersection		
	Type :	BOOL	
	Valeur :	TRUE	existence d'une intersection
		FALSE	pas d'intersection
<table de contour_1> :	Nom de la première table de contour		
<élément de contour_1> :	Numéro de l'élément de contour de la première table de contour		
<table de contour_2> :	Nom de la deuxième table de contour		
<élément de contour_2> :	Numéro de l'élément de contour de la deuxième table de contour		
<intersection> :	Coordonnées du point d'intersection dans le plan actif (G17 / G18 / G19)		
	Type :	REAL	
<Type d'usinage> :	Paramètre pour le type d'usinage		
	Type :	INT	
	Valeur :	0	Calcul du point d'intersection dans le plan activé avec le paramètre 2 (valeur par défaut)
		1	Calcul du point d'intersection indépendamment du plan transmis

Remarque

Les variables doivent avoir été définies avant leur utilisation.

La transmission des contours exige le respect des valeurs définies avec CONTPRON :

Paramètre	Signification
2	Codage du mode de contour pour le déplacement
3	Abscisse du point de départ du contour
4	Ordonnée du point de départ du contour
5	Abscisse du point final du contour
6	Ordonnée du point final du contour
9	Coordonnées du centre pour l'abscisse (uniquement pour le contour du cercle)
10	Coordonnées du centre pour l'ordonnée (uniquement pour le contour du cercle)

Exemple

Il s'agit de déterminer l'intersection de l'élément de contour 3 de la table TABNAME1 et de l'élément de contour 7 de la table TABNAME2. Les coordonnées de l'intersection, dans le plan actif, sont rangées dans la variable ISCOORD (1er élément = abscisse, 2e élément = ordonnée). S'il n'existe pas d'intersection, il y a saut vers PASINTER (aucune intersection trouvée).

Code de programme	Commentaire
DEF REAL TABNAME1[12,11]	; Table de contour 1

Code de programme	**Commentaire**
DEF REAL TABNAME2[10,11]	; Table de contour 2
DEF REAL ISCOORD [2]	; Variable pour les coordonnées du point d'intersection.
DEF BOOL ISPOINT	; Variable pour l'état du point d'intersection.
DEF INT MODE	; Variable pour le type d'usinage.
…	
MODE=1	; Calcul indépendant du plan actif.
N10 ISPOINT=INTERSEC(TABNAME1[3],TABNAME2[7], ISCOORD,MODE)	; Appel de la fonction Intersection des éléments de contour.
N20 IF ISPOINT==FALSE GOTOF PASINTER	; Saut vers PASINTER.
…	

3.25.5 Exécution des éléments de contour d'une table bloc par bloc (EXECTAB)

La fonction EXECTAB permet d'exécuter bloc par bloc les éléments de contour d'une table qui a été créée avec CONTPRON, par exemple.

Syntaxe

```
EXECTAB(<table de contour>[<élément de contour>])
```

Signification

EXECTAB :	Procédure prédéfinie pour exécuter un élément de contour
<Table de contour> :	Nom de la table de contour
<Elément de contour> :	Numéro de l'élément de contour

Exemple

Les éléments de contour 0 à 2 de la table TABC doivent être exécutés bloc par bloc.

Code de programme	**Commentaire**
N10 EXECTAB (TABC[0])	; Exécuter l'élément 0 de la table TABC.
N20 EXECTAB(TABC[1])	; Exécuter l'élément 1 de la table TABC.
N30 EXECTAB(TABC[2])	; Exécuter l'élément 2 de la table TABC.

3.25.6 Calcul de données de cercles (CALCDAT)

La fonction CALCDAT permet de calculer le rayon et les coordonnées du centre d'un cercle à partir de trois ou quatre points connus du cercle. Les points indiqués doivent être distincts.

Lorsque 4 points qui ne se trouvent pas exactement sur un cercle sont indiqués, une moyenne est formée pour le centre du cercle et le rayon.

Remarque

Règle de calcul pour la formation de moyenne

Le calcul de l'arc de cercle est effectué 4 fois :

1. avec les points 1, 2, 3 du cercle
2. avec les points 1, 2, 4 du cercle
3. avec les points 1, 3, 4 du cercle
4. avec les points 2, 3, 4 du cercle

La valeur d'abscisse et la valeur d'ordonnée des coordonnées du centre du cercle sont calculées en additionnant les valeurs d'abscisse ou les valeurs d'ordonnée des quatre calculs d'arc de cercle, puis en divisant le résultat par 4.

Le rayon est calculé en déterminant la racine carrée de la somme des quatre rayons des calculs d'arc de cercle, puis en multipliant le résultat par 0,5.

Syntaxe

```
<Etat>=CALCDAT(<points du
cercle>[<nombre>,<type>],<nombre>,<résultat>)
```

Signification

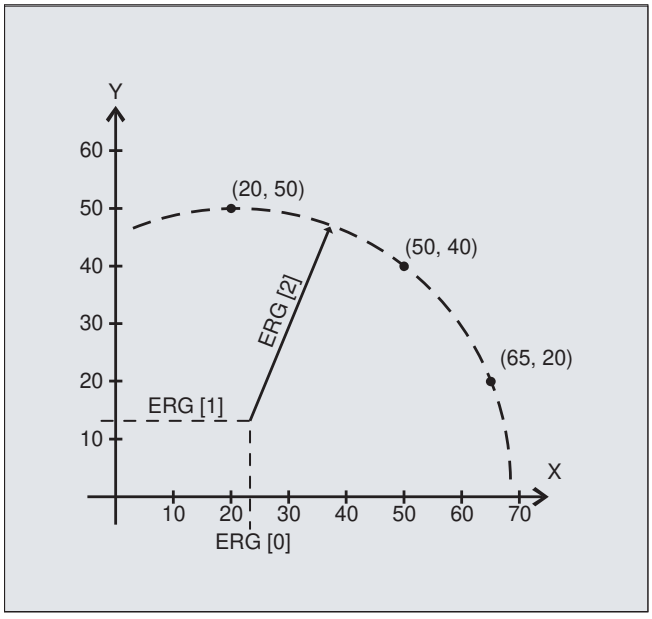
CALCDAT :	Fonction prédéfinie pour calculer le rayon et les coordonnées du centre d'un cercle à partir de 3 ou 4 points		
<Etat> :	Variable de l'état du calcul du cercle		
	Type :	BOOL	
	Valeur :	TRUE	Les points indiqués se situent sur un cercle.
		FALSE	Les points indiqués ne se situent **pas** sur un cercle.
<Points du cercle>[] :	Variable pour l'indication des points du cercle avec les paramètres :		
	<Nombre> :	Nombre de points du cercle (3 ou 4)	
	<Type> :	Type de coordonnées, par exemple 2 pour coordonnées à deux points	
<Nombre> :	Paramètre du nombre de points utilisés pour le calcul (3 ou 4)		
<Résultat>[3] :	Variables du résultat : Coordonnées du centre du cercle et rayon		
	0	Coordonnées du centre du cercle : abscisse	
	1	Coordonnées du centre du cercle : ordonnée	
	2	Rayon	

Remarque

Les variables doivent avoir été définies avant leur utilisation.

Exemple

On veut savoir si trois points sont situés sur un arc de cercle.



Code de programme	Commentaire
N10 DEF REAL PT[3,2]=(20,50,50,40,65,20)	; Variable pour l'indication des points du cercle.
N20 DEF REAL RES[3]	; Variable pour le résultat.
N30 DEF BOOL ETAT	; Variable pour l'état.
N40 ETAT=CALCDAT(POINT,3,RES)	; Appel des données de cercle calculées.
N50 IF ETAT == FALSE GOTOF ERROR	; Saut vers l'erreur.

3.25.7 Désactivation de la préparation du contour (EXECUTE)

La fonction EXECUTE permet d'arrêter la préparation du contour et de retourner au mode d'exécution normal.

Syntaxe

EXECUTE(<ERREUR>)

Signification

EXECUTE :	Procédure prédéfinie pour quitter la préparation du contour	
<ERREUR> :	Variable pour la signalisation d'erreur	
	Type :	INT
	La valeur de la variable indique si le contour a pu être préparé correctement :	
	0	Erreur
	1	Aucune erreur

Exemple

Code de programme

```
...
N30 CONTPRON(…)
N40 G1 X… Z…
...
N100 EXECUTE(…)
...
```

3.26 Programmation externe de cycles

3.26.1 Cycles technologiques

3.26.1.1 Introduction

Contenu

Le présent chapitre contient la description des cycles pour les technologies de tournage, fraisage et rectification.

Structure

La description d'un cycle est structurée comme suit :

- **Syntaxe**
  Nom du cycle et séquence d'appel des paramètres de transfert
- **Paramètres**
  Tableau explicatif des différents paramètres

Description des paramètres

Les données suivantes sont indiquées dans le tableau pour un paramètre : nom, description, plage de valeurs et dépendances par rapport à d'autres paramètres.

La colonne qui fait référence au paramètre du masque est destinée à faciliter la recherche de valeurs programmées sur la commande lors de la décompilation d'appels de cycle générés en externe.

Paramètres "réservés à l'interface"

Dans le tableau des paramètres sont identifiés par "réservé à l'interface". Ils sont sans signification pour le fonctionnement du cycle. Ils sont uniquement nécessaires pour pouvoir décompiler entièrement des appels de cycle. S'ils ne sont pas programmés, le cycle peut tout de même être décompilé. Les champs sont repérés par une couleur particulière et doivent être renseignés dans le masque.

Paramètre "réservé"

Les paramètres décrits comme "réservés" doivent être programmés avec la valeur 0 ou avec un caractère d'omission pour que l'affectation des paramètres d'appel suivants corresponde aux paramètres de cycle internes. Exception : pour les paramètres de type String, valeur "" ou caractère d'omission.

Répétition de cycles dans les modèles de position

Les cycles de perçage et de fraisage peuvent être répétés dans les modèles de position (appels modaux). Dans ce cas, `MCALL` doit être écrit avant le cycle sur la même ligne, p. ex. `MCALL CYCLE83(...)`.

Remarque

Si certains paramètres de transfert (p. ex. `<_VARI>, <_GMODE>, <_DMODE>, <_AMODE>`) sont indirectement programmés en tant que paramètres, le masque de saisie est ouvert lors de la décompilation mais il ne peut pas être enregistré, car il n'existe aucune affectation univoque pour certains champs de sélection.

3.26.1.2 Vue d'ensemble spécifique à la technologie

Le tableau synoptique suivant présente une liste de tous les cycles technologiques programmables en externe disponibles, avec la technologie à laquelle ils sont affectés :

Technologie	Cycle technologique
Perçage	• CYCLE81 - Perçage, centrage (Page 1103) • CYCLE82 - Perçage, lamage (Page 1104) • CYCLE85 - Alésage à l'alésoir (Page 1112) • CYCLE86 - Alésage au tour (Page 1113) • CYCLE83 - Perçage profond 1 (Page 1106) • CYCLE830 - Perçage profond 2 (Page 1141) • CYCLE84 - Taraudage sans porte-taraud compensateur (Page 1109) • CYCLE840 - Taraudage avec porte-taraud compensateur (Page 1151) • CYCLE78 - Fraisage de filetages (Page 1098) • CYCLE802 - Positions quelconques (Page 1137) • HOLES1 - Modèle de positions Rangée (Page 1064) • CYCLE801 - Modèle de positions Réseau ou cadre (Page 1136) • HOLES2 - Modèle de positions Cercle complet ou partiel (Page 1065)
Tournage	• CYCLE951 - Chariotage (Page 1161) • CYCLE930 - Gorge (Page 1156) • CYCLE940 - Dégagement forme E et F / Dégagement filet (Page 1158) • CYCLE99 - Filetage au tour (Page 1122) • CYCLE98 - Concaténation de filetages (Page 1118) • CYCLE92 - Tronçonnage (Page 1114)

Technologie	Cycle technologique
Tournage de contours	• CYCLE62 - Appel de contour (Page 1083) • CYCLE952 - Chariotage / Chariotage reste / Plongée / Plongée reste / Plongée G+D / Plongée G+D reste (Page 1164)
Fraisage	• CYCLE61 - Surfaçage (Page 1081) • POCKET3 - Poche rectangulaire (Page 1067) • POCKET4 - Poche circulaire (Page 1069) • CYCLE76 - Tourillon rectangulaire (Page 1094) • CYCLE77 - Tourillon circulaire (Page 1096) • CYCLE79 - Polyèdre (Page 1101) • SLOT1 - Rainure rectiligne (Page 1072) • SLOT2 - Rainure sur cercle (Page 1074) • CYCLE899 - Rainure borgne (Page 1153) • LONGHOLE - Trou oblong (Page 1077) • CYCLE70 - Fraisage de filetage (Page 1088) • CYCLE60 - Gravure (Page 1079)
Fraisage de contours	• CYCLE62 - Appel de contour (Page 1083) • CYCLE72 - Fraisage en contournage (Page 1090) • CYCLE63 - Fraisage de poche de contour / Matière restante du contour de poche / Fraisage du contour de tourillon / Matière restante du contour de tourillon (Page 1084) • CYCLE64 - Perçage ébauche d'une poche de contour (Page 1087)
Rectification	• CYCLE495 - Profilage (Page 1127) • CYCLE435 - Définir le système de coordonnées du dresseur (Page 1127) • CYCLE4071 - Rectification longitudinale avec pénétration au point d'inversion (Page 1170) • CYCLE4072 - Rectification longitudinale avec pénétration au point d'inversion et signal d'abandon (Page 1172) • CYCLE4073 - Rectification longitudinale avec pénétration continue (Page 1176) • CYCLE4074 - Rectification longitudinale avec pénétration continue et signal d'abandon (Page 1177) • CYCLE4075 - Rectification plane avec pénétration au point d'inversion (Page 1180) • CYCLE4077 - Rectification plane avec pénétration au point d'inversion et signal d'abandon (Page 1183) • CYCLE4078 - Rectification plane avec pénétration continue (Page 1187) • CYCLE4079 - Rectification plane avec pénétration intermittente (Page 1189)

Technologie	Cycle technologique
Divers	• CYCLE782 - Adapter à la charge (Page 1129) • CYCLE800 - Pivotement plan / Pivotement outil / Orientation outil (Page 1132) • CYCLE806 - Tournage par interpolation (Page 1140) • CYCLE832 - Réglages High Speed (Page 1147)
Tous	• GROUP_BEGIN - Début du bloc de programme (Page 1192) • GROUP_END - Fin du bloc de programme (Page 1192) • GROUP_ADDEND - Fin du supplément de mise au point (Page 1193)

3.26.1.3 HOLES1 - Modèle de positions Rangée

Syntaxe

```
HOLES1(<SPCA>, <SPCO>, <STA1>, <FDIS>, <DBH>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	X0	<SPCA>	REAL	Point de référence pour la rangée de trous du 1er axe (abs)		
2	Y0	<SPCO>	REAL	Point de référence pour la rangée de trous du 2ème axe (abs)		
3	α0	<STA1>	REAL	Angle de rotation de base (par rapport au 1er axe)		
4	L0	<FDIS>	REAL	Distance du 1er trou au point de référence		
5	L	<DBH>	REAL	Distance entre les trous		
6	N	<NUM>	INT	Nombre de trous		
7		<_VARI>	INT	Réservé		
8		<_UMODE>	INT	Réservé		
9		<_HIDE>	STRING [200]	Positions masquées • 198 caractères max. • Indication du numéro de position consécutif, par exemple "1,3" (les positions 1 et 3 ne sont pas exécutées)		
10		<_NSP>	INT	Réservé		
11		<_DMODE>	INT	Mode affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)

3.26.1.4 HOLES2 - Modèle de positions Cercle complet ou partiel

Syntaxe

```
HOLES2(<CPA>, <CPO>, <RAD>, <STA1>, <INDA>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification			
1	X0	<CPA>	REAL	Centre du cercle de trous du 1er axe (abs) Point de référence dans le 1er axe			(pour XY) (pour XA, YB, ZC)
2	Y0	<CPO>	REAL	Centre du cercle de trous du 2e axe (abs) Point de référence dans le 2e axe			(pour XY) (pour XA, YB, ZC)
3	R	<RAD>	REAL	Rayon du cercle de trous			(pour XY)
4	α0	<STA1>	REAL	Angle de départ ou 1ère position d'axe rotatif			(pour XY) (pour XA, YB, ZC)
5	α1	<INDA>	REAL	Incrément angulaire (seulement pour cercle partiel)			(pour XY, XA, YB, ZC)
					< 0 =	Sens horaire	
					> 0 =	Sens antihoraire	
6	N	<NUM>	INT	Nombre de positions			

N°	Masque de paramètre	Paramètre interne	Type de données	Signification			
7		<_VARI>	INT	Type d'usinage			
				UNITES :	Réservé		
				DIZAINES :	Type de positionnement		
					0 =	Accostage de la position - linéaire	
					1 =	Accostage de la position - sur trajectoire circulaire	
				CENTAINES :	Réservé		
				MILLIERS :	Modèle de cercle		
					0 =	Mode compatibilité, si INDA = 0 alors cercle complet, si INDA <> 0 alors cercle partiel	
					1 =	Cercle complet	
					2 =	Cercle partiel	
				DIZAINES DE MILLIERS :	Modèle de positions avec axe rotatif		
					0 =	XY (sans axe rotatif)	(pour XY)
					1 =	XA (axe X et axe rotatif autour de X)	(uniquement pour XA)
					2 =	YB (axe Y et axe rotatif autour de Y)	(uniquement pour YB)
					3 =	ZC (axe Z et axe rotatif autour de C)	(uniquement pour ZC)
				MILLIONS + CENTAINES DE MILLIERS :	Offset (pour plusieurs axes rotatifs autour du même axe, quand l'indice est trop élevé, alors le 1er axe)		
					00 =	1. Axe A, B ou C	
					01 =	2. Axe A, B ou C	
					...		
					10 =	20. Axe A, B ou C	
8		<_UMODE>	INT	Réservé			
9		<_HIDE>	STRING [200]	Réservé			
10		<_NSP>	INT	Réservé			
11		<_DMODE>	INT	Mode d'affichage			
				UNITES :	Plan d'usinage G17/G18/G19		
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif	
					1 =	G17 (actif uniquement dans le cycle)	
					2 =	G18 (actif uniquement dans le cycle)	
					3 =	G19 (actif uniquement dans le cycle)	

3.26.1.5 POCKET3 - Poche rectangulaire

Syntaxe

```
POCKET3(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_LENG>, <_WID>, <_CRAD>,
<_PA>, <_PO>, <_STA>, <_MID>, <_FAL>, <_FALD>, <_FFP1>, <_FFD>,
<_CDIR>, <_VARI>, <_MIDA>, <_AP1>, <_AP2>, <_AD>, <_RAD1>, <_DP1>,
<_UMODE>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<_RTP>	REAL	Plan de retrait (abs)		
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)		
3	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Profondeur de poche (abs/rel), voir <_AMODE>		
5	L	<_LENG>	REAL	Longueur de poche (rel, à saisir avec le signe)		
6	W	<_WID>	REAL	Largeur de poche (rel, à saisir avec le signe)		
7	R	<_CRAD>	REAL	Rayon d'angle de la poche		
8	X0	<_PA>	REAL	Point de référence, 1er axe (abs)		
9	YO	<_PO>	REAL	Point de référence, 2e axe (abs)		
10	α0	<_STA>	REAL	Angle de rotation, angle entre axe longitudinal (L) et 1er axe		
11	DZ	<_MID>	REAL	Profondeur de passe maximale		
12	UXY	<_FAL>	REAL	Surépaisseur de finition dans le plan		
13	UZ	<_FALD>	REAL	Surépaisseur de finition au fond		
14	F	<_FFP1>	REAL	Avance dans le plan		
15	FZ	<_FFD>	REAL	Avance de pénétration		
16		<_CDIR>	INT	Sens de fraisage :	0 =	En avalant
					1 =	En opposition
17		<_VARI>	INT	Type d'usinage		
				UNITES :		
					1 =	Ebauchage
					2 =	Finition
					4 =	Finition du bord
					5 =	Chanfreinage
				DIZAINES :		
					0 =	Avant-trou, passe avec G0
					1 =	Perpendiculaire, passe avec G1
					2 =	Hélicoïdal
					3 =	Avec oscillation dans l'axe longitudinal de la poche
				CENTAINES :	Réservé	
18	DXY	<_MIDA>	REAL	Profondeur de passe maximale dans le plan (unité, voir <_AMODE>)		
19	L1	<_AP1>	REAL	Longueur de l'ébauche (rel)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
20	W1	<_AP2>	REAL	Largeur de l'ébauche (rel)		
21	AZ	<_AD>	REAL	Profondeur de l'ébauchage (rel)		
22	ER	<_RAD1>	REAL	Rayon de la trajectoire hélicoïdale de pénétration hélicoïdale		
	EW			Angle de pénétration maximal pour oscillatoire		
23	EP	<_DP1>	REAL	Pas d'hélice pour pénétration hélicoïdale		
24		<_UMODE>	INT	Réservé		
25	FS	<_FS>	REAL	Largeur de chanfrein (rel)		
26	ZFS	<_ZFS>	REAL	Profondeur de pénétration (pointe de l'outil) lors du chanfreinage (abs/rel), voir <_AMODE>		
27		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération / calcul du point de départ uniquement	
					0 =	Mode de compatibilité
					1 =	Usinage normal
				MILLIERS :	Cotation à partir du milieu/coin	
					0 =	Mode de compatibilité
					1 =	Cotation à partir du milieu
					2 =	Cotation à partir du point d'angle, position de poche +LENG/+WID
					3 =	Cotation à partir du point d'angle, position de poche -LENG/+WID
					4 =	Cotation à partir du point d'angle, position de poche +LENG/-WID
					5 =	Cotation à partir du point d'angle, position de poche -LENG/-WID
				DIZAINES DE MILLIERS :	Usinage complet / reprise	
					0 =	Mode de compatibilité (<_AP1>, <_AP2> et <_AD> à traiter comme précédemment)
					1 =	Usinage complet
					2 =	Reprise

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
28		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Mode d'avance : groupe G (G94/G95) pour l'avance de surface et l'avance de profondeur	
					0 =	Mode de compatibilité
					1 =	instruction G comme avant l'appel de cycle. G94/G95 identique pour avance de surface et avance de profondeur
				CENTAINES :	---	Réservé
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée
29		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de poche (Z1)	
					0 =	Absolu (mode de compatibilité)
					1 =	Relatif
				DIZAINES :	Unité pour la profondeur de passe dans le plan (DXY)	
					0 =	mm
					1 =	% du diamètre de l'outil
				CENTAINES :	Profondeur de pénétration lors du chanfreinage (ZFS)	
					0 =	Absolu
					1 =	Relatif

3.26.1.6 POCKET4 - Poche circulaire

Syntaxe

```
POCKET4(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_CDIAM>, <_PA>, <_PO>,
<_MID>, <_FAL>, <_FALD>, <_FFP1>, <_FFD>, <_CDIR>, <_VARI>, <_MIDA>,
<_AP1>, <_AD>, <_RAD1>, <_DP1>, <_UMODE>, <_FS>, <_ZFS>, <_GMODE>,
<_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<_RTP>	REAL	Plan de retrait (abs)		
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)		
3	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Profondeur de poche (abs/rel), voir <_AMODE>		
5	Ø	<_CDIAM>	REAL	Diamètre de poche ou rayon de poche, voir <_DMODE>		
6	X0	<_PA>	REAL	Point de référence 1er axe (abs)		
7	Y0	<_PO>	REAL	Point de référence 2e axe (abs)		
8	DZ	<_MID>	REAL	Profondeur de passe maximale, voir <_VARI> = dans le plan Pas d'hélice maximal, voir <_VARI> = hélicoïdal		
9	UXY	<_FAL>	REAL	Surépaisseur de finition dans le plan		
10	UZ	<_FALD>	REAL	Surépaisseur de finition au fond		
11	F	<_FFP1>	REAL	Avance pour l'usinage de la surface		
12	FZ	<_FFD>	REAL	Avance de pénétration		
13		<_CDIR>	INT	Sens de fraisage	0 =	En avalant
					1 =	En opposition
14		<_VARI>	INT	Type d'usinage		
				UNITES :	Usinage	
					1 =	Ebauchage
					2 =	Finition
					4 =	Finition du bord
					5 =	Chanfreinage
				DIZAINES :	Type de passe (ébauchage et finition)	
					0 =	Avant-trou, passe avec G0 (la poche est ébauchée)
					1 =	Perpendiculaire, passe avec G1
					2 =	Hélicoïdal
				CENTAINES :	Réservé	
				MILLIERS :		
					0 =	Dans le plan
					1 =	Hélicoïdal
15	DXY	<_MIDA>	REAL	Profondeur de passe maximale dans le plan, voir <_AMODE>, 0 = 0,8 x diamètre de l'outil		
16	Ø	<_AP1>	REAL	Diamètre/rayon de l'ébauche (rel)		
17	AZ	<_AD>	REAL	Profondeur de l'ébauchage (rel)		
18	ER	<_RAD1>	REAL	Rayon de la trajectoire hélicoïdale de pénétration hélicoïdale		
19	EP	<_DP1>	REAL	Pas d'hélice pour pénétration sur trajectoire hélicoïdale		
20		<_UMODE>	INT	Réservé		
21	FS	<_FS>	REAL	Largeur de chanfrein (rel)		
22	ZFS	<_ZFS>	REAL	Profondeur de pénétration (pointe de l'outil) lors du chanfreinage (abs/rel), voir <_AMODE>		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
23		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Usinage / calcul du point de départ	
					0 =	Mode de compatibilité
					1 =	Usinage normal
				MILLIERS :	Réservé	
				DIZAINES DE MILLIERS :	Usinage complet / reprise	
					0 =	Mode de compatibilité (<_AP1> et <_AD> à traiter comme précédemment)
					1 =	Usinage complet
					2 =	Reprise
24		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Mode d'avance : groupe G (G94/G95) pour l'avance de surface et l'avance de profondeur	
					0 =	Mode de compatibilité
					1 =	instruction G comme avant l'appel de cycle. G94/G95 identique pour avance de surface et avance de profondeur
				CENTAINES :		
					0 =	Mode de compatibilité (<_CDIAM>/<_AP1> saisi en tant que rayon)
					1 =	<_CDIAM>/<_AP1> saisi en tant que diamètre
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
25		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de poche (Z1)	
					0 =	Absolu (mode de compatibilité)
					1 =	Relatif
				DIZAINES :	Unité pour la largeur de passe (DXY)	
					0 =	mm
					1 =	% du diamètre de l'outil
				CENTAINES :	Profondeur de pénétration lors du chanfreinage (ZFS)	
					0 =	Absolu
					1 =	Relatif

3.26.1.7 SLOT1 - Rainure rectiligne

Syntaxe

```
SLOT1 (<RTP>, <RFP>, <SDIS>, <_DP>, <_DPR>, <NUM>, <LENG>, <WID>,
<_CPA>, <_CPO>, <RAD>, <STA1>, <INDA>, <FFD>, <FFP1>, <_MID>,
<CDIR>, <_FAL>, <VARI>, <_MIDF>, <FFP2>, <SSF>, <_FALD>, <_STA2>,
<_DP1>, <_UMODE>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1	RP	<RTP>	REAL	Plan de retrait (abs)
2	Z0	<RFP>	REAL	Point de référence de l'axe outil (abs)
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)
4	Z1	<_DP>	REAL	Profondeur de rainure (abs)
5		<_DPR>	REAL	Profondeur de rainure (rel), par rapport à Z0 (à saisir sans signe)
6		<NUM>	INT	Nombre de rainures = 1
7	L	<LENG>	REAL	Longueur de la rainure
8	W	<WID>	REAL	Largeur de la rainure
9	X0	<_CPA>	REAL	Point de référence, 1er axe du plan
10	Y0	<_CPO>	REAL	Point de référence, 2e axe du plan
11		<RAD>	REAL	Réservé
12	α	<STA1>	REAL	Angle de rotation
13		<INDA>	REAL	Réservé
14	FZ	<FFD>	REAL	Avance de pénétration
15	F	<FFP1>	REAL	Avance
16	DZ	<_MID>	REAL	Profondeur de passe maximale

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
17		<CDIR>	INT	Sens de fraisage	0 =	En avalant
					1 =	En opposition
18	UXY	<_FAL>	REAL	Surépaisseur de finition Plan ou Bord de rainure		
19		<VARI>	INT	Type d'usinage		
				UNITES :		
					0 =	Réservé
					1 =	Ebauchage
					2 =	Finition
					4 =	Finition du bord (usiner bord uniquement)
					5 =	Chanfreinage
				DIZAINES :	Accostage	
					0 =	Avant-trou, passe avec G0 (la rainure est ébauchée)
					1 =	Perpendiculaire, passe avec G1
					2 =	Hélicoïdal
					3 =	Pendulaire
				CENTAINES :	Réservé	
20	DZF	<_MIDF>	REAL	Réservé		
21	FF	<FFP2>	REAL	Réservé		
22	SF	<SSF>	REAL	Réservé		
23	UZ	<_FALD>	REAL	Surépaisseur de finition au fond		
24	ER	<_STA2>	REAL	Rayon de la trajectoire hélicoïdale de pénétration hélicoïdale		
	EW			Angle de pénétration maximal pour oscillatoire		
25	EP	<_DP1>	REAL	Profondeur de pénétration par tour pour l'hélice		
26		<_UMODE>	INT	Réservé		
27	FS	<_FS>	REAL	Largeur de chanfrein (rel) pour le chanfreinage		
28	ZFS	<_ZFS>	REAL	Profondeur de pénétration (pointe de l'outil) lors du chanfreinage (abs/rel), voir <_AMODE>		
29		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération ou calcul du point de départ uniquement	
					1 =	Usinage normal
				MILLIERS :	Cotation à partir du point de référence, de la position de la rainure	
					0 =	Centre
					1 =	Gauche intérieur +L
					2 =	Droite intérieur -L
					3 =	Bord gauche +L
					4 =	Bord droit -L

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
30		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :	Identificateur de version de logiciel	
					1 =	Extension fonctionnelle SLOT1
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée
31		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur finale Z1 (abs/rel)	
					0 =	Compatibilité
					1 =	Z1 (rel)
					2 =	Z1 (abs)
				DIZAINES :	Réservé	
				CENTAINES :	Profondeur de pénétration lors du chanfreinage ZFS	
					0 =	ZFS (abs)
					1 =	ZFS (rel)

Remarque

Le cycle comporte de nouvelles fonctions par rapport aux précédentes versions du logiciel. Ceci a pour conséquence que certains paramètres ne sont plus affichés dans le masque de saisie (<NUM>, <RAD>, <INDA>). Plusieurs rainures peuvent être programmées sur un modèle de position au moyen de "MCALL" et de l'appel du modèle de position souhaité, p. ex. HOLES2.

3.26.1.8 SLOT2 - Rainure sur cercle

Syntaxe

```
SLOT2(<RTP>, <RFP>, <SDIS>, <_DP>, <_DPR>, <NUM>, <AFSL>, <WID>,
<_CPA>, <_CPO>, <RAD>, <STA1>, <INDA>, <FFD>, <FFP1>, <_MID>,
<CDIR>, <_FAL>, <VARI>, <_MIDF>, <FFP2>, <SSF>, <_FFCP>, <_UMODE>,
<_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<RTP>	REAL	Plan de retrait (abs)		
2	Z0	<RFP>	REAL	Point de référence de l'axe outil (abs)		
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Profondeur de rainure (abs)		
5		<_DPR>	REAL	Profondeur de rainure (rel), par rapport à Z0 (à saisir sans signe)		
6	N	<NUM>	INT	Nombre de rainures		
7	α1	<AFSL>	REAL	Angle au centre de la rainure		
8	W	<WID>	REAL	Largeur de la rainure		
9	X0	<_CPA>	REAL	Point de référence = centre du cercle, 1er axe du plan		
10	Y0	<_CPO>	REAL	Point de référence = centre du cercle, 2e axe du plan		
11	R	<RAD>	REAL	Rayon du cercle		
12	α0	<STA1>	REAL	Angle de départ		
13	α2	<INDA>	REAL	Incrément angulaire		
14	FZ	<FFD>	REAL	Avance de pénétration		
15	F	<FFP1>	REAL	Avance		
16	DZ	<_MID>	REAL	Profondeur de passe maximale		
17		<CDIR>	INT	Sens de fraisage	0 =	En avalant
					1 =	En opposition
18	UXY	<_FAL>	REAL	Surépaisseur de finition dans le plan ou bord de rainure		
19		<VARI>	INT	Type d'usinage		
				UNITES :		
					0 =	Usinage complet
					1 =	Ebauchage
					2 =	Finition
					3 =	Finition du bord
					5 =	Chanfreinage
				DIZAINES :		
					0 =	Positionnement intermédiaire avec droite G0
					1 =	Positionnement intermédiaire sur la trajectoire circulaire
				CENTAINES :	Réservé	
				MILLIERS :		
					0 =	Mode compatibilité, si <INDA> = 0 alors cercle complet, si <INDA> <> 0 alors cercle partiel)
					1 =	Cercle complet
					2 =	Cercle partiel
20	DZF	<_MIDF>	REAL	Réservé		
21		<FFP2>	REAL	Réservé		
22		<SSF>	REAL	Réservé		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
23	FF	<_FFCP>	REAL	Réservé		
24		<_UMODE>	INT	Réservé		
25	FS	<_FS>	REAL	Largeur de chanfrein (rel)		
26	ZFS	<_ZFS>	REAL	Profondeur de pénétration (pointe de l'outil) lors du chanfreinage (abs/rel), voir <_AMODE>		
27		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération ou calcul du point de départ uniquement	
					0 =	Mode de compatibilité
					1 =	Usinage normal
28		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :	Identificateur de version de logiciel	
					1 =	Fonctions SLOT2 à partir de la version 2.5
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée
29		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur finale Z1 (abs/rel)	
					0 =	Compatibilité
					1 =	Z1 (rel)
					2 =	Z1 (abs)
				DIZAINES :	Réservé	
				CENTAINES :	Profondeur de pénétration lors du chanfreinage ZFS	
					0 =	ZFS (abs)
					1 =	ZFS (rel)

3.26.1.9 LONGHOLE - Trou oblong

Syntaxe

```
LONGHOLE(<RTP>, <RFP>, <SDIS>, <_DP>, <_DPR>, <NUM>, <LENG>, <_CPA>,
<_CPO>, <RAD>, <STA1>, <INDA>, <FFD>, <FFP1>, <MID>, <_VARI>,
<_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<RTP>	REAL	Plan de retrait (abs)		
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)		
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Profondeur de trou oblong (abs)		
5		<_DPR>	REAL	Profondeur de trou oblong (rel), par rapport à Z0 (à saisir sans signe)		
6		<NUM>	INT	Nombre de trous oblongs = 1		
7	L	<LENG>	REAL	Longueur du trou oblong		
8	X0	<_CPA>	REAL	Point de référence, 1er axe du plan		
9	Y0	<_CPO>	REAL	Point de référence, 2e axe du plan		
10		<RAD>	REAL	Réservé		
11	α0	<STA1>	REAL	Angle de rotation		
12		<INDA>	REAL	Réservé		
13	FZ	<FFD>	REAL	Avance de pénétration		
14	F	<FFP1>	REAL	Avance		
15	DZ	<MID>	REAL	Profondeur de passe maximale		
16		<_VARI>	INT	Type d'usinage		
				UNITES :	Mode de pénétration	
					1 =	Perpendiculairement avec G1
					3 =	Pendulaire
				CENTAINES :	Réservé	
17		<_UMODE>	INT	Réservé		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
18		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération ou calcul du point de départ uniquement	
					0 =	Mode compatibilité
					1 =	Usinage normal
				MILLIERS :	Cotation à partir du point de référence, de la position de la rainure	
					0 =	Centre
					1 =	Gauche intérieur +L
					2 =	Droite intérieur -L
					3 =	Bord gauche +L
					4 =	Bord droit -L
19		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Mode d'avance : groupe G (G94/G95) pour l'avance de surface et l'avance de profondeur	
					0 =	Mode compatibilité
					1 =	instruction G comme avant l'appel de cycle. G94/G95 identique pour avance de surface et avance de profondeur
				CENTAINES :	Réservé	
				MILLIERS :	Identificateur de version de logiciel	
					1 =	Extension fonctionnelle LONGHOLE (cotation à partir du point de référence)
20		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur finale Z1 (abs/rel)	
					0 =	Compatibilité
					1 =	Z1 (rel)
					2 =	Z1 (abs)

Remarque

Le cycle comporte de nouvelles fonctions par rapport aux précédentes versions du logiciel. Ceci a pour conséquence que certains paramètres ne sont plus affichés dans le masque de saisie (<NUM>, <RAD>, <INDA>). Plusieurs rainures peuvent être programmées sur un modèle de position au moyen de "MCALL" et de l'appel du modèle de position souhaité, p. ex. HOLES2.

3.26.1.10 CYCLE60 - Gravure

Syntaxe

```
CYCLE60(<_TEXT>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_PA>,
<_PO>, <_STA>, <_CP1>, <_CP2>, <_WID>, <_DF>, <_FFD>, <_FFP1>,
<_VARI>, <_CODEP>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1		<_TEXT>	STRING [200]	Texte à graver (100 caractères max.)
2	RP	<_RTP>	REAL	Plan de retrait (abs)
3	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)
4	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au plan de référence, à saisir sans signe)
5	Z1	<_DP>	REAL	Profondeur (abs), voir <_AMODE>
6	Z1	<_DPR>	REAL	Profondeur (rel), voir <_AMODE>
7	X0	<_PA>	REAL	Point de référence, 1er axe du plan (abs) - cartésien, voir <_VARI>
	R			Point de référence, longueur (rayon) - polaire, voir <_VARI>
8	Y0	<_PO>	REAL	Point de référence, 2e axe du plan (abs) - cartésien, voir <_VARI>
	α0			Point de référence, angle par rapport au 1er axe - polaire, voir <_VARI>
9	α1	<_STA>	REAL	Orientation du texte, angle de la ligne de texte par rapport au 1er axe, voir <_VARI>
10	XM	<_CP1>	REAL	Centre du cercle de texte, 1er axe du plan (abs) - cartésien, voir <_VARI>
	LM			Centre du cercle de texte, longueur (rayon) rapportée à WNP - polaire, voir <_VARI>
11	YM	<_CP2>	REAL	Centre du cercle de texte, 2e axe du plan (abs) - cartésien, voir <_VARI>
	αM			Centre du cercle de texte, angle rapporté au 1er axe - polaire, voir <_VARI>
12	W	<_WID>	REAL	Hauteur des caractères (à saisir sans signe)
13	DX1 DX2	<_DF>	REAL	Espacement des caractères / largeur totale, voir <_VARI>
	α2			Angle au centre, voir <_VARI>

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
14	FZ	<_FFD>	REAL	Avance de pénétration, voir <_DMODE>		
15	F	<_FFP1>	REAL	Avance pour l'usinage de la surface		
16		<_VARI>	INT	Usinage (orientation et point de référence du texte à graver)		
				UNITES :	Point de référence	
					0 =	cartésien
					1 =	Polaire
				DIZAINES :	Orientation du texte	
					0 =	texte sur une ligne
					1 =	texte sur un arc de cercle en haut
					2 =	texte sur un arc de cercle en bas
				CENTAINES :	Réservé	
				MILLIERS :	Point de référence horizontal du texte	
					0 =	Gauche
					1 =	au centre
					2 =	Droite
				DIZAINES DE MILLIERS :	Point de référence vertical du texte	
					0 =	en bas
					1 =	au centre
					2 =	en haut
				CENTAINES DE MILLIERS :	Longueur du texte	
					0 =	espacement des caractères
					1 =	largeur totale du texte (uniquement pour texte linéaire)
					2 =	angle d'attaque (disposition du texte en arc de cercle uniquement)
				MILLIONS :	Centre du cercle	
					0 =	Cartésien
					1 =	Polaire
				DIZAINES DE MILLIONS :	Ecriture renversée	
					0 =	Compatibilité
					1 =	Ecriture renversée activée
					2 =	Ecriture renversée désactivée
17		<_CODEP>	INT	Numéro de la page de code pour la police (actuellement uniquement 1252)		
18		<_UMODE>	INT	Réservé		
19		_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération / calcul du point de départ uniquement	
					0 =	Mode de compatibilité
					1 =	Usinage normal

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
20		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Mode d'avance : groupe G (G94/G95) pour l'avance de surface et l'avance de profondeur	
					0 =	Mode de compatibilité
					1 =	instruction G comme avant l'appel de cycle. G94/G95 identique pour avance de surface et avance de profondeur
21		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur finale (<_DP>, <_DPR>)	
					0 =	Compatibilité
					1 =	Relatif (<_DPR>)
					2 =	Absolu (<_DP>)

3.26.1.11 CYCLE61 - Surfaçage

Syntaxe

```
CYCLE61(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_PA>, <_PO>, <_LENG>,
<_WID>, <_MID>, <_MIDA>, <_FALD>, <_FFP1>, <_VARI>, <_LIM>,
<_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1	RP	<_RTP>	REAL	Plan de retrait (abs)
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil, hauteur pièce brute (abs)
3	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)
4	Z1	<_DP>	REAL	Hauteur pièce finie (abs/rel), voir <_AMODE>
5	X0	<_PA>	REAL	Point d'angle 1 sur le 1er axe (abs)
6	Y0	<_PO>	REAL	Point d'angle 1 sur le 2e axe (abs)
7	X1	<_LENG>	REAL	Point d'angle 2 sur le 1er axe (abs/rel), voir <_AMODE>
8	Y1	<_WID>	REAL	Point d'angle 2 sur le 2e axe (abs/rel), voir <_AMODE>

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
9	DZ	<_MID>	REAL	Profondeur de passe maximale		
10	DXY	<_MIDA>	REAL	Profondeur de passe maximale dans le plan (unité, voir <_AMODE>)		
11	UZ	<_FALD>	REAL	Surépaisseur de finition au fond		
12	F	<_FFP1>	REAL	Avance d'usinage		
13		<_VARI>	INT	Type d'usinage		
				UNITES :	Usinage	
					1 =	Ebauchage
					2 =	Finition
				DIZAINES :	Sens d'usinage	
					1 =	Parallèle au 1er axe, un seul sens
					2 =	Parallèle au 2e axe, un seul sens
					3 =	Parallèle au 1er axe, sens alternés
					4 =	Parallèle au 2e axe, sens alternés
14		<_LIM>	INT	Limites		
				UNITES :	Limite 1er axe moins	
					0 =	Non
					1 =	Oui
				DIZAINES :	Limite 1er axe plus	
					0 =	Non
					1 =	Oui
				CENTAINES :	Limite 2e axe moins	
					0 =	Non
					1 =	Oui
				MILLIERS :	Limite 2e axe plus	
					0 =	Non
					1 =	Oui
15		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
16		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur finale (<_DP>)	
					0 =	Absolu
					1 =	Relatif
				DIZAINES :	Unité pour la profondeur de passe dans le plan (<_MIDA>)	
					0 =	mm
					1 =	% du diamètre de l'outil
				CENTAINES :	Réservé	
				MILLIERS :	Longueur de la surface	
					0 =	Relatif
					1 =	Absolu
				DIZAINES DE MILLIERS :	Largeur de la surface	
					0 =	Relatif
					1 =	Absolu

3.26.1.12 CYCLE62 - Appel de contour

Syntaxe

```
CYCLE62(<_KNAME>,<_TYPE>, <_LAB1>, <_LAB2>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	PRG/CON	<_KNAME>	STRING [140]	Le nom du contour ou le nom du sous-programme ne doit pas être programmé lorsque _TYPE = 2		
2		<_TYPE>	INT	Détermination de la saisie du contour		
					0 =	Sous-programme
					1 =	Nom du contour
					2 =	Etiquettes
					3 =	Etiquettes dans le sous-programme
3	LAB1	<_LAB1>	STRING[32]	Etiquette 1, début du contour		
4	LAB2	<_LAB2>	STRING[32]	Etiquette 2, fin du contour		

3.26.1.13 CYCLE63 - Fraisage de poche de contour / Matière restante du contour de poche / Fraisage du contour de tourillon / Matière restante du contour de tourillon

Syntaxe

```
CYCLE63(<_PRG>, <_VARI>, <_RP>, <_Z0>, <_SC>, <_Z1>, <_F>, <_FZ>,
<_DXY>, <_DZ>, <_UXY>, <_UZ>, <_CDIR>, <_XS>, <_YS>, <_ER>, <_EP>,
<_EW>, <_FS>, <_ZFS>, <_TR>, <_DR>, <_UMODE>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	PRG	<_PRG>	STRING [100]	Nom du programme d'évidement		
2		<_VARI>	INT	Type d'usinage		
				UNITES :	Usinage technologique	
					1 =	Ebauchage
					3 =	Finition du fond
					4 =	Finition du bord
					5 =	Chanfreinage
				DIZAINES :	Type de passe	
					0 =	Plongée centrée
					1 =	Plongée hélicoïdale
					2 =	Plongée pendulaire
				CENTAINES :	Réservé	
				MILLIERS :	Mode de relèvement	
					0 =	Relèvement vers le plan de retrait
					1 =	Relèvement vers le point de référence + distance de sécurité
				DIZAINES DE MILLIERS :	Point de départ pour l'ébauchage et la finition du fond	
					0 =	Auto
					1 =	Manuel
3	RP	<_RP>	REAL	Plan de retrait (abs)		
4	Z0	<_Z0>	REAL	Point de référence de l'axe outil (abs)		
5	SC	<_SC>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
6	Z1	<_Z1>	REAL	Profondeur finale, (voir <_AMODE> UNITES)		
7	F	<_F>	REAL	Avance dans le plan ébauchage/finition		
8	FZ	<_FZ>	REAL	Avance de pénétration		
9	DXY	<_DXY>	REAL	Prise de passe dans le plan - unité (voir <_AMODE> DIZAINES)		
10	DZ	<_DZ>	REAL	Pénétration Profondeur		
11	UXY	<_UXY>	REAL	Surépaisseur de finition dans le plan		
12	UZ	<_UZ>	REAL	Surépaisseur de finition au fond		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
13		<_CDIR>	INT	Sens de fraisage	0 =	En avalant
					1 =	En opposition
14	XS	<_XS>	REAL	Point de départ X, absolu		
15	YS	<_YS>	REAL	Point de départ Y, absolu		
16	ER	<_ER>	REAL	Pénétration hélicoïdale : rayon		
17	EP	<_EP>	REAL	Pénétration hélicoïdale : pas		
18	EW	<_EW>	REAL	Pénétration oscillatoire : angle de pénétration maximal		
19	FS	<_FS>	REAL	Largeur de chanfrein (rel) pour le chanfreinage		
20	ZFS	<_ZFS>	REAL	Profondeur de pénétration de la pointe d'outil lors du chanfreinage (voir <_AMODE> CENTAINES)		
21	TR	<_TR>	STRING[32]	Nom de l'outil de référence pour l'enlèvement de matière restante		
22	DR	<_DR>	INT	Outil de référence, numéro D, pour enlèvement de matière restante		
23		<_UMODE>	INT	Réservé		
24		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération / calcul du point de départ uniquement	
					0 =	Usinage normal (aucun mode de compatibilité n'est requis)
					1 =	Usinage normal
					2 =	Réservé

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
25		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Réservé	
				CENTAINES :	Mode technologie	
					1 =	Poche
					2 =	Tourillon
				MILLIERS :	Semi-finition	
					0 =	Non
					1 =	Oui
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée
				CENTAINES DE MILLIERS :	Nom du programme automatique	
					0 =	non
					1 =	Oui
26		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur finale (Z1)	
					0 =	Absolu (mode de compatibilité)
					1 =	Relatif
				DIZAINES :	Unité pour la profondeur de passe dans le plan (DXY)	
					0 =	mm
					1 =	% du diamètre de l'outil
				CENTAINES :	Profondeur de pénétration lors du chanfreinage (ZFS)	
					0 =	Absolu
					1 =	Relatif
				MILLIERS :	---	Réservé

3.26.1.14 CYCLE64 - Perçage ébauche d'une poche de contour

Syntaxe

```
CYCLE64(<_PRG>, <_VARI>, <_RP>, <_Z0>, <_SC>, <_Z1>, <_F>, <_DXY>,
<_UXY>, <_UZ>, <_CDIR>, <_TR>, <_DR>, <_UMODE>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	PRG	<_PRG>	STRING [100]	Nom du programme de perçage/centrage		
2		<_VARI>	INT	Type d'usinage		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :	Mode de relèvement	
					0 =	Relèvement vers le plan de retrait
					1 =	Relèvement vers le point de référence + distance de sécurité
3	RP	<_RP>	REAL	Plan de retrait (abs)		
4	Z0	<_Z0>	REAL	Point de référence (abs)		
5	SC	<_SC>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
6	Z1	<_Z1>	REAL	Profondeur de perçage/centrage, (voir <_AMODE> UNITES)		
7	F	<_F>	REAL	Avance Perçage/Centrage		
8	DXY	<_DXY>	REAL	Prise de passe dans le plan - unité (voir <_AMODE> DIZAINES)		
9	UXY	<_UXY>	REAL	Surépaisseur de finition dans le plan		
10	UZ	<_UZ>	REAL	Surépaisseur de finition au fond		
11		<_CDIR>	INT	Sens de fraisage	0 =	En avalant
					1 =	En opposition
12	TR	<_TR>	STRING[20]	Nom de l'outil de référence		
13	DR	<_DR>	INT	Outil de référence, numéro D		
14		<_UMODE>	INT	Réservé		
15		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération / calcul du point de départ uniquement	
					0 =	Usinage normal (aucun mode de compatibilité n'est requis)
					1 =	Usinage normal
					2 =	Réservé

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
25		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Mode technologie	
					1 =	Perçage ébauche
					2 =	Centrage
				CENTAINES :	---	Réservé
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	---	Réservé
				CENTAINES DE MILLIERS :	Nom du programme automatique	
					0 =	non
					1 =	Oui
26		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de perçage/centrage Z1	
					0 =	Absolu (mode de compatibilité)
					1 =	Relatif
				DIZAINES :	Unité pour la profondeur de passe dans le plan (DXY)	
					0 =	mm
					1 =	% du diamètre de l'outil

3.26.1.15 CYCLE70 - Fraisage de filetage

Syntaxe

```
CYCLE70(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DIATH>, <_H1>, <_FAL>,
<_PIT>, <_NT>, <_MID>, <_FFR>, <_TYPTH>, <_PA>, <_PO>, <_NSP>,
<_VARI>, <_PITA>, <_PITM>, <_PTAB>, <_PTABA>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1	RP	<_RTP>	REAL	Plan de retrait (abs)
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
3	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Longueur de filet (abs/rel), voir <_AMODE> Tenir compte du dégagement au fond du perçage (un demi pas minimum)		
5	Ø	<_DIATH>	REAL	Diamètre nominal du filetage		
6	H1	<_H1>	REAL	Profondeur de filet		
7	U	<_FAL>	REAL	Surépaisseur de finition		
8	P	<_PIT>	REAL	Pas de vis (sélection <_PITA> : mm, inch, MODULE, filets/inch)		
9	NT	<_NT>	INT	Nombre de dents sur la plaque de découpe Longueur d'outil toujours rapportée à la dent inférieure.		
10	DXY	<_MID>	REAL	Pénétration maximale par passe <_MID> > <_H1> : tout en une seule passe		
11	F	<_FFR>	REAL	Avance de fraisage		
12		<_TYPTH>	INT	Type de filetage	0 =	Filetage intérieur
					1 =	Filetage extérieur
13	X0	<_PA>	REAL	Centre du cercle, 1er axe (abs)		
14	Y0	<_PO>	REAL	Centre du cercle, 2e axe (abs)		
15	αS	<_NSP>	REAL	Angle d'attaque (filetage multifilet)		
16		<_VARI>	INT	Type d'usinage		
				UNITES :		
					1 =	Ebauchage
					2 =	Finition
				DIZAINES :		
					1 =	De haut en bas
					2 =	De bas en haut
				CENTAINES :		
					0 =	Filetage à droite
					1 =	Filetage à gauche
17		<_PITA>	INT	Evaluation du pas de vis		
					0 =	Mode de compatibilité
					1 =	Pas en mm
					2 =	Pas en filets par inch (TPI)
					3 =	Pas en inch
					4 =	Pas en tant que MODULE
18		<_PITM>	STRING[15]	Chaîne de caractères servant de mémento pour la saisie du pas de vis (réservé à l'interface)		
19		<_PTAB>	STRING[20]	Chaîne de caractères pour table de filetages ("", "ISO", "BSW", "BSP", "UNC") (réservé à l'interface)		
20		<_PTABA>	STRING[20]	Chaîne de caractères pour la sélection dans la table de filetages (p. ex. "M 10", "M 12"...) (réservé à l'interface)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
21		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Usinage / calcul du point de départ	
					0 =	Mode de compatibilité
					1 =	Usinage normal
22		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
23		<_AMODE>	INT	Mode alternatif		
				UNITES :	Longueur de filet (<_DP>)	
					0 =	Absolu
					1 =	Relatif

3.26.1.16 CYCLE72 - Fraisage en contournage

Syntaxe

```
CYCLE72(<_KNAME>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_MID>, <_FAL>,
<_FALD>, <_FFP1>, <_FFD>, <_VARI>, <_RL>, <_AS1>, <_LP1>, <_FF3>,
<_AS2>, <_LP2>,<_UMODE>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1		<_KNAME>	STRING [141]	Nom du sous-programme de contour
2	RP	<_RTP>	REAL	Plan de retrait (abs)
3	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)
4	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)
5	Z1	<_DP>	REAL	Point final, profondeur finale (abs/rel), voir <_AMODE>
6	DZ	<_MID>	REAL	Profondeur de passe maximale (rel, à saisir sans signe)
7	UXY	<_FAL>	REAL	Surépaisseur de finition dans le plan (rel), surépaisseur sur le contour du bord
8	UZ	<_FALD>	REAL	Surépaisseur de finition au fond (rel), surépaisseur au fond (à saisir sans signe)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
9	FX	<_FFP1>	REAL	Avance sur contour		
10	FZ	<_FFD>	REAL	Avance pour profondeur de passe (ou pénétration spatiale)		
11		<_VARI>	INT	Type d'usinage		
				UNITES :	Usinage	
					1 =	Ebauchage
					2 =	Finition
					5 =	Chanfreinage
				DIZAINES :		
					0 =	Trajets intermédiaires avec G0
					1 =	Trajets intermédiaires avec G1
				CENTAINES :	Retrait en fin de contour	
					0 =	Retrait en fin de contour jusqu'au point de référence
					1 =	Retrait en fin de contour jusqu'au point de référence + <_SDIS>
					2 =	Retrait en fin de contour de <_SDIS>
					3 =	Pas de retrait en fin de contour, le point de départ suivant est accosté avec l'avance sur contour
				MILLIERS :	Réservé	
				DIZAINES DE MILLIERS :	Usiner le contour	
					0 =	Usiner le contour en marche avant
					1 =	Usiner le contour en marche arrière Restrictions pour la marche arrière : • 170 éléments de contour max. (y compris chanfreins ou arrondis) • Seules les valeurs du plan (X/Y) et F sont évaluées
12		<_RL>	INT	Sens d'usinage		
					40 =	Au centre du contour (G40, accostage et retrait : droit ou perpendiculaire)
					41 =	A gauche du contour (G41, accostage et retrait : droit ou circulaire)
					42 =	A droite du contour (G42, accostage et retrait : droit ou circulaire)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
13		<_AS1>	INT	Mouvement d'accostage du contour		
				UNITES :		
					1 =	Droite
					2 =	Quart de cercle
					3 =	Demi-cercle
					4 =	Accostage et retrait perpendiculaires
				DIZAINES :		
					0 =	Dernier mouvement, dans le plan
					1 =	Dernier mouvement, spatial
14	L1	<_LP1>	REAL	Distance ou rayon d'accostage (rel, à saisir sans signe)		
15	FZ	<_FF3>	REAL	Avance pour trajets intermédiaires (G94/G95 comme sur contour)		
16		<_AS2>	INT	Mouvement de retrait du contour (sauf pour accostage/retrait perpendiculaires)		
				UNITES :		
					1 =	Droite
					2 =	Quart de cercle
					3 =	Demi-cercle
				DIZAINES :		
					0 =	Dernier mouvement, dans le plan
					1 =	Dernier mouvement, spatial
17	L2	<_LP2>	REAL	Distance ou rayon de retrait (rel, à saisir sans signe)		
18		<_UMODE>	INT	Réservé		
19	FS	<_FS>	REAL	Largeur de chanfrein (rel)		
20	ZFS	<_ZFS>	REAL	Profondeur de pénétration (pointe de l'outil) lors du chanfreinage (abs/rel), voir <_AMODE>		
21		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération / calcul du point de départ uniquement	
					0 =	Mode de compatibilité
					1 =	Usinage normal

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
22		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Mode d'avance : groupe G (G94/G95) pour l'avance de surface et l'avance de profondeur	
					0 =	Mode de compatibilité
					1 =	instruction G comme avant l'appel de cycle. G94/G95 identique pour avance de surface et avance de profondeur
				MILLIERS :		
					0 =	Mode compatibilité : le nom du contour figure dans <_KNAME>
					1 =	Le nom du contour est programmé dans le CYCLE62 et transféré dans _SC_CONT_NAME
23		<_AMODE>	INT	Mode alternatif		
				UNITES :	Point final Z1 (<_DP>)	
					0 =	Absolu (mode de compatibilité)
					1 =	Relatif
				DIZAINES :	Unité pour la profondeur de passe dans le plan	
					0 =	mm, inch
					1 =	Réservé
				CENTAINES :	Profondeur de pénétration lors du chanfreinage (ZFS) (<_ZFS>)	
					0 =	Absolu
					1 =	Relatif

Remarque

Si les paramètres de transfert ci-après sont programmés indirectement (en tant que paramètres), le masque de saisie n'est pas décompilé :

<_VARI>, <_RL>, <_AS1>, <_AS2>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>

3.26.1.17 CYCLE76 - Tourillon rectangulaire

Syntaxe

```
CYCLE76(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_LENG>, <_WID>,
<_CRAD>, <_PA>, <_PO>, <_STA>, <_MID>, <_FAL>, <_FALD>, <_FFP1>,
<_FFD>, <_CDIR>, <_VARI>, <_AP1>, <_AP2>, <_FS>, <_ZFS>, <_GMODE>,
<_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<_RTP>	REAL	Plan de retrait (abs)		
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)		
3	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Profondeur de tourillon (abs)		
5		<_DPR>	REAL	Profondeur de tourillon (rel), par rapport à Z0 (à saisir sans signe)		
6	L	<_LENG>	REAL	Longueur de tourillon, voir <_GMODE> (à saisir sans signe)		
7	W	<_WID>	REAL	Largeur de tourillon, voir <_GMODE> (à saisir sans signe)		
8	R	<_CRAD>	REAL	Rayon d'angle du tourillon (à saisir sans signe)		
9	X0	<_PA>	REAL	Point de référence du tourillon par rapport au 1er axe du plan (abs)		
10	Y0	<_PO>	REAL	Point de référence du tourillon par rapport au 2e axe du plan (abs)		
11	α0	<_STA>	REAL	Angle de rotation, angle entre axe longitudinal (L) et 1er axe du plan		
12	DZ	<_MID>	REAL	Profondeur de passe maximale (rel, à saisir sans signe)		
13	UXY	<_FAL>	REAL	Surépaisseur de finition dans le plan (rel), surépaisseur sur le contour du bord		
14	UZ	<_FALD>	REAL	Surépaisseur de finition au fond (rel), surépaisseur au fond (à saisir sans signe)		
15	FX	<_FFP1>	REAL	Avance sur contour		
16	FZ	<_FFD>	REAL	Avance de pénétration		
17		<_CDIR>	INT	Sens de fraisage (à saisir sans signe)		
				UNITES :		
					0 =	En avalant
					1 =	En opposition
18		<_VARI>	INT	Usinage		
				UNITES :		
					1 =	Ebauchage
					2 =	Finition
					5 =	Chanfreinage
19	L1	<_AP1>	REAL	Longueur du tourillon brut		
20	W1	<_AP2>	REAL	Largeur du tourillon brut		
21	FS	<_FS>	REAL	Largeur de chanfrein (rel)		
22	ZFS	<_ZFS>	REAL	Profondeur de pénétration (pointe de l'outil) lors du chanfreinage (abs, rel), voir <_AMODE>		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
23		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération ou calcul du point de départ uniquement	
					0 =	Mode de compatibilité
					1 =	Usinage normal
				MILLIERS :	Cotation du tourillon à partir du milieu ou du coin	
					0 =	Mode de compatibilité
					1 =	Cotation à partir du milieu
					2 =	Cotation à partir du point d'angle, tourillon +L +W
					3 =	Cotation à partir du point d'angle, tourillon -L +W
					4 =	Cotation à partir du point d'angle, tourillon +L -W
					5 =	Cotation à partir du point d'angle, tourillon -L -W
				DIZAINES DE MILLIERS :	Usinage complet ou reprise	
					0 =	Mode de compatibilité
					1 =	Usinage complet
					2 =	Reprise
24		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	---	Réservé
				CENTAINES :	---	Réservé
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
25		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur finale Z1 (DP)	
					0 =	Compatibilité
					1 =	Relatif
					2 =	Absolu
				DIZAINES :	Réservé	
				CENTAINES :	Profondeur de pénétration lors du chanfreinage (ZFS)	
					0 =	Absolu
					1 =	Relatif

3.26.1.18 CYCLE77 - Tourillon circulaire

Syntaxe

```
CYCLE77(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_CDIAM>, <_PA>,
<_PO>, <_MID>, <_FAL>, <_FALD>, <_FFP1>, <_FFD>, <_CDIR>, <_VARI>,
<_AP1>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<_RTP>	REAL	Plan de retrait (abs)		
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)		
3	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Profondeur de tourillon (abs)		
5		<_DPR>	REAL	Profondeur de tourillon (rel), par rapport à Z0 (à saisir sans signe)		
6	Ø	<_CDIAM>	REAL	Diamètre du tourillon (à saisir sans signe)		
7	X0	<_PA>	REAL	Point de référence du tourillon par rapport au 1er axe du plan (abs)		
8	Y0	<_PO>	REAL	Point de référence du tourillon par rapport au 2e axe du plan (abs)		
9	DZ	<_MID>	REAL	Profondeur de passe maximale (rel, à saisir sans signe)		
10	UXY	<_FAL>	REAL	Surépaisseur de finition dans le plan (rel), surépaisseur sur le contour du bord		
11	UZ	<_FALD>	REAL	Surépaisseur de finition au fond (rel), surépaisseur au fond (à saisir sans signe)		
12	FX	<_FFP1>	REAL	Avance sur contour		
13	FZ	<_FFD>	REAL	Avance de pénétration		
14		<_CDIR>	INT	Sens de fraisage (à saisir sans signe)		
				UNITES :		
					0 =	En avalant
					1 =	En opposition

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
15		<_VARI>	INT	Type d'usinage		
				UNITES :	Usinage	
					1 =	Ebauchage jusqu'à surépaisseur de finition
					2 =	Finition (surépaisseur X/Y/Z = 0)
					5 =	Chanfreinage
16	Ø1	<_AP1>	REAL	Diamètre du tourillon brut		
17	FS	<_FS>	REAL	Largeur de chanfrein (rel)		
18	ZFS	<_ZFS>	REAL	Profondeur de pénétration (pointe de l'outil) lors du chanfreinage (abs/rel), voir <_AMODE>		
19		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération / calcul du point de départ uniquement	
					0 =	Mode de compatibilité
					1 =	Usinage normal
				MILLIERS :	Réservé	
				DIZAINES DE MILLIERS :	Usinage complet / reprise	
					0 =	Mode de compatibilité (<_AP1> à traiter comme précédemment)
					1 =	Usinage complet
					2 =	Reprise
20		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	---	Réservé
				CENTAINES :	---	Réservé
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
21		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur finale Z1 (DP)	
					0 =	Absolu (mode de compatibilité)
					1 =	Relatif
					2 =	Absolu
				DIZAINES :	Réservé	
				CENTAINES :	Profondeur de pénétration lors du chanfreinage (ZFS)	
					0 =	Absolu
					1 =	Relatif

3.26.1.19 CYCLE78 - Fraisage de filetages

Syntaxe

```
CYCLE78(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_ADPR>, <_FDPR>, <_LDPR>,
<_DIAM>, <_PIT>, <_PITA>, <_DAM>, <_MDEP>, <_VARI>, <_CDIR>, <_GE>,
<_FFD>, <_FRDP>, <_FFR>, <_FFP2>, <_FFA>, <_PITM>, <_PTAB>,
<_PTABA>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<_RTP>	REAL	Plan de retrait (abs)		
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)		
3	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Profondeur de perçage finale (abs/rel), voir <_AMODE>		
5		<_ADPR>	REAL	Profondeur de pointage avec avance de perçage réduite (rel), avec <_VARI> DIZAINES DE MILLIERS actif		
6	D	<_FDPR>	REAL	Profondeur de passe maximale (rel) D ≥ Z1 ⇒ une passe à la profondeur de perçage finale D < Z1 ⇒ cycle de perçage de trous profonds avec plusieurs passes et débourrage		
7	ZR	<_LDPR>	REAL	Profondeur de perçage restante lors du perçage débouchant (rel), avec avance FR		
8	Ø	<_DIAM>	REAL	Diamètre nominal du filetage		
9	P	<_PIT>	REAL	Pas de vis en tant que valeur numérique		
10		<_PITA>	INT	Unité du pas de vis P		
					1 =	Pas en mm/tr
					2 =	Pas en filets/inch
					3 =	Pas en inch/tr
					4 =	Pas en tant que MODULE

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
11	DF	<_DAM>	REAL	Valeur/pourcentage pour chaque passe suivante (dégression), voir <_AMODE>		
12	V1	<_MDEP>	REAL	Profondeur de passe minimale (rel), ne prend effet que pour la dégression		
13		<_VARI>	INT	Type d'usinage		
				UNITES :	Réservé	
				DIZAINES :	Débourrage avant fraisage de filetage	
					0 =	Aucun débourrage avant le fraisage de filetages (ne prend effet qu'à la profondeur de perçage finale)
					1 =	Débourrage avant le fraisage de filetages (ne prend effet qu'à la profondeur de perçage finale)
				CENTAINES :	Filetage à droite / à gauche	
					0 =	Filetage à droite
					1 =	Filetage à gauche
				MILLIERS :	Profondeur de perçage restante avec avance de perçage	
					0 =	Aucune profondeur de perçage restante avec avance de perçage FR
					1 =	Profondeur de perçage restante avec avance de perçage FR
				DIZAINES DE MILLIERS :	Pointage avec avance réduite	
					0 =	Aucun pointage avec avance réduite
					1 =	Pointage avec avance réduite Avance de pointage = 0,3 F1, lorsque F1 < 0,15 mm/tour Avance de pointage = 0,1 mm/tour, lorsque F1 ≥ 0,15 mm/tour
14		<_CDIR>	INT	Sens de fraisage	0 =	En avalant
					1 =	En opposition
					4 =	En opposition + en avalant (combinaison ébauchage + finition)
15	Z2	<_GE>	REAL	Valeur de retrait avant fraisage de filetages (rel)		
16	F1	<_FFD>	REAL	Avance de perçage (mm/min ou in/min ou mm/tour)		
17	FR	<_FRDP>	REAL	Avance de perçage pour profondeur de perçage restante (mm/min ou mm/tour)		
18	F2	<_FFR>	REAL	Avance pour fraisage de filetages (mm/min ou mm/dent)		
19	FS	<_FFP2>	REAL	Avance de finition pour <_CDIR> = 4 (mm/min ou mm/dent)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
20		<_FFA>	INT	Pondération des avances		
				UNITES :	Avance de perçage F1	
				DIZAINES :	Avance de perçage pour profondeur de perçage restante FR	
				CENTAINES :	Avance pour fraisage de filetages F2	
				MILLIERS :	Avance de finition FS	
21		<_PITM>	STRING[15]	Chaîne de caractères servant de mémento pour la saisie du pas de vis (réservé à l'interface)[1)]		
22		<_PTAB>	STRING[20]	Chaîne de caractères pour table de filetages ("", "ISO", "BSW", "BSP", "UNC") (réservé à l'interface)[1)]		
23		<_PTABA>	STRING[20]	Chaîne de caractères pour la sélection dans la table de filetages (p. ex. "M 10", "M 12"...) (réservé à l'interface)[1)]		
24		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées), réservé		
25		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
26		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de perçage = profondeur de perçage finale Z1 abs/rel	
					0 =	Absolu
					1 =	Relatif
				DIZAINES :	Valeur/pourcentage DF pour chaque passe suivante (dégression)	
					0 =	Valeur absolue
					1 =	Pourcentage (0,001 à 100 %)

Remarque

[1)] Les paramètres 21, 22 et 23 sont utilisés uniquement lors de la sélection du filetage dans les tables de filetages du masque de saisie. L'accès aux tables de filetages par la définition de cycle n'est pas possible pendant l'exécution du cycle.

3.26.1.20 CYCLE79 - Polyèdre

Syntaxe

```
CYCLE79(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_NUM>, <_SWL>, <_PA>,
<_PO>, <_STA>, <_RC>, <_AP1>, <_MIDA>, <_MID>, <_FAL>, <_FALD>,
<_FFP1>, <_CDIR>, <_VARI>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<_RTP>	REAL	Plan de retrait (abs)		
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)		
3	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Profondeur du polyèdre (abs/rel), voir <_AMODE>		
5	N	<_NUM>	INT	Nombre d'arêtes (1...n)		
6	SW/L	<_SWL>	REAL	Cote sur plats ou longueur d'arête (en fonction de <_VARI>) ("SW" pour cote sur plats, "L" pour longueur d'arête) Cote sur plats uniquement dans le cas d'un nombre pair d'arêtes ou d'une arête unique		
7	X0	<_PA>	REAL	Point de référence du tourillon, 1er axe (abs)		
8	Y0	<_PO>	REAL	Point de référence du tourillon, 2e axe (abs)		
9	α0	<_STA>	REAL	Angle de rotation centre d'arête par rapport au 1er axe (axe X)		
10	R1/FS1	<_RC>	REAL	Arrondi lorsque <_NUM> > 2 (rayon/chanfrein, voir <_AMODE>) (rel, à saisir sans signe) ("R1" pour rayon, "FS1" pour chanfrein)		
11	∅	<_AP1>	REAL	Diamètre brut du tourillon		
12	DXY	<_MIDA>	REAL	Largeur de passe maximale (unité, voir <_AMODE>)		
13	DZ	<_MID>	REAL	Profondeur de passe maximale		
14	UXY	<_FAL>	REAL	Surépaisseur de finition dans le plan		
15	UZ	<_FALD>	REAL	Surépaisseur de finition au fond		
16	F	<_FFP1>	REAL	Avance d'usinage		
17		<_CDIR>	INT	Sens de fraisage	0 =	En avalant
					1 =	En opposition
18		<_VARI>	INT	Type d'usinage		
				UNITES :	Usinage	
					1 =	Ebauchage
					2 =	Finition
					3 =	Finition du bord
					5 =	Chanfreinage
				DIZAINES :	Cote sur plats ou longueur d'arête	
					0 =	Cote sur plats
					1 =	Longueur d'arête

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
19	FS	<_FS>	REAL	Largeur de chanfrein (rel)		
20	ZFS	<_ZFS>	REAL	Profondeur de pénétration (pointe de l'outil) lors du chanfreinage (abs/rel), voir <_AMODE>		
21		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération ou calcul du point de départ uniquement	
					1 =	Usinage normal
22		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	---	Réservé
				CENTAINES :	---	Réservé
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée
23		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur finale (<_DP>)	
					0 =	Absolu
					1 =	Relatif
				DIZAINES :	Unité pour la profondeur de passe dans le plan (<_MIDA>)	
					0 =	mm
					1 =	% du diamètre de l'outil
				CENTAINES :	Profondeur de pénétration lors du chanfreinage (ZFS) (<_ZFS>)	
					0 =	Absolu
					1 =	Relatif
				MILLIERS :	Arrondi (<_RC>)	
					0 =	Rayon
					1 =	Chanfrein

3.26.1.21 CYCLE81 - Perçage, centrage

Syntaxe

```
CYCLE81(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<RTP>	REAL	Plan de retrait (abs)		
2	Z0	<RFP>	REAL	Point de référence (abs)		
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1/∅	<DP>	REAL	Profondeur de perçage (abs) / diamètre de centrage (abs), voir <_GMODE>		
5	Z1	<DPR>	REAL	Profondeur de perçage (rel)		
6	DT	<DTB>	REAL	Arrêt temporisé à la profondeur de perçage finale, voir <_AMODE>		
7		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Centrage par rapport à la profondeur / au diamètre	
					0 =	Compatibilité, profondeur
					1 =	Diamètre
8		<_DMODE>	INT	Mode affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
9		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de perçage Z1 (abs/rel)	
					0 =	Compatibilité issue de la programmation DP/DPR
					1 =	Relatif
					2 =	Absolu
				DIZAINES :	Arrêt temporisé à la profondeur de perçage finale DT en secondes/tours	
					0 =	Compatibilité à partir du signe de DTB (> 0 seconde ou < 0 tour)
					1 =	En secondes
					2 =	En tours

3.26.1.22 CYCLE82 - Perçage, lamage

Syntaxe

```
CYCLE82(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>, <_VARI>, <S_ZA>, <S_FA>, <S_ZD>, <S_FD>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<RTP>	REAL	Plan de retrait (abs)		
2	Z0	<RFP>	REAL	Point de référence (abs)		
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<DP>	REAL	Profondeur de perçage (abs), voir <_AMODE>		
5	Z1	<DPR>	REAL	Profondeur de perçage (rel), voir <_AMODE>		
6	DT	<DTB>	REAL	Arrêt temporisé à la profondeur de perçage finale, voir <_AMODE>		
7		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Profondeur de perçage par rapport à pointe/corps	
					0 =	Compatibilité, pointe
					1 =	Corps
8		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :	Réservé	
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Entrée : Complète
					1 =	Entrée : Simplifiée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
9		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de perçage Z1 (abs/rel)	
					0 =	Compatibilité issue de la programmation DP/DPR
					1 =	Relatif
					2 =	Absolu
				DIZAINES :	Arrêt temporisé DT à la profondeur de perçage finale en secondes/tours	
					0 =	Compatibilité à partir du signe de DT (> 0 seconde / < 0 tour)
					1 =	En secondes
					2 =	En tours
				CENTAINES :	Profondeur de pointage ZA abs/rel	
					0 =	Relatif
					1 =	Absolu
				MILLIERS :	Evaluation avance de pointage	
					0 =	En % de l'avance de perçage
					1 =	F/min
					2 =	F/tr
				DIZAINES DE MILLIERS :	Profondeur de perçage restante ZD abs/rel	
					0 =	Relatif
					1 =	Absolu
				CENTAINES DE MILLIERS :	Evaluation avance de perçage restante	
					0 =	En % de l'avance de perçage
					1 =	F/min
					2 =	F/tr
10		<_VARI>	INT	Type d'usinage pointage/perçage débouchant		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :	Perçage débouchant	
					0 =	Perçage débouchant "Non"
					1 =	Perçage débouchant "Oui"
				DIZAINES DE MILLIERS :	Pointage	
					0 =	Pointage "Non"
					1 =	Pointage "Oui"
11	ZA	<S_ZA>	REAL	Profondeur de pointage relative par rapport au point de référence ou absolue (voir <_AMODE> CENTAINES)		
12	FA	<S_FA>	REAL	Avance de pointage comme valeur ou en % (en association avec <_AMODE> MILLIERS)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
13	ZD	<S_ZD>	REAL	Profondeur de perçage restante relative par rapport à la profondeur de perçage finale ou absolue (en association avec <_AMODE> DIZAINES DE MILLIERS)
14	FD	<S_FD>	REAL	Avance de perçage restante comme valeur ou en % (en association avec <_AMODE> CENTAINES DE MILLIERS)

3.26.1.23 CYCLE83 - Perçage profond 1

Syntaxe

```
CYCLE83(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <FDEP>, <FDPR>, <_DAM>,
<DTB>, <DTS>, <FRF>, <VARI>, <_AXN>, <_MDEP>, <_VRT>, <_DTD>,
<_DIS1>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<RTP>	REAL	Plan de retrait (abs)		
2	Z0	<RFP>	REAL	Point de référence (abs)		
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<DP>	REAL	Profondeur de perçage finale (abs), voir <_AMODE>		
5	Z1	<DPR>	REAL	Profondeur de perçage finale (rel), voir <_AMODE>		
6	D	<FDEP>	REAL	1ère profondeur de perçage (abs), voir <_AMODE>		
7	D	<FDPR>	REAL	1ère profondeur de perçage (rel), voir <_AMODE>		
8	DF	<_DAM>	REAL	Valeur/pourcentage de dégression pour chaque passe suivante, voir <_AMODE>		
9	DTB	<DTB>	REAL	Arrêt temporisé à la profondeur de perçage, voir <_AMODE>		
10	DTS	<DTS>	REAL	Arrêt temporisé au point de départ (pour le débourrage uniquement), voir <_AMODE>		
11	FD1	<FRF>	REAL	Valeur en pourcentage de l'avance pour la première passe, voir <_AMODE>		
12		<VARI>	INT	Type d'usinage		
				UNITES :	Bris de copeaux / débourrage	
					0 =	Bris de copeaux
					1 =	Débourrage
13		<_AXN>	INT	Axe d'outil		
					0 =	3ème axe géométrique
					1 =	1er axe géométrique
					2 =	2ème axe géométrique
					> 2	3ème axe géométrique
14	V1	<_MDEP>	REAL	Profondeur de passe minimale (uniquement avec pourcentage pour dégression)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
15	V2	<_VRT>	REAL	Valeur de retrait après chaque usinage (uniquement pour bris de copeaux)		
					> 0	Valeur de retrait variable
					0 =	Valeur par défaut 1 mm
16	DT	<_DTD>	REAL	Arrêt temporisé à la profondeur de perçage finale, voir <_AMODE>		
17	V3	<_DIS1>	REAL	Précote d'arrêt (uniquement pour débourrage), voir <_AMODE>		
18		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Profondeur de perçage par rapport à pointe/corps	
					0 =	Pointe
					1 =	Corps
19		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	---	Réservé
				CENTAINES :	---	Réservé
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
20		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de perçage = profondeur de perçage finale Z1 (abs/rel)	
					0 =	Compatibilité issue de la programmation <DP>/<DPR>
					1 =	Relatif
					2 =	Absolu
				DIZAINES :	Arrêt temporisé à la profondeur de perçage DTB en secondes/tours	
					0 =	Compatibilité à partir du signe de DTB (> 0 seconde ou < 0 tour)
					1 =	En secondes
					2 =	En tours
				CENTAINES :	Arrêt temporisé au point de départ de DTS en secondes/tours	
					0 =	Compatibilité à partir du signe de DTS (> 0 seconde ou < 0 tour)
					1 =	En secondes
					2 =	En tours
				MILLIERS :	Arrêt temporisé à la profondeur de perçage finale DTD en secondes/tours	
					0 =	Compatibilité à partir du signe de DTD (> 0 seconde ou < 0 tour)
					1 =	En secondes
					2 =	En tours
				DIZAINES DE MILLIERS :	1ère profondeur de perçage D (abs/rel)	
					0 =	Compatibilité issue de la programmation<FDEPF>/<DPR>
					1 =	Relatif
					2 =	Absolu
				CENTAINES DE MILLIERS :	Valeur/pourcentage de dégression <_DAM> pour chaque passe suivante	
					0 =	Compatibilité à partir du signe de <_DAM> (> 0 valeur de dégression ou < 0 facteur de 0,001 à 1,0)
					1 =	Valeur de la dégression
					2 =	Pourcentage (0,001 à 100 %)
				MILLIONS :	Précote d'arrêt V3 automatique/manuel	
					0 =	Compatibilité à partir du signe de <_DIS1> (= 0 automatique ou > 0 manuel)
					1 =	Automatique (calculée dans le cycle)
					2 =	Manuel (valeur programmée)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
				DIZAINES DE MILLIONS :	Facteur d'avance pour la première passe <FRF> en tant que facteur/pourcentage	
					0 =	Compatibilité en tant que facteur (0,001 à 1,0, FRF = 0 signifie 100 %)
					1 =	Pourcentage (0,001 à 999,999 %)

3.26.1.24 CYCLE84 - Taraudage sans porte-taraud compensateur

Syntaxe

```
CYCLE84(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDAC>, <MPIT>,
<PIT>, <POSS>, <SST>, <SST1>, <_AXN>, <_PITA>, <_TECHNO>, <_VARI>,
<_DAM>, <_VRT>, <_PITM>, <_PTAB>, <_PTABA>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<RTP>	REAL	Plan de retrait (abs)		
2	Z0	<RFP>	REAL	Point de référence (abs)		
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<DP>	REAL	Profondeur de perçage = profondeur de perçage finale (abs), voir <_AMODE>		
5	Z1	<DPR>	REAL	Profondeur de perçage = profondeur de perçage finale (rel), voir <_AMODE>		
6	DT	<DTB>	REAL	Arrêt temporisé à la profondeur de perçage en secondes		
7	SDE	<SDAC>	INT	Sens de rotation après fin du cycle		
8		<MPIT>	REAL	Taille de filetage uniquement pour "ISO métrique" (le pas est calculé en interne pendant l'exécution)		
9	P	<PIT>	REAL	Pas de vis en tant que valeur, unité de mesure voir <_PITA>		
10	αS[1)]	<POSS>	REAL	Position de broche pour un arrêt orienté de la broche		
11	S	<SST>	REAL	Vitesse de rotation de la broche pour taraudage		
12	SR	<SST1>	REAL	Vitesse de rotation de la broche pour le retrait		
13		<_AXN>	INT	Axe de perçage	0 =	3. axe géométrique
					1 =	1. axe géométrique
					2 =	2. axe géométrique
					≥ 3 =	3. axe géométrique

N°	Masque de paramètre	Paramètre interne	Type de données	Signification			
14		<_PITA>	INT	Unité de mesure du pas de vis (évaluation <PIT> et <MPIT>)			
					0 =	Pas en mm	- Evaluation <MPIT>/<PIT>
					1 =	Pas en mm	- Evaluation <PIT>
					2 =	Pas en TPI	- Evaluation <PIT> (nombre de filets par inch)
					3 =	Pas en inch	- Evaluation <PIT>
					4 =	MODULE	- Evaluation <PIT>
15		<_TECHNO>	INT	Technologie[1)]			
				UNITES :	Type d'arrêt précis		
					0 =	Type d'arrêt précis comme avant appel de cycle actif	
					1 =	Arrêt précis G601	
					2 =	Arrêt précis G602	
					3 =	Arrêt précis G603	
				DIZAINES :	Commande anticipatrice		
					0 =	Avec/sans commande anticipatrice actif comme avant l'appel de cycle	
					1 =	Avec commande anticipatrice FFWON	
					2 =	Sans commande anticipatrice FFWOF	
				CENTAINES :	Accélération		
					0 =	SOFT/BRISK/DRIVE actif comme avant l'appel de cycle	
					1 =	Avec limitation des à-coups SOFT	
					2 =	Sans limitation des à-coups BRISK	
					3 =	Accélération réduite DRIVE	
				MILLIERS :	Mode broche MCALL		
					0 =	Réactivation de la broche avec MCALL	
					1 =	Maintien de l'asservissement de position avec MCALL	
16		<_VARI>	INT	Type d'usinage			
				UNITES :	0 =	1 passe	
					1 =	Bris de copeaux (taraudage profond)	
					2 =	Débourrage (taraudage profond)	
				MILLIERS :	Mode ISO/SIEMENS non significatif pour masque de saisie		
					0 =	Appel à partir de compatibilité ISO	
					1 =	Appel à partir du contexte SIEMENS	
17	D	<_DAM>	REAL	Profondeur de passe maximale (uniquement pour débourrage / bris de copeaux)			
18	V2	<_VRT>	REAL	Valeur de retrait après chaque usinage (uniquement pour bris de copeaux), voir <_AMODE>			

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
19		<_PITM>	STRING[15]	Chaîne de caractères servant de mémento pour la saisie du pas de vis[2]		
20		<_PTAB>	STRING[5]	Chaîne de caractères pour table de filetages ("", "ISO", "BSW", "BSP", "UNC")[2)]		
21		<_PTABA>	STRING[20]	Chaîne de caractères pour la sélection dans la table de filetages (p. ex. "M 10", "M 12"...)[2)]		
22		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
23		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :	Mode de compatibilité (uniquement pour décompilation du masque de saisie), lorsque PM 52216 bit 0 = 1[1)]	
					0 =	Les paramètres technologiques sont affichés (compatibilité) : les paramètres TECHNO prennent effet
					1 =	Les paramètres technologiques ne sont pas affichés : la technologie "programmée avant l'appel de cycle" reste effective
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
24		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de perçage = profondeur de perçage finale Z1 (abs/rel)	
					0 =	Compatibilité issue de la programmation <DP>/<DPR>
					1 =	Relatif
					2 =	Absolu
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :	Sens de rotation du filet droite/gauche	
					0 =	Compatibilité à partir du signe PIT/MPIT
					1 =	Droite
					2 =	Gauche
				DIZAINES DE MILLIERS :	Réservé	
				CENTAINES DE MILLIERS :	Réservé	
				MILLIONS :	Valeur de retrait, en fonction de chaque usinage V2 manuel/automatique	
					0 =	Compatibilité issue de la programmation <_VRT> (> 0 valeur variable ou ≤ 0 valeur par défaut 1 mm/0,0394 inch)
					1 =	Automatique (valeur par défaut 1 mm/0,0394 inch)
					2 =	Manuel (comme programmé sous V2)

[1] Les champs Technologie peuvent être masqués, en fonction de la donnée de réglage SD52216 $MCS_FUNCTION_MASK_DRILL

[2] Les paramètres 19, 20 et 21 sont utilisés uniquement lors de la sélection du filetage dans les tables de filetages du masque de saisie. L'accès aux tables de filetages par la définition de cycle n'est pas possible pendant l'exécution du cycle.

3.26.1.25 CYCLE85 - Alésage à l'alésoir

Syntaxe

```
CYCLE85(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <FFR>, <RFF>,
<_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1	RP	<RTP>	REAL	Plan de retrait (abs)
2	Z0	<RFP>	REAL	Point de référence (abs)
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
4	Z1	<DP>	REAL	Profondeur de perçage (abs), voir <_AMODE>		
5	Z1	<DPR>	REAL	Profondeur de perçage (rel), voir <_AMODE>		
6	DT	<DTB>	REAL	Arrêt temporisé à la profondeur de perçage finale, voir <_AMODE>		
7	F	<FFR>	REAL	Avance		
8	FR	<RFF>	REAL	Avance pour le retrait		
9		<_GMODE>	INT	Réservé		
10		<_DMODE>	INT	Mode affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
11		<_AMODE>	INT	Mode alternatif (perçage)		
				UNITES :	Profondeur de perçage Z1 (abs/rel)	
					0 =	Compatibilité issue de la programmation DP/DPR
					1 =	Relatif
					2 =	Absolu
				DIZAINES :	Arrêt temporisé DT à la profondeur de perçage finale en secondes/tours	
					0 =	Compatibilité à partir du signe de DT (> 0 seconde ou < 0 tour)
					1 =	En secondes
					2 =	En tours

3.26.1.26 CYCLE86 - Alésage au tour

Syntaxe

```
CYCLE86(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDIR>, <RPA>,
<RPO>, <RPAP>, <POSS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1	RP	<RTP>	REAL	Plan de retrait (abs)
2	Z0	<RFP>	REAL	Point de référence (abs)
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
4	Z1	<DP>	REAL	Profondeur de perçage (abs), voir <_AMODE>		
5	Z1	<DPR>	REAL	Profondeur de perçage (rel), voir <_AMODE>		
6	DT	<DTB>	REAL	Arrêt temporisé à la profondeur de perçage finale, voir <_AMODE>		
7	DIR	<SDIR>	INT	Sens de rotation de la broche	3 =	M3
					4 =	M4
8	DX	<RPA>	REAL	Valeur de relèvement dans la direction X		
9	DY	<RPO>	REAL	Valeur de relèvement dans la direction Y		
10	DZ	<RPAP>	REAL	Valeur de relèvement dans le sens Z		
11	SPOS	<POSS>	REAL	Position de la broche pour le relèvement (pour arrêt orienté de la broche, en degrés)		
12		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Mode de relèvement	
					0 =	Relèvement, compatibilité
					1 =	Pas de relèvement
13		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
14		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de perçage Z1 (abs/rel)	
					0 =	Compatibilité issue de la programmation<DP>/<DPR>
					1 =	Relatif
					2 =	Absolu
				DIZAINES :	Arrêt temporisé à la profondeur de perçage finale DT en secondes/tours	
					0 =	Compatibilité à partir du signe de DT (> 0 seconde ou < 0 tour)
					1 =	En secondes
					2 =	En tours

3.26.1.27 CYCLE92 - Tronçonnage

Syntaxe

```
CYCLE92(<_SPD>, <_SPL>, <_DIAG1>, <_DIAG2>, <_RC>, <_SDIS>, <_SV1>,
<_SV2>, <_SDAC>, <_FF1>, <_FF2>, <_SS2>, <_DIAGM>, <_VARI>, <_DN>,
<_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	X0	<_SPD>	REAL	Point de référence (abs, toujours diamètre)		
2	Y0	<_SPL>	REAL	Point de référence (abs)		
3	X1	<_DIAG1>	REAL	Profondeur pour réduction de vitesse de rotation, voir <_AMODE> (UNITES)		
4	X2	<_DIAG2>	REAL	Profondeur finale, voir <_AMODE> (DIZAINES)		
5	R/FS	<_RC>	REAL	Rayon d'arrondi ou largeur de chanfrein, voir <_AMODE> (MILLIERS)		
6	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
7	S	<_SV1>	REAL	Vitesse de rotation de broche constante, voir <_AMODE> (DIZAINES DE MILLIERS)		
	V			Vitesse de coupe constante		
8	SV	<_SV2>	REAL	Vitesse de rotation maximale pour une vitesse de coupe constante		
9	DIR	<_SDAC>	INT	Sens de rotation de la broche	3 =	Pour M3
					4 =	Pour M4
10	F	<_FF1>	REAL	Avance jusqu'à la profondeur de réduction de vitesse de rotation		
11	FR	<_FF2>	REAL	Avance réduite jusqu'à la profondeur finale		
12	SR	<_SS2>	REAL	Vitesse de rotation réduite jusqu'à la profondeur finale		
13	XM	<_DIAGM>	REAL	Profondeur de sortie du récepteur de pièce (abs, toujours diamètre)		
14		<_VARI>	INT	Type d'usinage		
				UNITES :	Retrait	
					0 =	Retrait à <_SPD> + <_SDIS>
					1 =	Pas de retrait à la fin
				DIZAINES :	Récepteur de pièces	
					0 =	Non, n'exécuter aucune instruction M
					1 =	Oui, appel de CUST_TECHCYC(101)- ouverture du tiroir, CUST_TECHCYC(102)- fermeture du tiroir
15		<_DN>	INT	Numéro D pour 2e tranchant, si non programmé ⇒ D+1		
20		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
21		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur pour réduction de la vitesse de rotation (<_DIAG1>)	
					0 =	En absolu, valeur de l'axe transversal en diamètre
					1 =	En relatif, valeur de l'axe transversal en rayon
				DIZAINES :	Profondeur finale (<_DIAG2>)	
					0 =	En absolu, valeur de l'axe transversal en diamètre
					1 =	En relatif, valeur de l'axe transversal en rayon
				CENTAINES :	Réservé	
				MILLIERS :	Rayon/chanfrein (<_RC>)	
					0 =	Rayon
					1 =	Chanfrein
				DIZAINES DE MILLIERS :	Vitesse de rotation de la broche / vitesse de coupe (<_SV1>)	
					0 =	Vitesse de rotation de broche constante
					1 =	Vitesse de coupe constante

3.26.1.28 CYCLE95 - Chariotage au contour

Syntaxe

```
CYCLE95(<NPP>, <MID>, <FALZ>, <FALX>, <FAL>, <FF1>, <FF2>, <FF3>,
<_VARI>, <DT>, <DAM>, <_VRT>, <_GMODE>, <_DMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1	CON	<NPP>	STRING [140]	Nom du contour
2	D	<MID>	REAL	Profondeur de passe maximale en ébauchage, voir <_GMODE>
3	UZ	<FALZ>	REAL	Surépaisseur de finition en Z
4	UX	<FALX>	REAL	Surépaisseur de finition en X
5	U	<FAL>	REAL	Surépaisseur de finition parallèle au contour (prend effet dans les deux axes)
6	F	<FF1>	REAL	Avance pour ébauchage
7	FY	<FF2>	REAL	Avance de plongée Détalonnages
8	FS	<FF3>	REAL	Avance de finition

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
9		<_VARI>	INT	Type d'usinage		
				UNITES et DIZAINES :		
					1 =	Ebauchage, longitudinal, extérieur
					2 =	Ebauchage, transversal, extérieur
					3 =	Ebauchage, longitudinal, intérieur
					4 =	Ebauchage, transversal, intérieur
					5 =	Finition, longitudinale, extérieure
					6 =	Finition, transversale, extérieure
					7 =	Finition, longitudinale, intérieure
					8 =	Finition, transversale, intérieure
					9 =	Usinage complet, longitudinal, extérieur
					10 =	Usinage complet, transversal, extérieur
					11 =	Usinage complet, longitudinal, intérieur
					12 =	Usinage complet, transversal, intérieur
				CENTAINES :		
					0 =	Avec lissage sur le contour, sans matière restante dans les angles
					1 =	Sans lissage sur le contour
					2 =	Lissage uniquement jusqu'au point d'intersection précédent, de la matière peut rester dans les angles
10	DT	<DT>	REAL	Temporisation lors de l'interruption de l'avance		
11	DI	<DAM>	REAL	Ecart entre les interruptions de l'avance		
12	VRT	<_VRT>	REAL	Relèvement du contour		
					0 =	Un relèvement de 1 mm est utilisé en interne, indépendamment du système de mesure anglo-saxon / métrique
					> 0 =	Relèvement
13		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Evaluation de la profondeur de passe	
					0 =	Profondeur de passe calculée en fonction du groupe G DIAMON/ DIAMOF
					1 =	La profondeur de passe prend effet en tant que valeur de rayon (indépendamment de DIAMON/ DIAMOF)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
14		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				MILLIERS :		
					0 =	Mode de compatibilité : le nom du contour figure dans NPP
					1 =	Le nom du contour est programmé dans le CYCLE62 et transféré dans _SC_CONT_NAME

3.26.1.29 CYCLE98 - Concaténation de filetages

Syntaxe

```
CYCLE98(<_PO1>, <_DM1>, <_PO2>, <_DM2>, <_PO3>, <_DM3>, <_PO4>,
<_DM4>, <APP>, <ROP>, <TDEP>, <FAL>, <_IANG>, <NSP>, <NRC>, <NID>,
<_PP1>, <_PP2>, <_PP3>, <_VARI>, <_NUMTH>, <_VRT>, <_MID>, <_GDEP>,
<_IFLANK>, <_PITA>, <_PITM1>, <_PITM2>, <_PITM3>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1	Z0	<_PO1>	REAL	Point de référence en Z (abs)
2	X0	<_DM1>	REAL	Point de référence en X (abs), dans le diamètre
3	Z1	<_PO2>	REAL	Point intermédiaire 1 en Z (abs/rel), voir <_AMODE> (UNITES)
4	X1	<_DM2>	REAL	Point intermédiaire 1 en X (abs/rel), voir <_AMODE> (DIZAINES) ou
	X1α			Inclinaison du filetage 1 (-90° à 90°) abs toujours diamètre, rel toujours rayon
5	Z2	<_PO3>	REAL	Point intermédiaire 2 en Z (abs/rel), voir <_AMODE> (CENTAINES)
6	X2	<_DM3>	REAL	Point intermédiaire 2 en X (abs/rel), voir <_AMODE> (MILLIERS) ou
	X2α			Inclinaison du filetage 2 (-90° à 90°) abs toujours diamètre, rel toujours rayon
7	Z3	<_PO4>	REAL	Point final en Z (abs/rel), voir <_AMODE> (DIZAINES DE MILLIERS)
8	X3	<_DM4>	REAL	Point final en X (abs/rel), voir <_AMODE> (CENTAINES DE MILLIERS) ou
	X3α			Inclinaison du filetage 3 (-90° à 90°) abs toujours diamètre, rel toujours rayon

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
9	LW	<APP>	REAL	Course d'approche du filetage (rel, à saisir sans signe)		
10	LR	<ROP>	REAL	Sortie de filetage (rel, à saisir sans signe)		
11	H1	<TDEP>	REAL	Profondeur de filet (rel, à saisir sans signe)		
12	U	<FAL>	REAL	Surépaisseur de finition en X et Z		
13	DP	<_IANG>	REAL	Pente de pénétration en tant que distance ou angle, voir <_AMODE> (MILLIONS)		
	αP			La pente de pénétration prend effet en fonction du réglage du paramètre <_VARI> (CENTAINES).		
					Définition pour <_VARI_CENTAINES = 0 - mode compatibilité :	
					> 0 =	Pénétration oblique le long d'un flanc
					0 =	Pénétration perpendiculaire dans le filet
					< 0 =	Pénétration oblique avec flancs en alternance
					Définition pour _VARI_CENTAINES<>0 :	
					> 0 =	Pénétration sur front montant
					0 =	Pénétration centrale
					< 0 =	Pénétration sur front descendant
14	α0	<NSP>	REAL	Décalage de l'angle d'attaque pour le 1er filet		
15		<NRC>	INT	Nombre de passes d'ébauchage, voir <_VARI> (DIZAINES DE MILLIERS)		
16	NN	<NID>	INT	Nombre de passes à vide		
17	P0	<_PP1>	REAL	Pas de vis 1ère section du filet, voir <_PITA>		
18	P1	<_PP2>	REAL	Pas de vis 2e section du filet, voir <_PITA>		
19	P2	<_PP3>	REAL	Pas de vis 3e section du filet, voir <_PITA>		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
20		<_VARI>	INT	Usinage		
				UNITES :	Technologie	
					1 =	Filetage extérieur avec passe linéaire
					2 =	Filetage intérieur avec passe linéaire
					3 =	Filetage extérieur avec passe dégressive, la section du copeau reste constante
					4 =	Filetage intérieur avec passe dégressive, la section du copeau reste constante
				DIZAINES :	Réservé	
				CENTAINES :	Type de passe	
					0 =	Mode compatibilité pour <_IANG>
					1 =	Passe d'un seul côté
					2 =	Passe en alternance
				MILLIERS :	Réservé	
				DIZAINES DE MILLIERS :	Profondeur de passe alternative	
					0 =	Compatibilité, indication du nombre de passes d'ébauchage (<_NRC>)
					1 =	Indication d'une valeur pour la 1ère passe (<_MID>)
				CENTAINES DE MILLIERS :	Type d'usinage	
					0 =	Compatibilité (ébauchage et finition)
					1 =	Ebauchage
					2 =	Finition
					3 =	Ebauchage et finition
				MILLIONS :	Ordre d'exécution dans le cas d'un filetage multifilet	
					0 =	Ordre des filets croissant
					1 =	Ordre des filets par rapport à
21	N	<_NUMTH>	INT	Nombre de filets		
22		<_VRT>	REAL	Distance de retrait (rel)		
					0 =	Un relèvement de 1 mm est utilisé en interne, indépendamment du système de mesure anglo-saxon / métrique
					> 0 =	Relèvement
23	D1	<_MID>	REAL	Première passe, voir <_VARI> (DIZAINES DE MILLIERS)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
24	DA	<_GDEP>	REAL	Profondeur de changement de filet (ne prend effet que pour "multifilet")		
					0 =	Ne prendre en compte aucune profondeur de changement de filet
					> 0 =	Prendre en compte la profondeur de changement de filet
25		<_IFLANK>	REAL	Pente de pénétration en tant que largeur (uniquement pour la surface)		
26		<_PITA>	INT	Evaluation du pas de vis		
					0 =	Mode compatibilité pour le pas de vis, évaluation <_PP1> à <_PP3> comme précédemment, en fonction du système de mesure métrique / anglo-saxon
					1 =	Pas en mm
					2 =	Pas en TPI (nombre de filets par inch)
					3 =	Pas en inch
					4 =	MODULE
27		<_PITM1>	STRING[15]	Chaîne de caractères servant de mémento pour la saisie du pas de vis (réservé à l'interface)		
28		<_PITM2>	STRING[15]	Chaîne de caractères servant de mémento pour la saisie du pas de vis (réservé à l'interface)		
29		<_PITM3>	STRING[15]	Chaîne de caractères servant de mémento pour la saisie du pas de vis (réservé à l'interface)		
30		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	---	Réservé
				CENTAINES :	---	Réservé
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
31		<_AMODE>	INT	Mode alternatif		
				UNITES :	1. point intermédiaire en Z (Z1)	
					0 =	Absolu
					1 =	Relatif
				DIZAINES :	1. point intermédiaire en X (X1)	
					0 =	Absolu
					1 =	Relatif
					2 =	α
				CENTAINES :	2. point intermédiaire en Z (Z2)	
					0 =	Absolu
					1 =	Relatif
				MILLIERS :	2. point intermédiaire en X (X2)	
					0 =	Absolu
					1 =	Relatif
					2 =	α
				DIZAINES DE MILLIERS :	Point final en Z (Z3)	
					0 =	Absolu
					1 =	Relatif
				CENTAINES DE MILLIERS :	Point final en X (X3)	
					0 =	Absolu
					1 =	Relatif
					2 =	α
				MILLIONS :	Sélection de la pente de pénétration en tant qu'angle ou largeur	
					0 =	Angle de pénétration <_IANG>
					1 =	Pente de pénétration <_IFLANK>
				DIZAINES DE MILLIONS :	Passage unique / plusieurs passages	
					0 =	Mode compatibilité (l'angle d'attaque <_NSP> est évalué)
					1 =	Passage unique (avec décalage de l'angle d'attaque <_NSP>)
					2 =	Multifilet

3.26.1.30 CYCLE99 - Filetage au tour

Syntaxe

```
CYCLE99(<_SPL>, <_SPD>, <_FPL>, <_FPD>, <_APP>, <_ROP>, <_TDEP>,
<_FAL>, <_IANG>, <_NSP>, <_NRC>, <_NID>, <_PIT>, <_VARI>, <_NUMTH>,
<_SDIS>, <_MID>, <_GDEP>, <_PIT1>, <_FDEP>, <_GST>, <_GUD>,
<_IFLANK>, <_PITA>, <_PITM>, <_PTAB>, <_PTABA>, <_DMODE>, <_AMODE>,
<_S_XRS>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	Z0	<_SPL>	REAL	Point de référence (abs)		
2	X0	<_SPD>	REAL	Point de référence (abs, toujours diamètre)		
3	Z1	<_FPL>	REAL	Point final en association avec <_AMODE> (UNITES)		
4	X1	<_FPD>	REAL	Point final, en association avec <_AMODE> (DIZAINES)		
5	LW/LW2	<_APP>	REAL	Course d'approche du filetage en association avec <_AMODE> (CENTAINES) ou Entrée de filetage = sortie de filetage en association avec <_AMODE> (CENTAINES)		
6	LR	<_ROP>	REAL	Sortie de filetage		
7	H1	<_TDEP>	REAL	Profondeur de filet		
8	U	<_FAL>	REAL	Surépaisseur de finition en X et Z		
9	DP	<_IANG>	REAL	Pente de pénétration en tant que distance ou angle, en association avec <_AMODE> (MILLIERS)		
	αP				> 0 =	Pénétration sur front montant
					< 0 =	Pénétration sur front descendant
					0 =	Pénétration centrale
10	α0	<_NSP>	REAL	Décalage de l'angle d'attaque (effectif pour "passage unique" seulement)		
11	ND	<_NRC>	INT	Nombre de passes d'ébauchage, en association avec <_VARI> (DIZAINES DE MILLIERS)		
12	NN	<_NID>	INT	Nombre de passes à vide		
13	P	<_PIT>	REAL	Pas de vis en tant que valeur, en association avec <_PITA>		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
14		<_VARI>	INT	Type d'usinage		
				UNITES :	Technologie	
					1 =	Filetage extérieur avec passe linéaire
					2 =	Filetage intérieur avec passe linéaire
					3 =	Filetage extérieur avec passe dégressive, la section du copeau reste constante
					4 =	Filetage intérieur avec passe dégressive, la section du copeau reste constante
				DIZAINES :	Réservé	
				CENTAINES :	Type de passe	
					1 =	Passe d'un seul côté
					2 =	Passe en alternance
				MILLIERS :	Réservé	
				DIZAINES DE MILLIERS :	Profondeur de passe alternative	
					0 =	Indication du nombre de passes d'ébauchage (<_NRC>)
					1 =	Indication d'une valeur pour la 1ère passe (<_MID>)
				CENTAINES DE MILLIERS :	Type d'usinage	
					1 =	Ebauchage
					2 =	Finition
					3 =	Ebauchage et finition
				MILLIONS :	Ordre d'exécution dans le cas d'un filetage multifilet	
					0 =	Ordre des filets croissant
					1 =	Ordre des filets par rapport à
15	N	<_NUMTH>	INT	Nombre de filets		
16	VR	<_SDIS>	REAL	Distance de retrait, rel		
17	D1	<_MID>	REAL	Première profondeur de passe, en association avec <_VARI> (DIZAINES DE MILLIERS)		
18	DA	<_GDEP>	REAL	Profondeur de changement de filet (ne prend effet que pour "multifilet")		
					0 =	Ne prendre en compte aucune profondeur de changement de filet
					> 0 =	Prendre en compte la profondeur de changement de filet
19	G	<_PIT1>	REAL	Modification du pas par tour		
					0 =	Le pas de vis est constant (G33)
					> 0 =	Le pas de vis augmente (G34)
					< 0 =	Le pas de vis diminue (G35)
20		<_FDEP>	REAL	Profondeur de plongée (à saisir sans signe)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
21	N1	<_GST>	INT	Premier filet N1 = 1...N, en association avec <_AMODE> (CENTAINES DE MILLIERS)		
22		<_GUD>	INT	Réservé		
23		<_IFLANK>	REAL	Pente de pénétration en tant que largeur (uniquement pour la surface)		
24		<_PITA>	INT	Unité de mesure du pas de vis (évaluation PIT et/ou MPIT)		
					0 =	Pas en mm - évaluation MPIT/PIT
					1 =	Pas en mm - évaluation PIT
					2 =	Pas en TPI - évaluation PIT (nombre de filets par inch)
					3 =	Pas en inch - évaluation PIT
					4 =	MODULE - évaluation PIT
25		<_PITM>	STRING[15]	Chaîne de caractères servant de mémento pour la saisie du pas de vis (réservé à l'interface)[1)]		
26		<_PTAB>	STRING[20]	Chaîne de caractères pour table de filetages (réservé à l'interface)[1)]		
27		<_PTABA>	STRING[20]	Chaîne de caractères pour sélection dans la table de filetages (réservé à l'interface)[1)]		
28		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Mode de filetage	
					0 =	Filetage longitudinal
					1 =	Filetage transversal
					2 =	Filetage sur corps conique
				CENTAINES :	---	Réservé
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
29		<_AMODE>	INT	Mode alternatif		
				UNITES :	Longueur de filetage en Z	
					0 =	Absolu
					1 =	Relatif
				DIZAINES :	Longueur de filetage en X	
					0 =	En absolu, valeur de l'axe transversal en diamètre
					1 =	En relatif, valeur de l'axe transversal en rayon
					2 =	α
				CENTAINES :	Evaluation course/trajet d'approche <_APP>	
					0 =	Course d'approche du filetage <_APP>
					1 =	Entrée de filetage = sortie de filetage <_APP> = -<_ROP>
					2 =	Indiquer la distance de l'entrée de filetage <_APP> = -<_APP>
				MILLIERS :	Sélection de la pente de pénétration en tant qu'angle ou largeur	
					0 =	Angle de pénétration <_IANG>
					1 =	Pente de pénétration <_IFLANK>
				DIZAINES DE MILLIERS :	Passage unique / plusieurs passages	
					0 =	Passage unique (avec décalage de l'angle d'attaque <_NSP>)
					1 =	Multifilet
				CENTAINES DE MILLIERS :	Premier filet <_GST>	
					0 =	Usinage complet
					1 =	Commencer l'usinage à partir de ce filet
					2 =	Usiner uniquement ce filet
				MILLIONS :	Compensation de flèche pour filetage longitudinal	
					0 =	Hauteur de segment du filetage bombé XS
					1 =	Rayon du filetage bombé RS
30	XS / RS	<_S_XRS>	REAL	Compensation de flèche pour filetage longitudinal en association avec <_AMODE> : MILLIONS		

Remarque

[1)] Les paramètres <_PITM>, <_PTAB> et <_PTABA> sont utilisés uniquement lors de la sélection du filetage dans les tables de filetages du masque de saisie.
L'accès aux tables de filetages par la définition de cycle n'est pas possible pendant l'exécution du cycle.

3.26.1.31 CYCLE435 - Définir le système de coordonnées du dresseur

Syntaxe

```
CYCLE435(<_T>, <_DD>, <S_TA>, <S_DA>, <S_AD>, <S_AL>, <S_PVD>,
<S_PVL>, <S_PD>, <S_PL>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1		<_T>	STRING[32]	Nom d'outil de la meule		
2		<_DD>	INT	Numéro de tranchant de la meule		
3		<S_TA>	STRING[32]	Point de référence outil de dressage - Nom de l'outil de dressage		
4		<S_DA>	INT	Numéro de tranchant de l'outil de dressage		
5		<S_AD>	REAL	Profondeur de passe de dressage - diamètre		
6		<S_AL>	REAL	Profondeur de passe de dressage - plan		
7		<S_PVD>	REAL	Décalage de profilage - diamètre		
8		<S_PVL>	REAL	Décalage de profilage - plan		
9		<S_PD>	REAL	Surépaisseur de profilage - diamètre		
10		<S_PL>	REAL	Surépaisseur de profilage - plan		
11		<_AMODE>	INT	Mode alternatif		
				UNITES :	outil actif en fin de cycle	
					0 =	outil de dressage actif
					1 =	meule active

3.26.1.32 CYCLE495 - Profilage

Syntaxe

```
CYCLE495(<_T>, <_DD>, <_SC>, <_F>, <_VARI>, <_D>, <_DX>, <_DZ>,
<S_PA>, <S_N>, <_DMODE>, <_AMODE>, <S_FW>, <S_HW>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1		<_T>	STRING[20]	Nom d'outil de la meule		
2		<_DD>	INT	Numéro de tranchant de la meule		
3		<_SC>	REAL	Valeur de relèvement pour le contournement d'obstacles, en relatif		
4		<_F>	REAL	Avance profilage		
5		<_VARI>	INT	Type d'usinage		
				UNITES :	Type de profilage	
					1 =	Parallèle à l'axe
					2 =	Parallèle au contour
				DIZAINES :	Sens d'usinage	
					0 =	En tirant possible avec les positions de tranchant 1 à 4
					1 =	En poussant possible avec les positions de tranchant 1 à 4
					2 =	En alternance possible avec les positions de tranchant 1 à 8
					3 =	Début → fin possible avec les positions de tranchant 1 à 8
					4 =	Fin → début possible avec les positions de tranchant 1 à 8
				CENTAINES :	Direction de pénétration	
					1 =	Pénétration X- pour G18 ou Y- pour G19
					2 =	Pénétration X+ pour G18 ou Y+ pour G19
					3 =	Pénétration Z- pour G18 et pour G19
					4 =	Pénétration Z+ pour G18 et pour G19
6		<_D>	REAL	Profondeur de passe de dressage pour le type de profilage parallèle à l'axe		
7		<_DX>	REAL	Profondeur de passe de dressage X pour G18 ou Y pour G19 pour le type de profilage parallèle au contour		
8		<_DZ>	REAL	Profondeur de passe de dressage Z pour G18 et G19 pour le type de profilage parallèle au contour		
9		<S_PA>	REAL	Surépaisseur de profilage		
10		<S_N>	INT	Nombre de courses dans le programme de profilage		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
11		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
12		<_AMODE>	INT	Mode alternatif		
				UNITES :	Sélection profilage nouveau/poursuivre	
					1 =	Nouveau
					2 =	Poursuivre
				DIZAINES :	Sélection surépaisseur de profilage	
					0 =	Du contour brut au point le plus bas du contour
					1 =	Du contour brut au point le plus haut du contour
13		<S_FW>	REAL	Angle de débattement de l'outil de dressage		
14		<S_HW>	REAL	Equerre-support de l'outil de dressage		

Remarque

L'interprétation du contour lors du profilage avec CYCLE495 est effectuée via la procédure prédéfinie CONTDCON. Étant donné que CONTDCON n'est **pas** autorisé lorsque la correction du rayon d'outil (G41/G42) est active, la correction du rayon d'outil doit être désactivée avec G40 avant l'appel de CYCLE495.

3.26.1.33 CYCLE782 - Adapter à la charge

Remarque

Pour pouvoir utiliser CYCLE782, une licence est requise pour l'option "Adaptation de charge intelligente".

Syntaxe

```
CYCLE782(<S_MODE>, <S_TESTAXIS>, <S_VALUE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1		<S_MODE>	INT	Mode de traitement		
				UNITES :	Activation/désactivation	
					0 =	Désactiver l'adaptation
					1 =	Activer l'adaptation
					2 =	Déterminer et activer l'inertie de base de l'axe (fonction de mise en service)
				DIZAINES :	Variante de mesure (uniquement lorsque <S_MODE> POSITION DES CENTAINES = 0)	
					0 =	Mode normal
					1 =	Mode précis (par exemple, avec détermination du frottement, etc.)
				CENTAINES	Mesure ou spécification d'une valeur d'inertie	
					0 =	Mesure Dans ce mode, des déplacements sont effectués pour déterminer la charge.
					1 =	Spécification d'une valeur d'inertie Dans ce mode, aucun déplacement n'est effectué pour déterminer la charge. Au lieu de cela, une valeur d'inertie déterminée précédemment, par exemple par l'optimisation Servo automatique (AST), est transmise.
				MILLIERS	Affichage du résultat de mesure	
					0 =	Image des résultats de mesure ARRÊT
					1 =	Image des résultats de mesure MARCHE
				DIZAINES DE MILLIERS	Durée d'affichage du résultat de mesure	
					1 =	Désélection automatique de l'affichage après 8 s
					3 =	Acquittement avec Départ CN
				CENTAINES DE MILLIERS	Sélection de la charge [1]	
					0 =	Déterminer la charge totale : Axe vide + pièce
					1 =	Déterminer la charge séparément : Pièce
2		<S_TESTAXIS>	AXIS	Axe de canal : • pour lequel les adaptations doivent être activées. • pour lequel l'inertie doit être déterminée (uniquement lorsque <S_MODE> POSITION DES CENTAINES = 0)		
3		<S_VALUE>	REAL	Valeur d'inertie (uniquement lorsque <S_MODE> POSITION DES CENTAINES = 1)		

[1] La valeur est reprise automatiquement à partir du paramètre machine 52212 $MCS_FUNCTION_MASK_TECH Bit 18.

Remarque

La fonction de mise en service "Déterminer et activer l'inertie de base de l'axe" (<S_MODE> POSITION DES UNITÉS = 2) n'est pas prise en charge par l'interface utilisateur. Pour cela, l'appel de cycle doit être programmé. Le résultat de la mesure est consigné dans le paramètre machine 53350 $MAS_ILC_BASE_VALUE.

La valeur du paramètre "Sélection de la charge" (<S_MODE> POSITION DES CENTAINES DE MILLIERS) est lue par l'interface utilisateur à partir du paramètre machine 52212 $MCS_FUNCTION_MASK_TECH bit 18 et consignée dans l'appel de cycle.

La charge peut être déterminée et prescrite pour les axes linéaires comme pour les axes rotatifs. Pour les axes linéaires, c'est toujours une masse qui est affichée dans l'image de résultat de mesure ou prescrite dans l'interface utilisateur ; pour les axes rotatifs, c'est une inertie.

Indépendamment du type d'axe, c'est toujours une inertie qui est inscrite dans le paramètre machine 53350 $MAS_ILC_BASE_VALUE et dans l'adaptation.

Exemples

Exemple 1 :

Au début d'un programme CN, la charge de l'axe MX1 doit être déterminée et les adaptations ainsi actualisées. Le résultat doit être affiché et l'affichage acquitté avec Départ CN.

Code de programme

```
...
CYCLE782(31011,MX1)
...
```

Exemple 2 :

La valeur d'inertie déterminée par exemple via la fonction AST doit être reprise au cours du programme dans la variable _MY_VALUE, le cycle de charge doit être appelé avec cette valeur et les adaptations doivent ainsi être activées pour l'axe MX2. Aucun résultat ne doit être affiché.

Code de programme

```
DEF REAL _MY_VALUE
_MY_VALUE=...
...
CYCLE782(101,MX2,_MY_VALUE)
...
```

Exemple 3 :

Les adaptations doivent être désactivées pour l'axe MX2.

Code de programme

```
...
CYCLE782(0,MX2)
...
```

3.26.1.34 CYCLE800 - Pivotement plan / Pivotement outil / Orientation outil

Syntaxe

```
CYCLE800(<_FR>, <_TC>, <_ST>, <_MODE>, <_X0>, <_Y0>, <_Z0>, <_A>,
<_B>, <_C>, <_X1>, <_Y1>, <_Z1>, <_DIR>, <_FR_I>, <_DMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1		<_FR>	INT	Mode de dégagement :	0 =	Pas de dégagement
					1 =	Dégagement axe machine Z
					2 =	Dégagement axe machine Z puis XY
					3 =	Réservé
					4 =	Dégagement maximal dans le sens de l'outil
					5 =	Dégagement relatif dans le sens de l'outil
2		<_TC>	STRING[32]	Nom du bloc de données de pivotement :	""	(aucun nom) en présence d'un seul bloc de données de pivotement
					"0"	Désélection du bloc de données de pivotement (suppression des frames de pivotement)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
3		<_ST>	INT	Etat des transformations		
				UNITES :		
					0 =	Nouveau, suppression du plan orienté et recalcul avec les paramètres actuels
					1 =	Additif, le plan orienté est ajouté au plan orienté actif
				DIZAINES :	Asservir la pointe de l'outil oui/non (actif uniquement si la fonction est configurée lors de la MeS PIVOTEMENT)	
					0 =	Ne pas asservir la pointe de l'outil
					1 =	Asservir la pointe de l'outil (TRAORI)
				CENTAINES :	Approcher/pivoter l'outil (la fonction est affichée dans le masque de saisie PIVOTER Outil)	
					0 =	Ne pas approcher l'outil
					1 =	Approcher l'outil (de préférence fraise radiale)
					2 =	Pivoter l'outil de **tournage** (si la cinématique de l'axe B pour la technologie de tournage est configurée dans MeS Pivotement)
					3 =	Pivoter l'outil de **fraisage** (si la cinématique de l'axe B pour la technologie de tournage est configurée dans MeS Pivotement)
				MILLIERS :	Paramètre interne Pivotement en mode JOG	
				DIZAINES DE MILLIERS :	Voir paramètre direction <_DIR>	
					0 =	Pivotement "oui"
					1 =	Pivotement "non", sens "moins"[3)]
					2 =	Pivotement "non", sens "plus"[3)]
				CENTAINES DE MILLIERS :	Voir paramètre direction <_DIR>	
					0 =	Compatibilité
					1 =	Sélection du sens "moins" optimisé (uniquement pour l'interface utilisateur) [4)]
					2 =	Sélection du sens "plus" optimisé (uniquement pour l'interface utilisateur) [4)]

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
4		<_MODE> [5]	INT	Mode de pivotement : Evaluation de l'angle de pivotement et de la séquence de pivotements (codée sur bits)		
				Bit : 7 6	0 0 :	Angle de pivotement axe par axe -> voir paramètres <_A>, <_B>, <_C>
					0 1 :	Angle solide -> voir paramètres <_A>, <_B>[1]
					1 0 :	Angle projeté -> voir Paramètres <_A>, <_B>, <_C> [1]
					1 1 :	Mode de pivotement axes rotatifs en direct -> voir paramètres <_A>, <_B>[1]
				Bit : 5 4 3 2 1 0 (sans signification pour les angles solides)	x x x x 0 1	1re rotation _A autour de X
					x x x x 1 0	1re rotation _A autour de Y
					x x x x 1 1	1re rotation _A autour de Z
					x x 0 1 x x	2e rotation _B autour de X
					x x 1 0 x x	2e rotation _B autour de Y
					x x 1 1 x x	2e rotation _B autour de Z
					0 1 x x x x	3e rotation _C autour de X
					1 0 x x x x	3e rotation _C autour de Y
					1 1 x x x x	3e rotation _C autour de Z
5	X0	<_X0>	REAL	Point de référence X avant la rotation		
6	Y0	<_Y0>	REAL	Point de référence Y avant la rotation		
7	Z0	<_Z0>	REAL	Point de référence Z avant la rotation		
8	X(A)	<_A>	REAL	1ère rotation conforme au réglage du paramètre <_MODE>		
9	Y(B)	<_B>	REAL	2ème rotation conforme au réglage du paramètre <_MODE>		
10	Z(C)	<_C>	REAL	3ème rotation conforme au réglage du paramètre <_MODE>		
11	X1	<_X1>	REAL	Point de référence X après la rotation		
12	Y1	<_Y1>	REAL	Point de référence Y après la rotation		
13	Z1	<_Z1>	REAL	Point de référence Z après la rotation		
14	- ou +	<_DIR>	INT	Déclencher le déplacement des axes rotatifs (par défaut = -1)		
					-1 =	Positionner sur la plus petite valeur de l'axe rotatif 1 ou 2[2]
					+1 =	Positionner sur la plus grande valeur de l'axe rotatif 1 ou 2[2]
					0 =	Pivotement non (calcul du frame de pivotement uniquement)[1] [3]
15	FR	<_FR_I>	REAL	Valeur (rel) du dégagement relatif dans l'axe de l'outil		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
16		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Représentation de la valeur bêta pour orientation de l'outil	
					0 =	Valeur
					1 =	Flèche

Remarque

Si les paramètres de transfert ci-après sont programmés indirectement (en tant que paramètres), le masque de saisie n'est pas décompilé : <_FR>, <_ST>, <_TC>, <_MODE>, <_DIR>

[1)] Sélection possible, si configuré dans la fonction MeS PIVOTEMENT.

[2)] Sélection possible si la référence de sens est réglée sur l'axe rotatif 1 ou 2 dans MeS PIVOTEMENT.

Aucun champ de sélection en l'absence de référence de sens

[3)]La sélection Pivotement "non" peut être masquée SD 55221 bit 0

Pivotement "non" sens "moins" correspond à <_DIR> = 0 et _ST DIZAINES DE MILLIERS = 1

Pivotement "non" sens "plus" correspond à <_DIR> = 0 et _ST DIZAINES DE MILLIERS = 2

[4)] La sélection de sens de l'axe rotatif 1 ou 2 s'effectue également lorsque l'axe rotatif se trouve dans la position polaire avec la référence de sens (valeur de position égale à zéro).

[5)] Exemple de codage : rotation axe par axe, séquence de rotation ZYX

Binaire : 00011011 ; décimal : 27

Les descripteurs d'axe XYZ correspondent aux axes géométriques du canal CN. Les rotations autour des axes XYZ peuvent être exécutées individuellement. P. ex., l'ordre de rotation autour de ZXZ n'est pas autorisé dans un appel du CYCLE800.

3.26.1.35 CYCLE801 - Modèle de positions Réseau ou cadre

Syntaxe

```
CYCLE801(<_SPCA>, <_SPCO>, <_STA>, <_DIS1>, <_DIS2>, <_NUM1>,
<_NUM2>, <_VARI>, <_UMODE>, <_ANG1>, <_ANG2>, <_HIDE>, <_NSP>,
<_DMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	X0	<_SPCA>	REAL	Point de référence pour modèles de position (réseau/cadre) du 1er axe (abs)		
2	Y0	<_SPCO>	REAL	Point de référence pour modèles de position (réseau/cadre) dans le 2e axe (abs)		
3	α0	<_STA>	REAL	Angle de rotation de base (par rapport au 1er axe)	< 0 =	Rotation dans le sens horaire
					> 0 =	Rotation dans le sens antihoraire
4	L1	<_DIS1>	REAL	Distance entre les colonnes (écart de position du 1er axe, à saisir sans signe)		
5	L2	<_DIS2>	REAL	Distance entre les lignes (écart de position du 2e axe, à saisir sans signe)		
6	N1	<_NUM1>	INT	Nombre de colonnes		
7	N2	<_NUM2>	INT	Nombre de lignes		
8		<_VARI>	INT	Type d'usinage		
				UNITES :	Modèle de position	
					0 =	Réseau
					1 =	Cadre
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
9		<_UMODE>	INT	Réservé		
10	αX	<_ANG1>	REAL	Angle de déformation par rapport au 1er axe (inclinaison des lignes par rapport au 1er axe)		
					< 0 =	Mesure dans le sens horaire (0 à -90 degrés)
					> 0 =	Mesure dans le sens antihoraire (0 à 90 degrés)
11	αY	<_ANG2>	REAL	Angle de déformation par rapport au 2e axe (inclinaison des colonnes par rapport au 2e axe)		
					< 0 =	Mesure dans le sens horaire (0 à -90 degrés)
					> 0 =	Mesure dans le sens antihoraire (0 à 90 degrés)
12		<_HIDE>	STRING [200]	Positions masquées • 198 caractères max. • Indication du numéro de position consécutif, par exemple "1,3" (les positions 1 et 3 ne sont pas exécutées)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
13		<_NSP>	INT	Réservé		
14		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)

3.26.1.36 CYCLE802 - Positions quelconques

Syntaxe

```
CYCLE802(<_XA>, <_YA>, <_X0>, <_Y0>, <_X1>, <_Y1>, <_X2>, <_Y2>,
<_X3>, <_Y3>, <_X4>, <_Y4>, <_X5>, <_Y5>, <_X6>, <_Y6>, <_X7>, <_Y7>,
<_X8>, <_Y8>, <_VARI>, <_UMODE>, <_DMODE>, <S_ABA>, <S_AB0>,
<S_AB1>, <S_AB2>, <S_AB3>, <S_AB4>, <S_AB5>, <S_AB6>, <S_AB7>,
<S_AB8>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1		<_XA>	INT	Alternative pour toutes les positions X (valeur décimale à 9 caractères) Nombre de caractères : 876543210 (le caractère correspond à la position de perçage Xn)		
				Valeur du caractère :	1 =	Absolu (1ère position programmée, toujours absolue)
					2 =	Relatif
2		<_YA>	INT	Alternative pour toutes les positions Y (valeur décimale à 9 caractères) Nombre de caractères : 876543210 (le caractère correspond à la position de perçage Yn)		
				Valeur du caractère :	1 =	Absolu (1ère position programmée, toujours absolue)
					2 =	Relatif
3	X0	<_X0>	REAL	1. position X		
4	Y0	<_Y0>	REAL	1. position Y		
5	X1	<_X1>	REAL	2. position X		
6	Y1	<_Y1>	REAL	2. position Y		
7	X2	<_X2>	REAL	3. position X		
8	Y2	<_Y2>	REAL	3. position Y		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
9	X3	<_X3>	REAL	4. position X
10	Y3	<_Y3>	REAL	4. position Y
11	X4	<_X4>	REAL	5. position X
12	Y4	<_Y4>	REAL	5. position Y
13	X5	<_X5>	REAL	6. position X
14	Y5	<_Y5>	REAL	6. position Y
15	X6	<_X6>	REAL	7. position X
16	Y6	<_Y6>	REAL	7. position Y
17	X7	<_X7>	REAL	8. position X
18	Y7	<_Y7>	REAL	8. position Y
19	X8	<_X8>	REAL	9. position X
20	Y8	<_Y8>	REAL	9. position Y

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
21		<_VARI>	INT	Usinage		
				CENTAINES :	(uniquement pour appel à partir de Jobshop) (pour l'instant seuls 0 et 2 sont exploités)	
					0 =	Pas de blocage de broche
					1 =	Blocage de broche uniquement pour pénétration perpendiculaire avec G00 ou G01
					2 =	Blocage de broche pendant tout l'usinage
				MILLIERS :	Réservé	
				DIZAINES DE MILLIERS :	Modèle de position avec/sans axe rotatif – combinaison d'axes (avec <_VARI> CENTAINES DE MILLIERS)	
					0 =	XY (uniquement XY sans axe rotatif, compatibilité)
					1 =	X, Y ou Z et axe rotatif : XA, YB, ZC (1 axe rotatif avec axe géométrique autour duquel l'axe rotatif tourne)
					2 =	XY et axe rotatif : XYA, XYB, XYC (1 axe rotatif avec 1er et 2e axe géométrique, sans TRACYL)
				CENTAINES DE MILLIERS :	Axe rotatif	
					0 =	Sans axe rotatif (uniquement XY, compatibilité)
					1 =	Axe A (axe rotatif autour de X)
					2 =	Axe B (axe rotatif autour de Y)
					3 =	Axe C (axe rotatif autour de Z)
				DIZAINES DE MILLIONS + MILLIONS :	Modèle de positions avec axe rotatif – offset (pour plusieurs axes rotatifs autour du même axe ; quand l'indice est trop élevé, alors 1er axe)	
					00 =	1. Axe A, B ou C ou pour compatibilité
					01 =	2. Axe A, B ou C
					...	
					19 =	20. Axe A, B ou C
22		<_UMODE>	INT	Sélection de la broche à bloquer : (uniquement pour appel à partir de Jobshop) (appel du cycle utilisateur CUST_TECHCYC)		
					3 =	Bloquer/débloquer broche principale
					23 =	Bloquer/débloquer contre-broche

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
23		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
24		<S_ABA>	INT	Alternative pour toutes les positions AB (valeur décimale à 9 caractères) Nombre de caractères : 876543210 (le caractère correspond à la position ABn)		
				Valeur du caractère :	1 =	Absolu (1ère position programmée, toujours absolue)
					2 =	Relatif
25	A0	<S_AB0>	REAL	1. position d'axe rotatif pour modèle de position avec axe rotatif (en association avec <_VARI>)		
26	A1	<S_AB1>	REAL	2. position d'axe rotatif pour modèle de positions avec axe rotatif		
27	A2	<S_AB2>	REAL	3. position d'axe rotatif pour modèle de positions avec axe rotatif		
28	A3	<S_AB3>	REAL	4. position d'axe rotatif pour modèle de positions avec axe rotatif		
29	A4	<S_AB4>	REAL	5. position d'axe rotatif pour modèle de positions avec axe rotatif		
30	A5	<S_AB5>	REAL	6. position d'axe rotatif pour modèle de positions avec axe rotatif		
31	A6	<S_AB6>	REAL	7. position d'axe rotatif pour modèle de positions avec axe rotatif		
32	A7	<S_AB7>	REAL	8. position d'axe rotatif pour modèle de positions avec axe rotatif		
33	A8	<S_AB8>	REAL	9. position d'axe rotatif pour modèle de positions avec axe rotatif		

Remarque

Les positions inutiles des paramètres X1/Y1/A1 à X8/Y8/A8 peuvent être ignorées. Toutefois, les valeurs alternatives pour <_XA>, <_YA> et <S_ABA> doivent toujours être indiquées pour l'ensemble des 9 positions.

Pour le modèle de positions XA, YB ou ZC (un axe géométrique et un axe rotatif), l'axe du plan d'usinage dans lequel il n'y a pas de déplacement via le modèle de positions (Y pour G17 et XA) doit être positionné avant l'appel du cycle.

3.26.1.37 CYCLE806 - Tournage par interpolation

Remarque

Le tournage par interpolation avec CYCLE806 est une option soumise à licence !

Syntaxe

```
CYCLE806(<_MODE>, <S_XC>, <S_YC>, <_STA>, <_SV1>, <_SV2>, <_SDIR>,
<_AMODE>, <_DMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1		<_MODE>	INT	Mode		
				UNITES :	Activation/désactivation	
					0 =	Désélectionner le tournage par interpolation
					1 =	Sélectionner le tournage par interpolation
2	XC	<S_XC>	REAL	Centre de rotation X		
3	YC	<S_YC>	REAL	Centre de rotation Y		
4	Gamma	<_STA>	REAL	Angle gamma (rotation autour de l'axe d'outil) ; réservé		
5	S	<_SV1>	REAL	Vitesse de rotation de broche constante ou vitesse de coupe constante (voir <_AMODE> EINER)		
6	SV	<_SV2>	REAL	Vitesse de rotation maximale pour une vitesse de coupe constante (réservé)		
7		<_SDIR>	INT	Réservé		
					0 =	Sens de rotation à partir de l'outil
					3 =	Sens de rotation M3
					4 =	Sens de rotation M4
8		<_AMODE>	INT	Mode alternatif		
				UNITES :	Mouvement de broche (<_SV1>)	
					0 =	Vitesse de rotation de broche constante
					1 =	Vitesse de coupe constante
9		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/18/19	
					1 =	G17 (actif uniquement dans le cycle)
10	TC	<S_TRA>	STRING	• Nom de la transformation • vide : Utiliser la 1re transformation cinématique TRAINT		

3.26.1.38 CYCLE830 - Perçage profond 2

Syntaxe

```
CYCLE830(<RTP>, <RFP>, <SDIS>, <_DP>, <FDEP>, <_DAM>, <DTB>, <DTS>,
<FRF>, <VARI>, <_MDEP>, <_VRT>, <_DTD>, <_DIS1>, <S_FP>, <S_SDAC2>,
<S_SV2>, <S_FB>, <_SDAC>, <_SV1>, <S_SPOS>, <S_ZA>, <S_FA>, <S_ZP>,
<S_FS>, <S_ZS1>, <S_ZS2>, <S_N>, <S_ZD>, <S_FD>, <S_FR>, <S_SDAC3>,
```

```
<S_SV3>, <S_CON>, <S_COFF>, <_GMODE>, <_DMODE>, <_AMODE>,
<S_AMODE2>, <S_AMODE3>, <S_ZPV>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<RTP>	REAL	Plan de retrait (abs)		
2	Z0	<RFP>	REAL	Point de référence (abs)		
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, sans signe)		
4	Z1	<_DP>	REAL	Profondeur de perçage finale abs/rel (voir <_AMODE> UNITES)		
5	D	<FDEP>	REAL	1re profondeur de perçage pour bris de copeaux / débourrage absolue ou relative par rapport au point de référence avec pointage / sans pointage ou par rapport à la profondeur de perçage pilote (voir <_AMODE> DIZAINES DE MILLIERS)		
6	DF	<_DAM>	REAL	Valeur/pourcentage pour chaque passe suivante valeur ou pourcentage de dégression (voir <_AMODE> CENTAINES DE MILLIERS)		
7	DTB	<DTB>	REAL	Arrêt temporisé à chaque profondeur de perçage (voir <_AMODE> DIZAINES)		
8	DTS	<DTS>	REAL	Arrêt temporisé pour le débourrage au point de départ (voir <_AMODE> CENTAINES)		
9	FD1	<FRF>	REAL	Valeur en pourcentage de l'avance pour la première passe (voir <_AMODE> DIZAINES DE MILLIONS)		
10		<VARI>	INT	Usinage		
				UNITES :	Bris de copeaux / débourrage	
					0 =	En une passe
					1 =	Bris de copeaux
					2 =	Débourrage
					3 =	Bris de copeaux et débourrage
				DIZAINES :	Retrait lors du débourrage	
					0 =	A la profondeur de perçage pilote
					1 =	A la distance de sécurité
				CENTAINES :	Attaque souple	
					0 =	non
					1 =	Oui
				MILLIERS :	Perçage débouchant	
					0 =	non
					1 =	Oui
				DIZAINES DE MILLIERS :	Pointage / perçage pilote	
					0 =	Sans pointage
					1 =	Avec pointage
					2 =	Avec trou pilote
				CENTAINES DE MILLIERS :	Retrait	
					0 =	A la profondeur de perçage pilote
					1 =	Jusqu'au plan de retrait

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
11	V1	<_MDEP>	REAL	Profondeur de passe minimale relative (uniquement avec pourcentage pour dégression)		
12	V2	<_VRT>	REAL	Valeur de retrait relative après chaque usinage (uniquement pour bris de copeaux)		
					0 =	Valeur par défaut 1 mm
					> 0 =	Valeur de retrait variable
13	DT	<_DTD>	REAL	Arrêt temporisé à la profondeur de perçage finale (voir <_AMODE> MILLIERS)		
14	V3	<_DIS1>	REAL	Précote d'arrêt relative uniquement pour débourrage (voir <_AMODE> MILLIONS)		
15	FP	<S_FP>	REAL	Avance pour l'accostage du perçage pilote comme valeur ou en % (en association avec <S_AMODE2> CENTAINES)		
16		<S_SDAC2>	INT	Sens de rotation de la broche lors de l'accostage		
					3 =	M3
					4 =	M4
					5 =	M5 (par défaut)
17	SP	<S_SV2>	REAL	Accostage avec	Vitesse de rotation de broche constante (voir <S_AMODE2> DIZAINES DE MILLIONS)	
	V4				Vitesse de coupe constante	
				Vitesse de rotation de broche en % de la vitesse de rotation de perçage		
18	F	<S_FB>	REAL	Avance de perçage (voir <S_AMODE2> UNITES)		
19		<_SDAC>	REAL	Sens de rotation de broche lors du perçage		
					3 =	M3
					4 =	M4
20	S	<_SV1>	REAL	Perçage avec	Vitesse de rotation de broche constante (voir <S_AMODE2> MILLIONS)	
	V5				Vitesse de coupe constante	
21	SPOS	<S_SPOS>	REAL	Position de la broche, uniquement pour accostage avec M5		
22	ZA	<S_ZA>	REAL	Profondeur de pointage relative par rapport au point de référence ou absolue (voir <S_AMODE3> UNITES)		
23	FA	<S_FA>	REAL	Avance de pointage comme valeur ou en % (en association avec <S_AMODE2> DIZAINES)		
24	ZP	<S_ZP>	REAL	Perçage pilote relatif par rapport au point de référence, absolu ou facteur du diamètre de perçage (voir <S_AMODE3> DIZAINES)		
25	FS	<S_FS>	REAL	Avance de l'attaque comme valeur ou en % (en association avec <S_AMODE2> MILLIERS)		
26	ZS1	<S_ZS1>	REAL	Profondeur de chaque attaque avec avance constante (rel)		
27	ZS2	<S_ZS2>	REAL	Profondeur de chaque attaque pour augmentation d'avance (rel)		
28	N	<S_N>	INT	Nombre des coups de bris de copeaux avant chaque débourrage		
29	ZD	<S_ZD>	REAL	Profondeur de perçage restante relative par rapport à la profondeur de perçage finale ou absolue (en association avec <S_AMODE3> CENTAINES)		
30	FD	<S_FD>	REAL	Avance de perçage restante comme valeur ou en % (en association avec <S_AMODE2> DIZAINES DE MILLIERS)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
31	FR	<S_FR>	REAL	Avance de retrait (en association avec <S_AMODE2> CENTAINES DE MILLIERS)		
32		<S_SDAC3>	INT	Sens de rotation de broche lors du retrait		
					3 =	M3
					4 =	M4
					5 =	M5
33	SR	<S_SV3>	REAL	Retrait avec	Vitesse de rotation de broche constante (voir <S_AMODE2> CENTAINES DE MILLIONS)	
	V6				Vitesse de coupe constante	
				Vitesse de rotation de broche en % de la vitesse de rotation de perçage		
34	Liquide d'arrosage en marche	<S_CON>	STRING[10]	Arrosage en marche, instruction M ou appel de sous-programme		
35	Arrêt de l'arrosage	<S_COFF>	STRING[10]	Arrêt de l'arrosage, instruction M ou appel de sous-programme		
36		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Profondeur de perçage par rapport à pointe/corps	
					0 =	Pointe
					1 =	Corps
37		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :	Réservé	
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Entrée : Complète
					1 =	Entrée : Simplifiée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
38		<_AMODE>	INT	Mode alternatif 1		
				UNITES :	Profondeur de perçage = profondeur de perçage finale Z1 abs/rel	
					0 =	Relatif
					1 =	Absolu
				DIZAINES :	Temporisation à chaque profondeur de perçage DTB en secondes/tours	
					0 =	En secondes
					1 =	En tours
				CENTAINES :	Temporisation pour le débourrage DTS en secondes/tours	
					0 =	En secondes
					1 =	En tours
				MILLIERS :	Arrêt temporisé à la profondeur de perçage finale DT en secondes/tours	
					0 =	En secondes
					1 =	En tours
				DIZAINES DE MILLIERS :	1re profondeur de perçage D abs/rel	
					0 =	Relatif
					1 =	Absolu
				CENTAINES DE MILLIERS :	Valeur/pourcentage DF pour chaque passe suivante (dégression)	
					0 =	Valeur absolue
					1 =	Pourcentage (0,001 à 100 %)
				MILLIONS :	Précote d'arrêt V3 automatique/manuel	
					0 =	Automatique (calculée dans le cycle)
					1 =	Manuel (valeur programmée)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
39		<S_AMODE2>	INT	Mode alternatif 2		
				UNITES :	UNITES : Avance de perçage F	
					0 =	F/min
					1 =	F/tr
				DIZAINES :	Evaluation de l'avance de pointage FA	
					0 =	En % de l'avance de perçage
					1 =	F/min
					2 =	F/tr
				CENTAINES :	Evaluation de l'avance pour l'accostage dans le perçage pilote FP	
					0 =	En % de l'avance de perçage
					1 =	F/min
					2 =	F/tr
				MILLIERS :	Evaluation de l'avance d'attaque FS	
					0 =	En % de l'avance de perçage
					1 =	F/min
					2 =	F/tr
				DIZAINES DE MILLIERS :	Evaluation de l'avance de perçage débouchant FD	
					0 =	En % de l'avance de perçage
					1 =	F/min
					2 =	F/tr
				CENTAINES DE MILLIERS :	Avance de retrait FR	
					0 =	F/min
					1 =	Marche rapide
				MILLIONS :	Perçage - vitesse de rotation de broche / vitesse de coupe (S/V5)	
					0 =	Vitesse de rotation de broche constante
					1 =	Vitesse de coupe constante
				DIZAINES DE MILLIONS :	Accostage avec vitesse de rotation de broche / vitesse de coupe (SP/V4)	
					0 =	Vitesse de rotation de broche constante
					1 =	Vitesse de coupe constante
					2 =	Vitesse de rotation de broche en % de la vitesse de rotation de perçage
				CENTAINES DE MILLIONS :	Retrait - vitesse de rotation de broche / vitesse de coupe (SR/V6)	
					0 =	Vitesse de rotation de broche constante
					1 =	Vitesse de coupe constante

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
					2 =	Vitesse de rotation de broche en % de la vitesse de rotation de perçage
40		<S_AMODE3>	INT	Mode alternatif 3		
				UNITES :	Profondeur de pointage ZA abs/rel	
					0 =	Relatif
					1 =	Absolu
				DIZAINES :	Profondeur du perçage pilote ZP	
					0 =	Relatif
					1 =	Absolu
					2 =	Facteur du diamètre de perçage
				CENTAINES :	Profondeur de perçage restante ZD rel/abs	
					0 =	Relatif
					1 =	Absolu
41	ZPV	<S_ZPV>	REAL	Précote d'arrêt relative sur profondeur de perçage pilote		

3.26.1.39 CYCLE832 - Réglages High Speed

Syntaxe

```
CYCLE832(<S_TOL>, <S_TOLM>, <S_OTOL>)
```

Remarque

Le CYCLE832 ne décharge pas le constructeur de machines des tâches d'optimisation nécessaires lors de la mise en service de la machine. Ceci concerne l'optimisation des axes intervenant dans l'usinage et les réglages de la NCU (commande anticipatrice, limitation des à-coups, etc.).

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	Tolérance	<S_TOL>	REAL	Tolérance de contour La tolérance de contour correspond à la tolérance des axes géométriques.		
2		<S_TOLM>	INT	Type d'usinage (technologique)		
				UNITES :		
					0 =	Désactivation
					1 =	Finition (Finish)
					2 =	Préfinition (Semifinish)
					3 =	Ebauchage (Rough)
					4 =	Superfinition (Precision)
				DIZAINES :		
					0 =	Compatibilité[1] ou aucune tolérance d'orientation
					1 =	Tolérance d'orientation dans le paramètre <S_OTOL>
				CENTAINES ... CENTAINES DE MILLIERS	occupé pour cause de Compatibilité	
				MILLIONS :		
					0 =	Compatibilité. La meilleure fonction de moulage disponible est automatiquement utilisée : • Option Top Surface inactive : ⇒ Advanced Surface • Option Top Surface active : ⇒ Top Surface avec lissage
					1 =	Top Surface sans lissage
					2 =	Top Surface avec lissage
3	Tolérance ORI	<S_OTOL>	REAL	Tolérance d'orientation ou identificateur de version CYCLE832 Paramètre de tolérance pour l'orientation de l'outil. Nécessaire pour l'exécution d'un programme d'usinage utilisant des vitesses élevées sur des machines avec transformation d'orientation dynamique (p. ex. usinage 5 axes). La programmation du paramètre <S_OTOL> est **obligatoire**. Cette remarque s'applique également aux applications exécutées sur des machines 3 axes en présence de programmes sans orientation de l'outil (<S_OTOL> = 1).		

[1] Tolérance d'orientation dérivée des données de réglage de cycle SD55451 ... SD55454 (tolérance d'orientation pour mode dynamique...) ou SD55445 ... SD55449 (tolérance de contour pour mode dynamique...) multipliée par le facteur de SD55441 ... SD55444.
Informations complémentaires : Manuel de mise en service SINUMERIK Operate

Saisie de texte en clair

Pour améliorer la lisibilité de l'appel de cycle, le paramètre <S_TOLM> (type d'usinage) peut également être saisi en clair. Les textes en clair sont indépendants de la langue. Les saisies suivantes sont autorisées :

_OFF	pour	0	Désactivation
_FINISH	pour	1	Finition
_SEMIFIN	pour	2	Préfinition
_ROUGH	pour	3	Ebauche
_PRECISION	pour	4	Superfinition
_ORI_FINISH	pour	11	Finition avec saisie d'une tolérance d'orientation
_ORI_SEMIFIN	pour	12	Préfinition avec saisie d'une tolérance d'orientation
_ORI_ROUGH	pour	13	Ebauchage avec saisie d'une tolérance d'orientation
_ORI_PRECISION	pour	14	Superfinition avec saisie d'une tolérance d'orientation
_TOP_SURFACE_SMOOTH_OFF	pour	1000000	Top Surface sans lissage
_TOP_SURFACE_SMOOTH_ON	pour	2000000	Top Surface avec lissage

Pour la saisie de texte en clair avec Top Surface, les textes en clair sont combinés comme dans l'exemple suivant :

```
CYCLE832(0.1, _TOP_SURFACE_SMOOTH_OFF+_ORI_FINISH, 1)
```

Remarque

Les textes en clair s'appuient sur les noms de fonction du groupe G 59 (mode dynamique avec interpolation de trajectoire). Grâce à ces noms explicites, les machines 3 axes et les machines avec transformation d'orientation multi-axe (TRAORI) sont clairement séparées dans l'application.

Désélection de CYCLE832

Lorsque CYCLE832 est désélectionné, le paramètre <S_TOL> doit être transmis avec la valeur zéro.

Exemple : `CYCLE832(0,0,1)`

La syntaxe `CYCLE832()` est également autorisée pour la désélection de CYCLE832.

Exemples

Exemple 1 : CYCLE832 sur machine 3 axes sans transformation d'orientation

a) Appel de cycle avec saisie de texte en clair

Code de programme	Commentaire
G710	; Le système de mesure est le système métrique
CYCLE832(0.004,_FINISH,1)	; Appel du CYCLE832 avec : tolérance de contour = 0,004 mm, type d'usinage : finition
...	; Exécution d'un programme d'usinage utilisant des vitesses élevées

b) Appel du cycle sans saisie de texte en clair

Code de programme	Commentaire
G710	; cf. supra
CYCLE832(0.004,1,1)	; cf. supra
...	; cf. supra

Exemple 2 : CYCLE832 sur machine 5 axes avec transformation d'orientation

a) Appel de cycle et désélection avec saisie de texte en clair

Code de programme	Commentaire
G710	; Le système de mesure est le système métrique
TRAORI	; Activation de la transformation d'orientation
CYCLE832(0.3,_ORI_ROUGH,0.8)	; Appel du CYCLE832 avec : tolérance de contour = 0,3 mm, type d'usinage : ébauchage avec saisie d'une tolérance d'orientation, tolérance d'orientation = 0,8 degré
...	; Exécution d'un programme d'usinage utilisant des vitesses élevées
CYCLE832(0,_OFF,1)	; Tolérance de contour = 0, type d'usinage : Désélection de CYCLE832, Tolérance d'orientation = 0 degré

b) Appel du cycle et désélection sans saisie de texte en clair

Code de programme	Commentaire
G710	; cf. supra
TRAORI	; cf. supra
CYCLE832(0.3,13,0.8)	; cf. supra
...	; cf. supra
CYCLE832(0,0,1)	; cf. supra

3.26.1.40 CYCLE840 - Taraudage avec porte-taraud compensateur

Syntaxe

```
CYCLE840(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDR>, <SDAC>,
<ENC>, <MPIT>, <PIT>, <_AXN>, <_PITA>, <_TECHNO>, <_PITM>, <_PTAB>,
<_PTABA>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification			
1	RP	<RTP>	REAL	Plan de retrait (abs)			
2	Z0	<RFP>	REAL	Point de référence (abs)			
3	SC	<SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)			
4	Z1	<DP>	REAL	Profondeur de perçage (abs), voir <_AMODE>			
5	Z1	<DPR>	REAL	Profondeur de perçage (rel), voir <_AMODE>			
6	DT	<DTB>	REAL	Arrêt temporisé à la profondeur de perçage / à la distance de sécurité après le retrait, en secondes, voir <ENC>			
7		<SDR>	INT	Sens de rotation pour le retrait			
8	SDE	<SDAC>	INT	Sens de rotation après fin du cycle			
9		<ENC>	INT	Taraudage avec capteur de broche (G33) / taraudage sans capteur de broche (G63)			
					0 =	Avec capteur de broche	- Pas à partir de <MPIT>/ <PIT> - sans DT
					20 =	Avec capteur de broche	- Pas à partir de <MPIT>/<PIT> - avec DT après retrait à la distance de sécurité
					11 =	Sans capteur de broche	- Pas à partir de <MPIT>/ <PIT> - avec DT à la profondeur de perçage
					1 =	Sans capteur de broche	- Pas provenant de l'avance programmée - avec DT à la profondeur de perçage (avance = vitesse de rotation pas)
10		<MPIT>	REAL	Taille de filetage uniquement pour "ISO métrique" (le pas est calculé en interne pendant l'exécution) Plage de valeurs : 3 à 48 (pour M3 à M48), alternativement à <PIT>			
11		<PIT>	REAL	Pas de vis en tant que valeur, unité de mesure voir<_PITA> Plage de valeurs : > 0, en alternative à MPIT			

N°	Masque de paramètre	Paramètre interne	Type de données	Signification			
12		<_AXN>	INT	Axe de perçage	0 =	3. axe géométrique	
					1 =	1. axe géométrique	
					2 =	2. axe géométrique	
					≥ 3 =	3. axe géométrique	
13		<_PITA>	INT	Unité de mesure du pas de vis (évaluation <PIT> et <MPIT>)			
					0 =	Pas en mm	- Evaluation <MPIT>/<PIT>
					1 =	Pas en mm	- Evaluation <PIT>
					2 =	Pas en TPI	- Evaluation <PIT> (nombre de filets par inch)
					3 =	Pas en inch	- Evaluation <PIT>
					4 =	MODULE	- Evaluation <PIT>
14		<_TECHNO>	INT	Technologie[1]			
				UNITES :	Type d'arrêt précis		
					0 =	Type d'arrêt précis comme avant appel de cycle actif	
					1 =	Arrêt précis G601	
					2 =	Arrêt précis G602	
					3 =	Arrêt précis G603	
				DIZAINES :	Commande anticipatrice		
					0 =	Avec/sans commande anticipatrice actif comme avant l'appel de cycle	
					1 =	Avec commande anticipatrice FFWON	
					2 =	Sans commande anticipatrice FFWOF	
15		<_PITM>	STRING[15]	Chaîne de caractères servant de mémento pour la saisie du pas de vis[2]			
16		<_PTAB>	STRING[5]	Chaîne de caractères pour table de filetages ("", "ISO", "BSW", "BSP", "UNC")[2]			
17		<_PTABA>	STRING[20]	Chaîne de caractères pour la sélection dans la table de filetages (p. ex. "M 10", "M 12"...)[2]			
18		<_GMODE>	INT	Réservé			

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
19		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :	Mode de compatibilité (uniquement pour décompilation du masque de saisie), lorsque PM 52216 bit 0 = 1[1)]	
					0 =	Les paramètres technologiques sont affichés (compatibilité) : les paramètres TECHNO prennent effet
					1 =	Les paramètres technologiques ne sont pas affichés : la technologie "programmée avant l'appel de cycle" reste effective
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée
20		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de perçage Z1 (abs/rel)	
					0 =	Compatibilité issue de la programmation <DP>/<DPR>
					1 =	Relatif
					2 =	Absolu

[1)] Les champs Technologie peuvent être masqués, en fonction de la donnée de réglage SD52216 MCS_FUNCTION_MASK_DRILL

[2)] Les paramètres 15, 16 et 17 sont utilisés uniquement lors de la sélection du filetage dans les tables de filetages du masque de saisie. L'accès aux tables de filetages par la définition de cycle n'est pas possible pendant l'exécution du cycle.

3.26.1.41 CYCLE899 - Rainure borgne

Syntaxe

```
CYCLE899(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_LENG>, <_WID>, <_PA>,
<_PO>, <_STA>, <_MID>, <_MIDA>, <_FAL>, <_FALD>, <_FFP1>, <_CDIR>,
<_VARI>, <_GMODE>, <_DMODE>, <_AMODE>, <_UMODE>, <_FS>, <_ZFS>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	RP	<_RTP>	REAL	Plan de retrait (abs)		
2	Z0	<_RFP>	REAL	Point de référence de l'axe outil (abs)		
3	SC	<_SDIS>	REAL	Distance de sécurité (ajoutée au point de référence, à saisir sans signe)		
4	Z1	<_DP>	REAL	Profondeur de rainure (abs/rel), voir <_AMODE>		
5	L	<_LENG>	REAL	Longueur de la rainure (rel)		
6	W	<_WID>	REAL	Largeur de la rainure (rel)		
7	X0	<_PA>	REAL	Point de référence/position de départ 1er axe (abs)		
8	Y0	<_PO>	REAL	Point de référence/position de départ 2e axe (abs)		
9	α0	<_STA>	REAL	Angle de rotation par rapport au 1er axe		
10	DZ	<_MID>	REAL	Profondeur de passe maximale (rel), uniquement pour le fraisage en hélice		
11	DXY	<_MIDA>	REAL	Profondeur de passe maximale dans le plan, voir <_AMODE>		
12	UXY	<_FAL>	REAL	Surépaisseur de finition dans le plan		
13	UZ	<_FALD>	REAL	Surépaisseur de finition au fond		
14	F	<_FFP1>	REAL	Avance		
15		<_CDIR>	INT	Sens de fraisage		
				UNITES :		
					0 =	En avalant
					1 =	En opposition
					4 =	En alternance
16		<_VARI>	INT	Usinage		
				UNITES :		
					1 =	Ebauchage
					2 =	Finition
					3 =	Finition du fond
					4 =	Finition du bord
					5 =	Préfinition
					6 =	Chanfreinage
				DIZAINES :	Réservé	
				CENTAINES :	Réservé	
				MILLIERS :		
					1 =	Fraisage en hélice
					2 =	Fraisage en plongée

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
17		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération / calcul du point de départ uniquement	
					1 =	Usinage normal
				MILLIERS :	Cotation à partir du milieu / de l'arête	
					0 =	Cotation à partir du milieu
					1 =	Cotation à partir de l'arête "gauche" (sens "-" du 1er axe)
					2 =	Cotation à partir de l'arête "droite" (sens "+" du 1er axe)
18		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	---	Réservé
				CENTAINES :	---	Réservé
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée
19		<_AMODE>	INT	Mode alternatif		
				UNITES :	Profondeur de rainure Z1	
					0 =	Absolu
					1 =	Relatif
				DIZAINES :	Unité pour la profondeur de passe dans le plan (<_MIDA>) DXY	
					0 =	mm
					1 =	% du diamètre de l'outil
				CENTAINES :	Profondeur de pénétration lors du chanfreinage ZFS	
					0 =	Absolu
					1 =	Relatif
20		<_UMODE>	INT	Réservé		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
21	FS	<_FS>	REAL	Largeur de chanfrein (rel)
22	ZFS	<_ZFS>	REAL	Profondeur de pénétration (pointe de l'outil) lors du chanfreinage (abs/rel), voir <_AMODE>

3.26.1.42 CYCLE930 - Gorge

Syntaxe

```
CYCLE930(<_SPD>, <_SPL>, <_WIDG>, <_WIDG2>, <_DIAG>, <_DIAG2>,
<_STA>, <_ANG1>, <_ANG2>, <_RCO1>, <_RCI1>, <_RCI2>, <_RCO2>,
<_FAL>, <_IDEP1>, <_SDIS>, <_VARI>, <_DN>, <_NUM>, <_DBH>, <_FF1>,
<_NR>, <_FALX>, <_FALZ>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1	X0	<_SPD>	REAL	Point de référence dans l'axe transversal (toujours diamètre)		
2	Z0	<_SPL>	REAL	Point de référence dans l'axe longitudinal		
3	B1	<_WIDG>	REAL	Largeur de gorge en bas		
4	B2	<_WIDG2>	REAL	Largeur de gorge en haut (uniquement pour la surface)		
5	T1	<_DIAG>	REAL	Profondeur de gorge au point de référence, pour abs et usinage longitudinal = diamètre, sinon rel		
6	T2	<_DIAG2>	REAL	Profondeur de gorge par rapport au point de référence (uniquement pour la surface), pour abs et usinage longitudinal = diamètre, sinon rel		
7	α0	<_STA>	REAL	Angle des obliques (-180 ≤ <_STA> ≤ 180)		
8	α1	<_ANG1>	REAL	Angle de flanc 1 (0 ≤ <_ANG1> < 90) sur le côté de la gorge déterminé par le point de référence		
9	α2	<_ANG2>	REAL	Angle de flanc 2 (0 ≤ <_ANG2> < 90) par rapport au point de référence		
10	R1/FS1	<_RCO1>	REAL	Rayon d'arrondi ou largeur de chanfrein 1, à l'extérieur du point de référence		
11	R2/FS2	<_RCI1>	REAL	Rayon d'arrondi ou largeur de chanfrein 2, à l'intérieur du point de référence		
12	R3/FS3	<_RCI2>	REAL	Rayon d'arrondi ou largeur de chanfrein 3, à l'intérieur par rapport au point de référence		
13	R4/FS4	<_RCO2>	REAL	Rayon d'arrondi ou largeur de chanfrein 4, à l'extérieur par rapport au point de référence		
14	U	<_FAL>	REAL	Surépaisseur de finition en X et Z, voir <_VARI> (DIZAINES DE MILLIERS) (à saisir sans signe)		
15	D	<_IDEP1>	REAL	Profondeur de passe maximale lors de la plongée (à saisir sans signe)		
					0 =	1. passe directement sur toute la profondeur
					> 0 =	1. passe <_IDEP1>, 2e passe 2 · <_IDEP1>, etc.

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
16	SC	<_SDIS>	REAL	Distance de sécurité (à saisir sans signe)		
17		<_VARI>	INT	Type d'usinage		
				UNITES :	Réservé	
				DIZAINES :	Usinage technologique	
					1 =	Ebauchage
					2 =	Finition
					3 =	Ebauchage et finition
				CENTAINES :	Position longitudinale/transversale extérieure/intérieure +Z/+Z ou +X/-X	
					1 =	Longitudinal/extérieur +Z
					2 =	Transversal/intérieur -X
					3 =	Longitudinal/intérieur +Z
					4 =	Transversal/intérieur +X
					5 =	Longitudinal/extérieur -Z
					6 =	Transversal/extérieur -X
					7 =	Longitudinal/intérieur -Z
					8 =	Transversal/extérieur +X
				MILLIERS :	Position du point de référence	
					0 =	Point de référence supérieur
					1 =	Point de référence inférieur
				DIZAINES DE MILLIERS :	Définition de la prise d'effet des surépaisseurs de finition	
					0 =	Surépaisseur de finition U parallèle au contour
					1 =	Surépaisseurs de finition UX et UZ séparées
18		<_DN>	INT	Numéro D pour 2e tranchant de l'outil		
					> 0 =	Numéro D pour la correction d'outil du 2e tranchant de l'outil à plonger
					0 =	Pas de 2e tranchant programmé
19	N	<_NUM>	INT	Nombre de gorges (0 = 1 gorge)		
20	DP	<_DBH>	REAL	Distance entre gorges (requis uniquement lorsque <_NUM> > 1)		
21	F	<_FF1>	REAL	Avance		
22		<_NR>	INT	Identificateur de la forme de la gorge (correspond à la touche logicielle verticale de sélection de la forme)		
					0 =	Flancs de 90° sans chanfreins/ arrondis
					1 =	Flancs obliques avec chanfreins/ arrondis (sans α0)
					2 =	Comme 1, mais sur cône (avec α0)
23	UX	<_FALX>	REAL	Surépaisseur de finition dans l'axe X, voir <_VARI> (DIZAINES DE MILLIERS) (à saisir sans signe)		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
24	UZ	<_FALZ>	REAL	Surépaisseur de finition dans l'axe Z, voir <_VARI> (DIZAINES DE MILLIERS) (à saisir sans signe)		
25		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
26		<_AMODE>	INT	Mode alternatif		
				UNITES :	Cotation de la profondeur (uniquement pour la surface)	
					0 =	Au point de référence
					1 =	Par rapport au point de référence
				DIZAINES :	Profondeur	
					0 =	Absolu
					1 =	Relatif
				CENTAINES :	Cotation de la largeur (uniquement pour la surface)	
					0 =	Sur diamètre externe (en haut)
					1 =	Sur diamètre interne (en bas)
				MILLIERS :	Rayon/chanfrein 1 (<_RCO1>)	
					0 =	Rayon
					1 =	Chanfrein
				DIZAINES DE MILLIERS :	Rayon/chanfrein 2 (<_RCI1>)	
					0 =	Rayon
					1 =	Chanfrein
				CENTAINES DE MILLIERS :	Rayon/chanfrein 3 (<_RCI2>)	
					0 =	Rayon
					1 =	Chanfrein
				MILLIONS :	Rayon/chanfrein 4 (<_RCO2>)	
					0 =	Rayon
					1 =	Chanfrein

3.26.1.43 CYCLE940 - Dégagement forme E et F / Dégagement filet

Le cycle CYCLE940 permet de programmer différents dégagements. Ceux-ci peuvent différer considérablement les uns des autres en fonction du paramétrage.

Les colonnes supplémentaires du tableau indiquent quel paramètre est requis pour chaque forme de dégagement. Elles correspondent aux touches logicielles de sélection verticales du masque de cycle :

- E : dégagement forme E
- F : dégagement forme F
- A-D : dégagement filet DIN (formes A-D)
- T : dégagement filet (définition libre de la forme)

Syntaxe

```
CYCLE940(<_SPD>, <_SPL>, <_FORM>, <_LAGE>, <_SDIS>, <_FFP>, <_VARI>,
<_EPD>, <_EPL>, <_R1>, <_R2>, <_STA>, <_VRT>, <_MID>, <_FAL>,
<_FALX>, <_FALZ>, <_PITI>, <_PTAB>, <_PTABA>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètres	Paramètre interne	Type de données	Progr. pour la forme				Signification		
				E	F	A-D	T			
1	X0	<_SPD>	REAL	x	x	x	x	Point de référence dans l'axe transversal (toujours diamètre)		
2	Z0	<_SPL>	REAL	x	x	x	x	Point de référence de l'axe longitudinal (abs)		
3	FORM	<_FORM>	CHAR	x	x	x	x	Forme du dégagement (en majuscules, p. ex. "T") Sélection, table à partir de laquelle les valeurs de dégagement doivent être prises		
									A =	Extérieur, référence DIN76, A = normal
									B =	Extérieur, référence DIN76, B = court
									C =	Intérieur, référence DIN76, C = normal
									D =	Intérieur, référence DIN76, D = court
									E =	Référence DIN509
									F =	Référence DIN509
									T =	Forme libre
4	LAGE	<_LAGE>	INT	x	x	x	x	Position du dégagement (parallèle Z)	0 =	Extérieur +Z : ____|
									1 =	Extérieur -Z : |____/
									2 =	Intérieur +Z : /-----|
									3 =	Intérieur -Z : |-----\
5	SC	<_SDIS>		x	x	x	x	Distance de sécurité (rel)		
6	F	<_FFP>		x	x	x	x	Avance d'usinage (mm/tr)		

N°	Masque de paramètres	Paramètre interne	Type de données	Progr. pour la forme				Signification			
7		<_VARI>	INT	-	-	x	x	Type d'usinage			
								UNITES :	Usinage		
									1 =	Ebauchage	
									2 =	Finition	
									3 =	Ebauchage + finition	
								DIZAINES :	Stratégie d'usinage		
									0 =	Parallèle au contour	
									1 =	Longitudinal	
				Les dégagements de forme E et F sont toujours usinés en une seule passe, comme pour la finition.							
8	X1	<_EPD>		x	x	-	-	Surépaisseur X (abs/rel), voir <_AMODE>			
				-	-	-	x	Profondeur de dégagement (abs/rel), voir <_AMODE>			
9	Z1	<_EPL>		-	x	-	-	Surépaisseur Z			
				-	-	-	x	Largeur de dégagement (abs/rel), voir <_AMODE>			
10	R1	<_R1>		-	-	-	x	Rayon d'arrondi sur les obliques			
11	R2	<_R2>		-	-	-	x	Rayon d'arrondi dans le coin			
12	α	<_STA>		-	-	x	x	Angle de pénétration			
13	VX	<_VRT>		x	x	-	-	Chariotage transversal X (abs/rel), voir <_AMODE>			
				-	-	x	x	Chariotage transversal X pour finition (abs/rel), voir <_AMODE>			
14	D	<_MID>		-	-	x	x	Profondeur de passe			
15	U	<_FAL>		-	-	x	x	Surépaisseur de finition parallèle au contour, voir <_AMODE>			
16	UX	<_FALX>		-	-	x	x	Surépaisseur de finition X			
17	UZ	<_FALZ>		-	-	x	x	Surépaisseur de finition Z			
18	P	<_PITI>	INT	-	-	x	-	Sélection du pas, forme A-D, correspond à M1 ... M68			
								0 = 0,20 1 = 0,25 2 = 0,30 3 = 0,35 4 = 0,40 5 = 0,45	6 = 0,50 7 = 0,60 8 = 0,70 9 = 0,75 10 = 0,80 11 = 1,00	12 = 1,25 13 = 1,50 14 = 1,75 15 = 2,00 16 = 2,50 17 = 3,00	18 = 3,50 19 = 4,00 20 = 4,50 21 = 5,00 22 = 5,50 23 = 6,00
				x	x	-	-	Sélection rayon/profondeur, forme E, F			
								0 = 0,6 · 0,3 1 = 1,0 · 0,4 2 = 1,0 · 0,2 3 = 1,6 ··0,3	4 = 2,5 · 0,4 5 = 4,0 · 0,5 6 = 0,4 · 0,2 7 = 0,6 · 0,2	8 = 0,1 · 0,1 9 = 0,2 ··0,1	
19		<_PTAB>	STRING [5]					Chaîne de caractères pour table de filetages ("", "ISO", "BSW", "BSP", "UNC") (réservé à l'interface)			
20		<_PTABA>	STRING [20]					Chaîne de caractères pour la sélection dans la table de filetages (p. ex. "M 10", "M 12"...) (réservé à l'interface)			

N°	Masque de paramètres	Paramètre interne	Type de données	Progr. pour la forme				Signification		
21		<_DMODE>	INT					Mode d'affichage		
				x	x	x	x	UNITES :	Plan d'usinage G17/G18/G19	
									0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
									1 =	G17 (actif uniquement dans le cycle)
									2 =	G18 (actif uniquement dans le cycle)
									3 =	G19 (actif uniquement dans le cycle)
22		<_AMODE>	INT					Mode alternatif		
				x	x	-	x	UNITES :	Paramètre <_EPD> surépaisseur X ou profondeur de dégagement	
									0 =	Absolu (toujours diamètre)
									1 =	Relatif
				x	x	-	x	DIZAINES :	Paramètre <_EPL> surépaisseur Z ou largeur de dégagement	
									0 =	Absolu
									1 =	Relatif
				x	x	x	x	CENTAINES :	Paramètre <_VRT> chariotage transversal X	
									0 =	Absolu (toujours diamètre)
									1 =	Relatif
				-	-	x	x	MILLIERS :	Surépaisseur de finition	
									0 =	Surépaisseur de finition parallèle au contour (<_FAL>)
									1 =	Surépaisseur de finition distincte (<_FALX>/<_FALZ>)

3.26.1.44 CYCLE951 - Chariotage

Syntaxe

```
CYCLE951(<_SPD>, <_SPL>, <_EPD>, <_EPL>, <_ZPD>, <_ZPL>, <_LAGE>,
<_MID>, <_FALX>, <_FALZ>, <_VARI>, <_RF1>, <_RF2>, <_RF3>, <_SDIS>,
<_FF1>, <_NR>, <_DMODE>, <_AMODE>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1	X0	<_SPD>	REAL	Point de référence (abs, toujours diamètre)
2	Z0	<_SPL>	REAL	Point de référence (abs)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
3	X1	<_EPD>	REAL	Point final		
4	Z1	<_EPL>	REAL	Point final		
5	XM α1 α2	<_ZPD>	REAL	Point intermédiaire, voir <_DMODE> (DIZAINES)		
6	ZM α1 α2	<_ZPL>	REAL	Point intermédiaire, voir <_DMODE> (DIZAINES)		
7	Position	<_LAGE>	INT	Position du coin de chariotage	0 =	Extérieur/arrière
					1 =	Extérieur/avant
					2 =	Intérieur/arrière
					3 =	Intérieur/avant
8	D	<_MID>	REAL	Profondeur de passe maximale lors de la pénétration		
9	UX	<_FALX>	REAL	Surépaisseur de finition en X		
10	UZ	<_FALZ>	REAL	Surépaisseur de finition en Z		
11		<_VARI>	INT	Type d'usinage		
				UNITES :	Direction de chariotage (longitudinale ou transversale) dans le système de coordonnées	
					1 =	Longitudinal
					2 =	Transversal
				DIZAINES :		
					1 =	Ebauchage jusqu'à surépaisseur de finition
					2 =	Finition
				CENTAINES :	Réservé	
				MILLIERS :	Réservé	
				DIZAINES DE MILLIERS :	Réservé	
12	R1/FS1	<_RF1>	REAL	Rayon d'arrondi ou largeur de chanfrein 1, voir <_AMODE> (DIZAINES DE MILLIERS)		
13	R2/FS2	<_RF2>	REAL	Rayon d'arrondi ou largeur de chanfrein 2, voir <_AMODE> (CENTAINES DE MILLIERS)		
14	R3/FS3	<_RF3>	REAL	Rayon d'arrondi ou largeur de chanfrein 3, voir <_AMODE> (MILLIONS)		
15	SC	<_SDIS>	REAL	Distance de sécurité		
16	F	<_FF1>	REAL	Avance pour ébauchage/finition		
17		<_NR>	INT	Identificateur du mode de chariotage (correspond à la touche logicielle verticale de sélection de la forme) :		
					0 =	Chariotage 1, 90 degrés, coin sans chanfreins/arrondis
					1 =	Chariotage 2, 90 degrés, coin avec chanfreins/arrondis
					2 =	Chariotage 3, coin quelconque avec chanfreins/arrondis

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
18		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Forme de saisie <_ZPD>/<_ZPL>	
					0 =	Xm/Zm
					1 =	Xm/α1
					2 =	Xm/α2
					3 =	α1/Zm
					4 =	α2/Zm
					5 =	α1/α2
21		<_AMODE>	INT	Mode alternatif		
				UNITES :	Point intermédiaire en X	
					0 =	En absolu, valeur de l'axe transversal en diamètre
					1 =	En relatif, valeur de l'axe transversal en rayon
				DIZAINES :	Point intermédiaire en Z	
					0 =	Absolu
					1 =	Relatif
				CENTAINES :	Point final en X	
					0 =	En absolu, valeur de l'axe transversal en diamètre
					1 =	En relatif, valeur de l'axe transversal en rayon
				MILLIERS :	Point final en Z	
					0 =	Absolu
					1 =	Relatif
				DIZAINES DE MILLIERS :	Rayon/Chanfrein 1	
					0 =	Rayon
					1 =	Chanfrein
				CENTAINES DE MILLIERS :	Rayon/Chanfrein 2	
					0 =	Rayon
					1 =	Chanfrein
				MILLIONS :	Rayon/Chanfrein 3	
					0 =	Rayon
					1 =	Chanfrein

3.26.1.45 CYCLE952 - Chariotage / Chariotage reste / Plongée / Plongée reste / Plongée G +D / Plongée G+D reste

Syntaxe

```
CYCLE952(<_PRG>, <_CON>, <_CONR>, <_VARI>, <_F>, <_FR>, <_RP>, <_D>,
<_DX>, <_DZ>, <_UX>, <_UZ>, <_U>, <_U1>, <_BL>, <_XD>, <_ZD>, <_XA>,
<_ZA>, <_XB>, <_ZB>, <_XDA>, <_XDB>, <_N>, <_DP>, <_DI>, <_SC>,
<_DN>, <_GMODE>, <_DMODE>, <_AMODE>, <_PK>, <_DCH>, <_FS>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification
1	PRG	<_PRG>	STRING[100]	Nom du programme de chariotage
2	CON	<_CON>	STRING[100]	Nom du programme à partir duquel est lu le contour de pièce brute actualisé (pour usinage de la matière restante)
3	CONR	<_CONR>	STRING[100]	Nom du programme dans lequel est écrit le contour de pièce brute actualisé (voir <_AMODE> DIZAINES DE MILLIERS)

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
4		<_VARI>	INT	Type d'usinage		
				UNITES :	Mode de chariotage	
					1 =	Longitudinal
					2 =	Transversal
					3 =	Parallèle au contour
				DIZAINES :	Usinage technologique (voir <_GMODE> CENTAINES)	
					1 =	Ebauchage
					2 =	Finition
					3 =	Réservé
					4 =	Ebauchage sur deux canaux
					5 =	Finition sur deux canaux
				CENTAINES :	Sens d'usinage	
					1 =	Sens d'usinage X -
					2 =	Sens d'usinage X +
					3 =	Sens d'usinage Z -
					4 =	Sens d'usinage Z +
				MILLIERS :	Direction de pénétration	
					1 =	Extérieur X -
					2 =	Intérieur X +
					3 =	Face frontale Z -
					4 =	Face arrière Z +
				DIZAINES DE MILLIERS :	Définition de la prise d'effet des surépaisseurs de finition	
					0 =	Surépaisseurs de finition UX et UZ séparées
					1 =	Surépaisseur de finition U parallèle au contour
				CENTAINES DE MILLIERS :	Lissage	
					0 =	Compatibilité, lissage automatique
					1 =	Avec lissage sur le contour
					2 =	Sans lissage
					3 =	Lissage automatique
				MILLIONS :	Détalonnages	
					0 =	La position n'est pas évaluée pour plongée, - matière restante et plongée G+D, - matière restante
					1 =	Usinage des détalonnages
					2 =	Ne pas usiner les détalonnages
				DIZAINES DE MILLIONS :	Derrière/devant le centre de rotation	
					0 =	Usinage devant le centre de rotation
					1 =	Réservé

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
5	F	<_F>	REAL	Avance pour ébauchage/finition		
	FZ			Avance abscisse de plongée G+D		
6	FR	<_FR>	REAL	Avance pour la pénétration dans les détalonnages d'ébauchage		
	FX			Avance de l'ordonnée de plongée G+D		
7	RP	<_RP>	REAL	Plan de retrait pour usinage intérieur (abs, toujours diamètre)		
8	D	<_D>	REAL	Passe d'ébauchage (voir <_AMODE> UNITES)		
9	DX	<_DX>	REAL	Passe X (voir <_AMODE> UNITES)		
10	DZ	<_DZ>	REAL	Passe Z (voir <_AMODE> UNITES)		
11	UX	<_UX>	REAL	Surépaisseur de finition X (voir <_VARI> DIZAINES DE MILLIERS)		
12	UZ	<_UZ>	REAL	Surépaisseur de finition Z (voir <_VARI> DIZAINES DE MILLIERS)		
13	U	<_U>	REAL	Surépaisseur de finition parallèle au contour (voir <_VARI> DIZAINES DE MILLIERS)		
14	U1	<_U1>	REAL	Surépaisseur supplémentaire lors de la finition (voir <_AMODE> MILLIERS)		
15	BL	<_BL>	INT	Définition de la pièce brute	1 =	Cylindre avec surépaisseur
					2 =	Surépaisseur sur le contour de la pièce finie
					3 =	Le contour de la pièce brute est indiqué
16	XD	<_XD>	REAL	Définition de la pièce brute X (voir <_AMODE> CENTAINES DE MILLIERS)		
17	ZD	<_ZD>	REAL	Définition de la pièce brute Z (voir <_AMODE> MILLIONS)		
18	XA	<_XA>	REAL	Limite 1 X (abs, toujours diamètre)		
19	ZA	<_ZA>	REAL	Limite 1 Z (abs)		
20	XB	<_XB>	REAL	Limite 2 X (voir <_AMODE> DIZAINES DE MILLIONS)		
21	ZB	<_ZB>	REAL	Limite 2 Z (voir <_AMODE> CENTAINES DE MILLIONS)		
22	XDA	<_XDA>	REAL	Limite de plongée 1 pour la 1re position de gorge sur face frontale (abs, toujours diamètre)		
23	XDB	<_XDB>	REAL	Limite de plongée 2 pour la 1re position de gorge sur face frontale (abs, toujours diamètre)		
24	N	<_N>	INT	Nombre de gorges		
25	DP	<_DP>	REAL	Distance entre gorges	Gorge longitudinale : parallèle à l'axe Z	
					Gorge transversale : parallèle à l'axe X	
26	DI	<_DI>	REAL	Distance pour interruption de l'avance	0 =	Sans interruption
					> 0 =	Avec interruption
27	SC	<_SC>	REAL	Distance de sécurité pour le contournement d'obstacles, en relatif		
28	D2	<_DN>	INT	Numéro D pour 2e tranchant, si non programmé ⇒ D+1		

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
29		<_GMODE>	INT	Mode géométrie (évaluation des valeurs géométriques programmées)		
				UNITES :	Réservé	
				DIZAINES :	Réservé	
				CENTAINES :	Sélection de l'opération / calcul du point de départ uniquement	
					0 =	Usinage normal (aucun mode de compatibilité n'est requis)
					1 =	Usinage normal
					2 =	Calculer la position de départ - aucun usinage (uniquement pour l'appel à partir de ShopMill/ShopTurn)
				MILLIERS :	Limitation	
					0 =	Non
					1 =	Oui
				DIZAINES DE MILLIERS :	Saisir la limite 1 X	
					0 =	Non
					1 =	Oui
				CENTAINES DE MILLIERS :	Saisir la limite 2 X	
					0 =	Non
					1 =	Oui
				MILLIONS :	Saisir la limite 1 Z	
					0 =	Non
					1 =	Oui
				DIZAINES DE MILLIONS :	Saisir la limite 2 Z	
					0 =	Non
					1 =	Oui

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
30		<_DMODE>	INT	Mode d'affichage		
				UNITES :	Plan d'usinage G17/G18/G19	
					0 =	Compatibilité, le plan actif avant l'appel de cycle reste actif
					1 =	G17 (actif uniquement dans le cycle)
					2 =	G18 (actif uniquement dans le cycle)
					3 =	G19 (actif uniquement dans le cycle)
				DIZAINES :	Mode technologie	
					1 =	Chariotage du contour
					2 =	Plongée de contour
					3 =	Plongée G+D
				CENTAINES :	Semi-finition	
					0 =	Non
					1 =	Oui
				MILLIERS :	---	Réservé
				DIZAINES DE MILLIERS :	Mise à l'échelle de la technologie dans les masques de cycle (Page 1193)	
					0 =	Saisie : complète
					1 =	Saisie : simplifiée
				CENTAINES DE MILLIERS :	Nom du programme automatique	
					0 =	non
					1 =	Oui

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
31		<_AMODE>	INT	Mode alternatif		
				UNITES :	Sélection de passe	
					0 =	Passes DX et DZ pour mode de chariotage parallèle au contour
					1 =	Passe D
				DIZAINES :	Stratégie de passe	
					0 =	Profondeur de passe variable (90 à 100 %)
					1 =	Profondeur de passe constante
				CENTAINES :	Plan de passes	
					0 =	Uniforme
					1 =	Aligner sur les arêtes
				MILLIERS :	Sélection de surépaisseur sur contour U1, double finition	
					0 =	Non
					1 =	Oui
				DIZAINES DE MILLIERS :	Actualiser la sélection de la pièce brute	
					0 =	Non
					1 =	Oui
				CENTAINES DE MILLIERS :	Sélection de la surépaisseur sur la pièce brute XD	
					0 =	En absolu, valeur de l'axe transversal en diamètre
					1 =	En relatif, valeur de l'axe transversal en rayon
				MILLIONS :	Sélection de la surépaisseur sur la pièce brute ZD	
					0 =	Absolu
					1 =	Relatif
				DIZAINES DE MILLIONS :	Sélection de la limite 2 XB	
					0 =	En absolu, valeur de l'axe transversal en diamètre
					1 =	En relatif, valeur de l'axe transversal en rayon
				CENTAINES DE MILLIONS :	Sélection de la limite 2 ZB	
					0 =	Absolu
					1 =	Relatif
				MILLIARDS :		
					0 =	Canal pilote
					1 =	Canal asservi
32		<_PK>	INT	Numéro du canal partenaire lorsque plus de 2 canaux sont disponibles sur la machine.		
33	DCH	<_DCH>	REAL	Décalage de canal		
34	FS	<_FS>	REAL	Avance finition pour usinage complet		

3.26.1.46 CYCLE4071 - Rectification longitudinale avec pénétration au point d'inversion

Syntaxe

```
CYCLE4071(<S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>, <S_H>, <S_A1>,
<S_A2>)
```

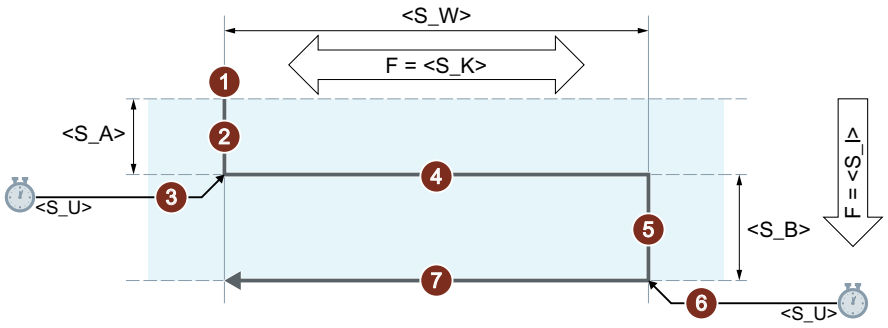
Paramètres

N°	Paramètre	Type de données	Signification
1	<S_A>	REAL	Profondeur de passe au départ
2	<S_B>	REAL	Profondeur de passe à la fin
3	<S_W>	REAL	Largeur de rectification
4	<S_U>	REAL	Durée des passes à lécher
5	<S_I>	REAL	Avance pour passe
6	<S_K>	REAL	Avance pour passe transversale
7	<S_H>	INT	Nombre de répétitions
8	<S_A1>	AXIS	Axe de pénétration (facultatif) ou 1er axe géométrique
9	<S_A2>	AXIS	Axe d'oscillation (facultatif) ou 2e axe géométrique

Fonction

Le cycle sert à exécuter des passes itératives. Les profondeurs de passe au départ et à la fin peuvent être différentes. Un mouvement tangentiel a lieu entre chaque passe.

Déroulement



① Démarrage du cycle dans la position actuelle de l'axe d'oscillation.
② Déplacement de l'axe de pénétration sur la profondeur de passe au départ <S_A> avec l'avance pour passe <S_I>.
③ Étincelage avec durée des passes à lécher <S_U>.
④ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_W> en tant que course et avec l'avance pour passe transversale <S_K>.
⑤ Déplacement de l'axe de pénétration sur la profondeur de passe à la fin <S_B> avec l'avance pour passe <S_I>.
⑥ Étincelage avec durée des passes à lécher <S_U>.
⑦ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_W> en tant que course sur le point de départ et avec l'avance pour passe transversale <S_K>.

Indique les étapes itératives du déroulement.
Le déroulement est répété jusqu'à ce que le nombre de répétitions programmé <S_H> soit atteint.

Remarque

Le déroulement ne peut pas être interrompu en bloc par bloc.

Exemple

Exécuter deux mouvements pendulaires avec les paramètres de cycle suivants :

- Profondeur de passe au départ : 0,02 mm
- Profondeur de passe à la fin : 0,01 mm
- Course : 100 mm
- Durée des passes à lécher : 1 s
- Avance pour passe : 1 mm/min
- Avance transversale : 1000 mm/min
- Répétitions : 2
- Axe d'oscillation et de pénétration : Axes géométriques standard

Code de programme

```
N10 T1 D1
```

Code de programme

```
N20 CYCLE4071(0.02,0.01,100,1,1,1000,2)
N30 M30
```

3.26.1.47 CYCLE4072 - Rectification longitudinale avec pénétration au point d'inversion et signal d'abandon

Syntaxe

```
CYCLE4072(<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>,
<S_H>, <S_A1>, <S_A2>)
```

Paramètres

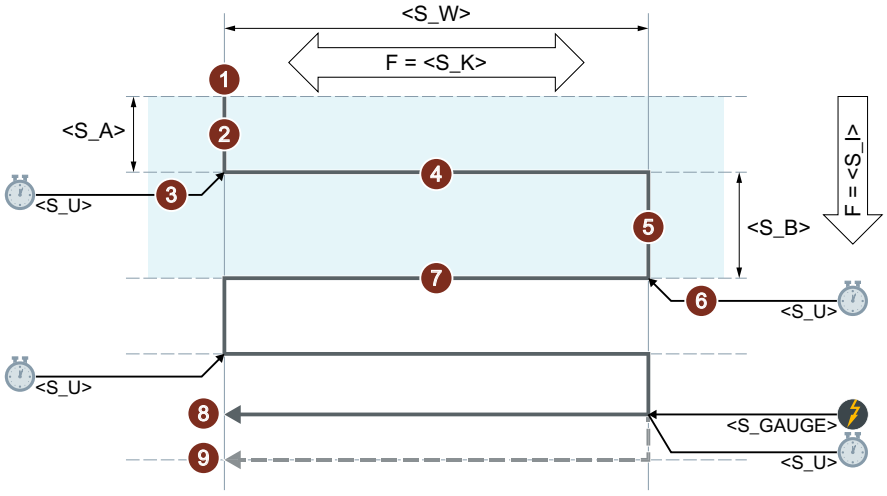
N°	Paramètre	Type de données	Signification
1	<S_GAUGE>	STRING	Conditions d'abandon pour la passe : 1. N° d'une entrée rapide 2. Expression logique
2	<S_A>	REAL	Profondeur de passe au départ
3	<S_B>	REAL	Profondeur de passe à la fin
4	<S_W>	REAL	Largeur de rectification
5	<S_U>	REAL	Durée des passes à lécher
6	<S_I>	REAL	Avance pour passe
7	<S_K>	REAL	Avance pour passe transversale
8	<S_H>	INT	Nombre de répétitions
9	<S_A1>	AXIS	Axe de pénétration (facultatif) ou 1er axe géométrique
10	<S_A2>	AXIS	Axe d'oscillation (facultatif) ou 2e axe géométrique

Fonction

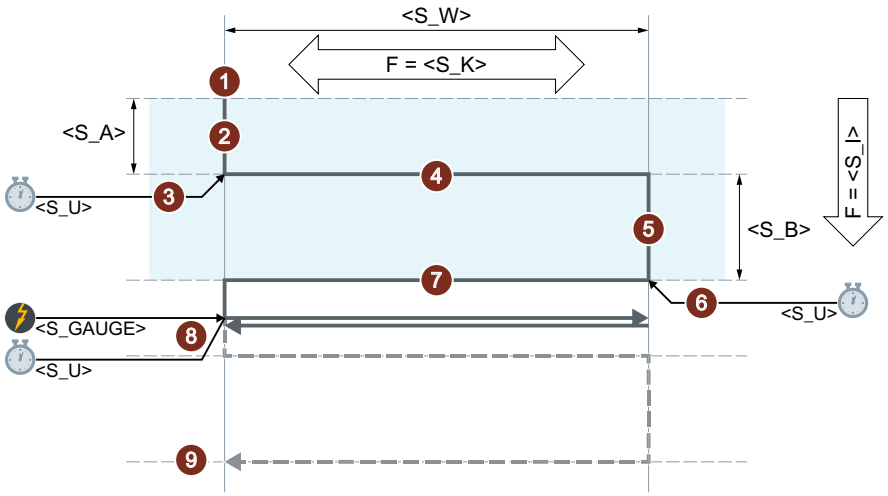
Le cycle sert à exécuter des passes itératives en tenant compte d'un signal d'abandon externe. Les profondeurs de passe au départ et à la fin peuvent être différentes. Un mouvement tangentiel a lieu entre chaque passe. La pénétration est annulée lorsque la condition d'abandon est remplie. Une course complète est toujours effectuée après l'abandon de la pénétration.

Déroulement

Abandon de la passe à la fin



Abandon de la passe au départ



① Démarrage du cycle dans la position actuelle de l'axe d'oscillation.
② Déplacement de l'axe de pénétration sur la profondeur de passe au départ <S_A> avec l'avance pour passe <S_I>.
③ Étincelage avec durée des passes à lécher <S_U>.
④ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_W> en tant que course et avec l'avance pour passe transversale <S_K>.
⑤ Déplacement de l'axe de pénétration sur la profondeur de passe à la fin <S_B> avec l'avance pour passe <S_I>.
⑥ Étincelage avec durée des passes à lécher <S_U>.
⑦ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_W> en tant que course sur le point de départ et avec l'avance pour passe transversale <S_K>.
⑧ Signal d'abandon : l'usinage se termine au prochain point de départ.
⑨ Sans signal d'abandon : Le déroulement est répété jusqu'à ce que le nombre de répétitions programmé <S_H> soit atteint.
Indique les étapes itératives du déroulement.

Remarque

Le déroulement ne peut pas être interrompu en bloc par bloc.

Ressources

Le cycle utilise comme ressources une action synchrone pour l'ensemble des blocs et une variable à action synchrone. L'action synchrone est déterminée de manière dynamique à partir de la plage libre de la bande d'actions synchrones (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). SYG_IS[1] est utilisé comme variable à action synchrone.

Exemples

Exemple 1 : Oscillation avec deux courses

Paramètres de cycle :

- Profondeur de passe au départ : 0,02 mm
- Profondeur de passe à la fin : 0,01 mm
- Course : 100 mm
- Durée des passes à lécher : 1 s
- Avance pour passe : 1 mm/min
- Avance transversale : 1000 mm/min
- Répétitions : 2
- Axe d'oscillation et de pénétration : axes géométriques standard

Signal d'abandon : entrée rapide 1 ($A_IN[1])

Code de programme

```
N10 T1 D1
N20 CYCLE4072("1",0.02,0.01,100,1,1,1000,2)
N30 M30
```

Exemple 2 : Oscillation avec deux courses

Paramètres de cycle :

- Profondeur de passe au départ : 0,02 mm
- Profondeur de passe à la fin : 0,01 mm
- Course : 100 mm
- Durée des passes à lécher : 1 s
- Avance pour passe : 1 mm/min
- Avance transversale : 1000 mm/min
- Répétitions : 2
- Axe d'oscillation et de pénétration : axes géométriques standard

Signal d'abandon : variable $A_DBR[20] < 0,01

Code de programme

```
N10 T1 D1
N20 CYCLE4072("($A_DBR[20]<0.01)",0.02,0.01,100,1,1,1000,2)
N30 M30
```

3.26.1.48 CYCLE4073 - Rectification longitudinale avec pénétration continue

Syntaxe

```
CYCLE4073(<S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>, <S_A1>, <S_A2>)
```

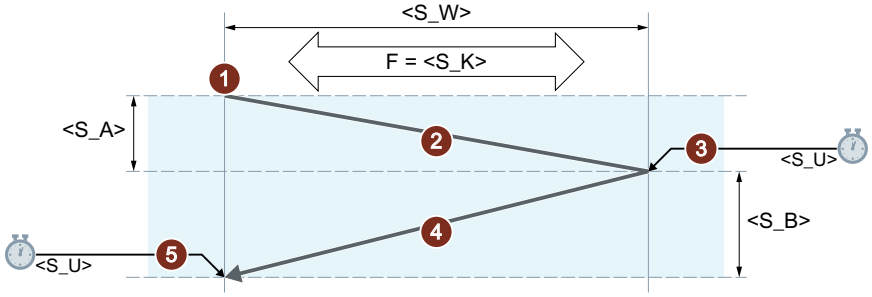
Paramètres

N°	Paramètre	Type de données	Signification
1	<S_A>	REAL	Profondeur de passe au départ
2	<S_B>	REAL	Profondeur de passe à la fin
3	<S_W>	REAL	Largeur de rectification
4	<S_U>	REAL	Durée des passes à lécher
5	<S_K>	REAL	Avance pour passe transversale
6	<S_H>	INT	Nombre de répétitions
7	<S_A1>	AXIS	Axe de pénétration (facultatif) ou 1er axe géométrique
8	<S_A2>	AXIS	Axe d'oscillation (facultatif) ou 2e axe géométrique

Fonction

Le cycle sert à exécuter des passes itératives. La passe du départ vers la fin peut être différente de la passe de la fin vers le départ.

Déroulement



① Démarrage du cycle dans la position actuelle de l'axe d'oscillation avec la profondeur de passe 0.

② Déplacement de l'axe d'oscillation avec la largeur de rectification <S_W> en tant que course et avec l'avance pour passe transversale <S_K> avec augmentation continue de la profondeur de passe jusqu'à la profondeur de passe au départ <S_A>.

③ Étincelage avec durée des passes à lécher <S_U>.

④ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_W> en tant que course sur le point de départ et avec l'avance pour passe transversale <S_K> avec augmentation continue de la profondeur de passe jusqu'à la profondeur de passe à la fin <S_B>.

⑤ Étincelage avec durée des passes à lécher <S_U>.

Indique les étapes itératives du déroulement.

Le déroulement est répété jusqu'à ce que le nombre de répétitions programmé <S_H> soit atteint.

Remarque

Le déroulement ne peut pas être interrompu en bloc par bloc.

Exemple

Oscillation avec deux courses

Paramètres de cycle :

- Profondeur de passe au départ : 0,02 mm
- Profondeur de passe à la fin : 0,01 mm
- Course : 100 mm
- Durée des passes à lécher : 1 s
- Avance transversale : 1000 mm/min
- Répétitions : 2
- Axe d'oscillation et de pénétration : Axes géométriques standard

Code de programme

```
N10 T1 D1
N20 CYCLE4073(0.02,0.01,100,1,1000,2)
N30 M30
```

3.26.1.49 CYCLE4074 - Rectification longitudinale avec pénétration continue et signal d'abandon

Syntaxe

```
CYCLE4074(<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>,
<S_A1>, <S_A2>)
```

Paramètres

N°	Paramètre	Type de données	Signification
1	<S_GAUGE>	STRING	Conditions d'abandon pour la passe : 1. N° d'une entrée rapide 2. Expression logique
2	<S_A>	REAL	Profondeur de passe au départ
3	<S_B>	REAL	Profondeur de passe à la fin
4	<S_W>	REAL	Largeur de rectification
5	<S_U>	REAL	Durée des passes à lécher
6	<S_K>	REAL	Avance pour passe transversale

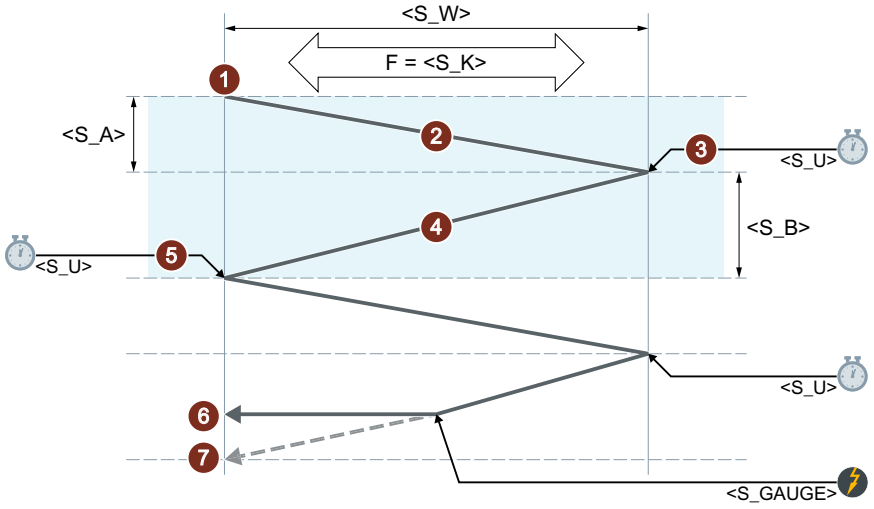
N°	Paramètre	Type de données	Signification
7	<S_H>	INT	Nombre de répétitions
8	<S_A1>	AXIS	Axe de pénétration (facultatif) ou 1er axe géométrique
9	<S_A2>	AXIS	Axe d'oscillation (facultatif) ou 2e axe géométrique

Fonction

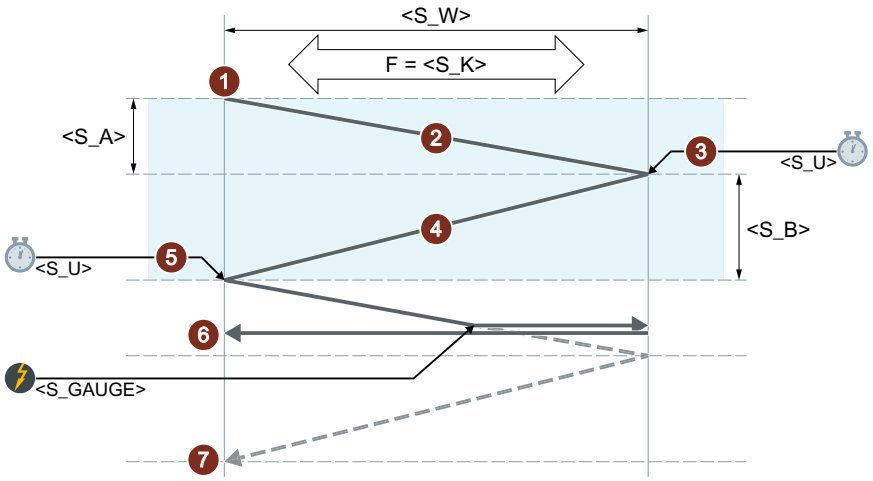
Le cycle sert à exécuter des passes itératives en tenant compte d'un signal d'abandon externe, par exemple. Les profondeurs de passe au départ et à la fin peuvent être différentes. La pénétration est annulée lorsque la condition d'abandon est remplie. Une course complète est toujours effectuée après l'abandon de la pénétration.

Déroulement

Abandon de la passe de la fin vers le départ



Abandon de la passe du départ vers la fin



① Démarrage du cycle dans la position actuelle de l'axe d'oscillation avec la profondeur de passe 0.

② Déplacement de l'axe d'oscillation avec la largeur de rectification <S_W> en tant que course et avec l'avance pour passe transversale <S_K> avec augmentation continue de la profondeur de passe jusqu'à la profondeur de passe au départ <S_A>.

③ Étincelage avec durée des passes à lécher <S_U>.

④ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_W> en tant que course sur le point de départ et avec l'avance pour passe transversale <S_K> avec augmentation continue de la profondeur de passe jusqu'à la profondeur de passe à la fin <S_B>.

⑤ Étincelage avec durée des passes à lécher <S_U>.

⑥ Signal d'abandon : la pénétration est annulée. l'usinage se termine au prochain point de départ.

⑦ Sans signal d'abandon : Le déroulement est répété jusqu'à ce que le nombre de répétitions programmé <S_H> soit atteint.

Indique les étapes itératives du déroulement.

Remarque

Le déroulement ne peut pas être interrompu en bloc par bloc.

Ressources

Le cycle utilise comme ressources une action synchrone pour l'ensemble des blocs et une variable à action synchrone. L'action synchrone est déterminée de manière dynamique à partir de la plage libre de la bande d'actions synchrones (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). SYG_IS[1] est utilisé comme variable à action synchrone.

Exemples

Exemple 1 : Oscillation avec deux courses

Paramètres de cycle :

- Profondeur de passe au départ : 0,02 mm
- Profondeur de passe à la fin : 0,01 mm
- Course : 100 mm
- Durée des passes à lécher : 1 s
- Avance transversale : 1000 mm/min
- Répétitions : 2
- Axe d'oscillation et de pénétration : axes géométriques standard

Signal d'abandon : entrée rapide 1 ($A_IN[1])

Code de programme

```
N10 T1 D1
N20 CYCLE4074("1",0.02,0.01,100,1,1000,2)
N30 M30
```

Exemple 2 : Oscillation avec deux courses

Paramètres de cycle :

- Profondeur de passe au départ : 0,02 mm
- Profondeur de passe à la fin : 0,01 mm
- Course : 100 mm
- Durée des passes à lécher : 1 s
- Avance transversale : 1000 mm/min
- Répétitions : 2
- Axe d'oscillation et de pénétration : axes géométriques standard

Signal d'abandon : Variable $A_DBR[20] < 0,01

Code de programme

```
N10 T1 D1
N20 CYCLE4074("($A_DBR[20]<0.01)",0.02,0.01,100,1,1000,2)
N30 M30
```

3.26.1.50 CYCLE4075 - Rectification plane avec pénétration au point d'inversion

Syntaxe

```
CYCLE4075(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>,
<S_A2>)
```

Paramètres

N°	Paramètre	Type de données	Signification
1	<S_I>	REAL	Profondeur de passe au départ
2	<S_J>	REAL	Profondeur de passe à la fin
3	<S_K>	REAL	Profondeur de passe totale
4	<S_A>	REAL	Largeur de rectification
5	<S_R>	REAL	Avance pour passe
6	<S_F>	REAL	Avance pour passe transversale
7	<S_P>	REAL	Durée des passes à lécher
8	<S_A1>	AXIS	Axe de pénétration (facultatif)
9	<S_A2>	AXIS	Axe d'oscillation (facultatif)

Fonction

Le cycle sert à l'usinage avec une profondeur de passe totale par étapes de pénétration. Les profondeurs de passe au départ et à la fin peuvent être différentes. Un mouvement tangentiel a lieu entre chaque passe.

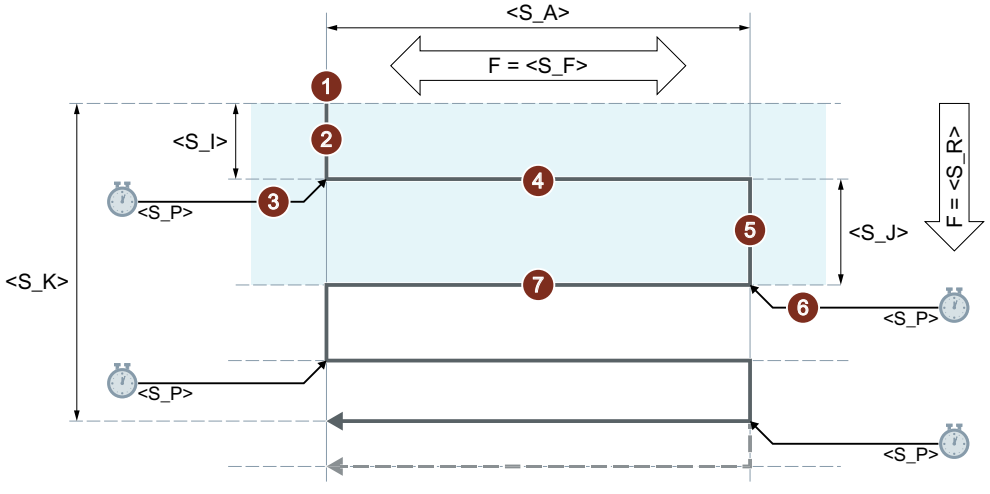
Les indications de course P1 à P4 peuvent être négatives ou positives.

L'indication de l'axe de pénétration et/ou de l'axe d'oscillation est facultative. Si l'un ou aucun des paramètres n'est indiqué, le cycle utilise les deux premiers axes géométriques du canal.

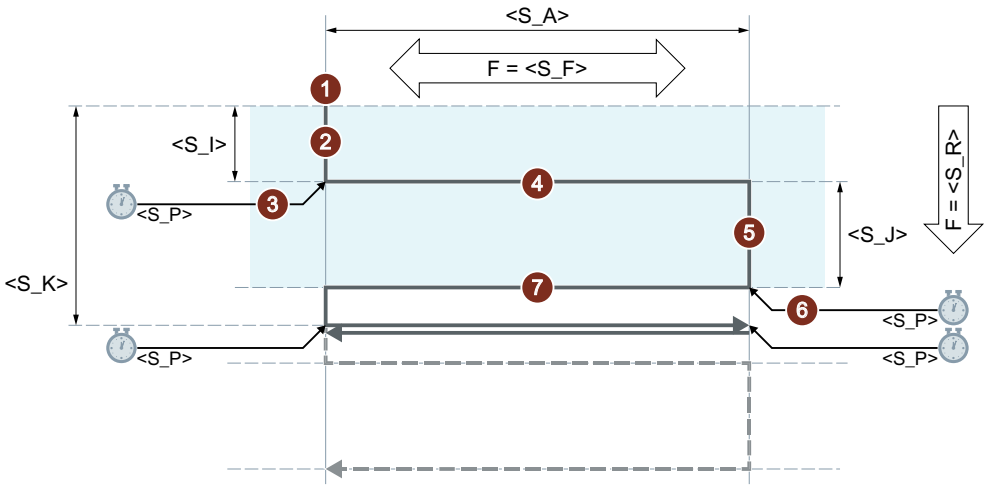
Lorsque la somme des profondeurs de passe au départ et à la fin est égale à 0 ou que la profondeur de passe totale est égale à 0, une seule course d'étincelage est exécutée.

Déroulement

Profondeur de passe totale atteinte lors de la pénétration au deuxième point d'inversion



Profondeur de passe totale atteinte lors de la pénétration au premier point d'inversion



① Démarrage du cycle dans la position actuelle de l'axe d'oscillation.

② Déplacement de l'axe de pénétration sur la profondeur de passe au départ <S_I> avec l'avance pour passe <S_R>.

③ Étincelage avec durée des passes à lécher <S_P>.

④ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_A> en tant que course et avec l'avance pour passe transversale <S_F>.

⑤ Déplacement de l'axe de pénétration sur la profondeur de passe à la fin <S_J> avec l'avance pour passe <S_R>.

⑥ Étincelage avec durée des passes à lécher <S_P>.

⑦ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_A> en tant que course sur le point de départ et avec l'avance pour passe transversale <S_F>.

Indique les étapes itératives du déroulement.

Le déroulement est répété jusqu'à ce que la profondeur de passe totale <S_K> soit atteinte. La dernière course est donc décomposée de manière irrégulière.

Remarque

Le déroulement ne peut pas être interrompu en bloc par bloc.

Exemple

Oscillation avec :

- Profondeur de passe de 0,02 mm au départ
- Profondeur de passe de 0,01 mm à la fin
- Profondeur de passe totale d'1 mm
- Course de 100 mm
- Avance de passe d'1 mm/min
- Avance transversale de 1000 mm/min

- Durée des passes à lécher d'1 seconde
- Axes géométriques standard

Code de programme

```
N10 T1 D1
N20 CYCLE4075(0.02,0.01,1,100,1,1000,1)
N30 M30
```

3.26.1.51 CYCLE4077 - Rectification plane avec pénétration au point d'inversion et signal d'abandon

Syntaxe

```
CYCLE4077(<S_GAUGE>, <S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>,
<S_P>, <S_A1>, <S_A2>)
```

Paramètres

N°	Paramètre	Type de données	Signification
1	<S_GAUGE>	STRING	Condition d'abandon pour la passe : • N° d'une entrée rapide • Expression logique
2	<S_I>	REAL	Profondeur de passe au départ
3	<S_J>	REAL	Profondeur de passe à la fin
4	<S_K>	REAL	Profondeur de passe totale
5	<S_A>	REAL	Largeur de rectification
6	<S_R>	REAL	Avance pour passe
7	<S_F>	REAL	Avance pour passe transversale
8	<S_P>	REAL	Durée des passes à lécher
9	<S_A1>	AXIS	Axe de pénétration (facultatif)
10	<S_A2>	AXIS	Axe d'oscillation (facultatif)

Fonction

Le cycle sert à l'usinage avec une profondeur de passe totale par étapes de pénétration. Les profondeurs de passe au départ et à la fin peuvent être différentes. Un mouvement tangentiel a lieu entre chaque passe. La pénétration est annulée lorsque le signal d'abandon de l'entrée rapide est égal à 1 ou que la condition d'abandon est remplie. Une course complète est effectuée après l'abandon.

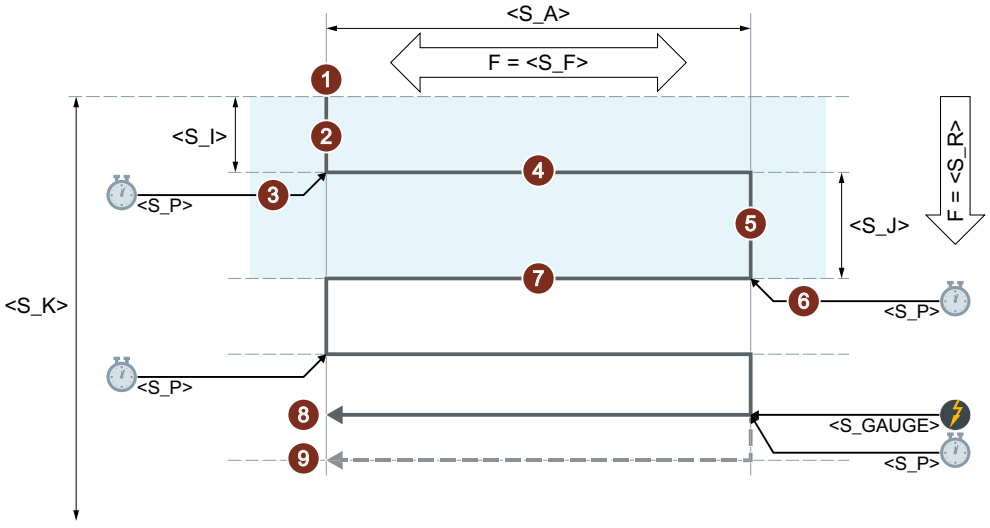
Les indications de course P2 à P5 peuvent être négatives ou positives.

L'indication de l'axe de pénétration et/ou de l'axe d'oscillation est facultative. Si l'un ou aucun des paramètres n'est indiqué, le cycle utilise les deux premiers axes géométriques du canal.

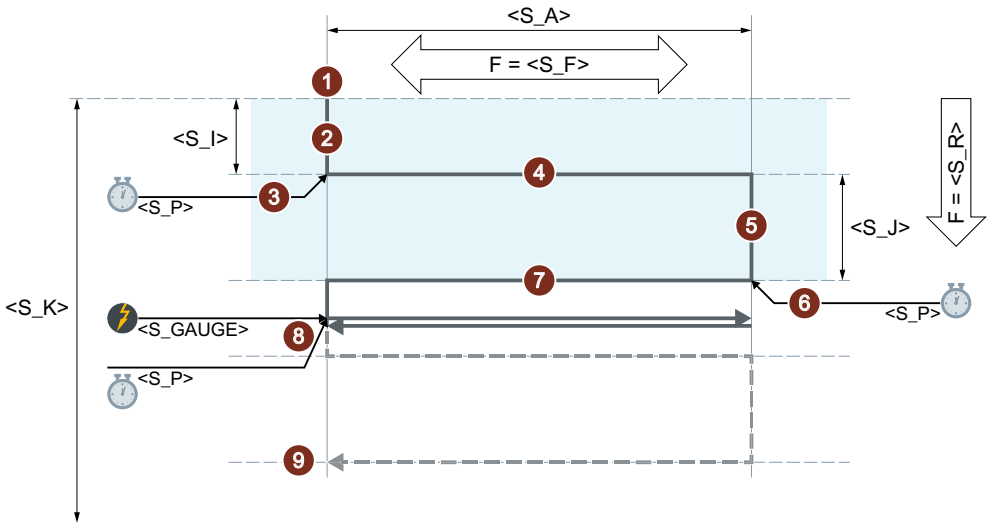
Lorsque la somme des profondeurs de passe au départ et à la fin est égale à 0 ou que la profondeur de passe totale est égale à 0, une seule course d'étincelage est exécutée.

Déroulement

Abandon de la passe à la fin



Abandon de la passe au départ



① Démarrage du cycle dans la position actuelle de l'axe d'oscillation.
② Déplacement de l'axe de pénétration sur la profondeur de passe au départ <S_I> avec l'avance pour passe <S_R>.
③ Étincelage avec durée des passes à lécher <S_P>.
④ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_A> en tant que course et avec l'avance pour passe transversale <S_F>.
⑤ Déplacement de l'axe de pénétration sur la profondeur de passe à la fin <S_J> avec l'avance pour passe <S_R>.
⑥ Étincelage avec durée des passes à lécher <S_P>.
⑦ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_A> en tant que course sur le point de départ et avec l'avance pour passe transversale <S_F>.
⑧ Signal d'abandon : l'usinage se termine au prochain point de départ.
⑨ Sans signal d'abandon : Le déroulement est répété jusqu'à ce que la profondeur de passe totale <S_K> soit atteinte. La dernière course est donc décomposée de manière irrégulière.

Indique les étapes itératives du déroulement.

Remarque

Le déroulement ne peut pas être interrompu en bloc par bloc.

Ressources

Le cycle utilise comme ressources une action synchrone pour l'ensemble des blocs et une variable à action synchrone. L'action synchrone est déterminée de manière dynamique à partir de la plage libre de la bande d'actions synchrones (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). SYG_IS[1] est utilisé comme variable à action synchrone.

Exemples

Exemple 1

Oscillation avec :

- Profondeur de passe de 0,02 mm au départ
- Profondeur de passe de 0,01 mm à la fin
- Profondeur de passe totale d'1 mm
- Course de 100 mm
- Avance de passe d'1 mm/min
- Avance transversale de 1000 mm/min
- Durée des passes à lécher d'1 seconde
- Axes géométriques standard

Signal d'abandon : entrée rapide 1 ($A_IN[1])

Code de programme

```
N10 T1 D1
N20 CYCLE4077("1",0.02,0.01,1,100,1,1000,1)
N30 M30
```

Exemple 2

Oscillation avec :

- Profondeur de passe de 0,02 mm au départ
- Profondeur de passe de 0,01 mm à la fin
- Profondeur de passe totale d'1 mm
- Course de 100 mm
- Avance de passe d'1 mm/min
- Avance transversale de 1000 mm/min
- Durée des passes à lécher d'1 seconde
- Axes géométriques standard

Signal d'abandon : variable RAM 20 à double port inférieure à 0,01 ($A_DBR[20] < 0,01)

Code de programme

```
N10 T1 D1
N20 CYCLE4077("($A_DBR[20]<0.01)",0.02,0.01,1,100,1,1000,1)
N30 M30
```

3.26.1.52 CYCLE4078 - Rectification plane avec pénétration continue

Syntaxe

```
CYCLE4078(<S_I>, <S_J>, <S_K>, <S_A>, <S_F>, <S_P>, <S_A1>, <S_A2>)
```

Paramètres

N°	Paramètre	Type de données	Signification
1	<S_I>	REAL	Profondeur de passe du départ vers la fin
2	<S_J>	REAL	Profondeur de passe de la fin vers le départ
3	<S_K>	REAL	Profondeur de passe totale
4	<S_A>	REAL	Largeur de rectification
5	<S_F>	REAL	Avance
6	<S_P>	REAL	Durée des passes à lécher
7	<S_A1>	AXIS	Axe de pénétration (facultatif)
8	<S_A2>	AXIS	Axe d'oscillation (facultatif)

Fonction

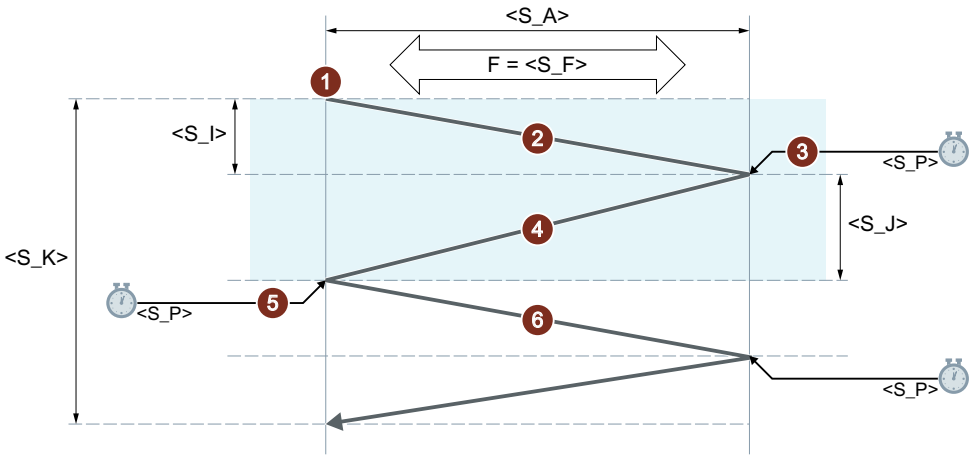
Le cycle sert à l'usinage avec une profondeur de passe totale par pénétration continue. Les profondeurs de passe du départ vers la fin et de la fin vers le départ peuvent être différentes.

Les indications de course P1 à P4 peuvent être négatives ou positives.

L'indication de l'axe de pénétration et/ou de l'axe d'oscillation est facultative. Si l'un ou aucun des paramètres n'est indiqué, le cycle utilise les deux premiers axes géométriques du canal.

Lorsque la somme des profondeurs de passe P1 et P2 est égale à 0 ou que la profondeur de passe totale est égale à 0, une seule course d'étincelage est exécutée.

Déroulement



① Démarrage du cycle dans la position actuelle de l'axe d'oscillation avec la profondeur de passe 0.

② Déplacement de l'axe d'oscillation avec la largeur de rectification <S_A> en tant que course et avec l'avance <S_F> avec augmentation continue de la profondeur de passe jusqu'à la profondeur de passe au départ <S_I>.

③ Étincelage avec durée des passes à lécher <S_P>.

④ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_A> en tant que course sur le point de départ et avec l'avance <S_F> avec augmentation continue de la profondeur de passe jusqu'à la profondeur de passe à la fin <S_J>.

⑤ Étincelage avec durée des passes à lécher <S_P>.

⑥ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_A> en tant que course sur le point de départ et avec l'avance <S_F>.

Indique les étapes itératives du déroulement.

Le déroulement est répété jusqu'à ce que la profondeur de passe totale <S_K> soit atteinte. La dernière course est donc décomposée de manière irrégulière.

Remarque

Le déroulement ne peut pas être interrompu en bloc par bloc.

Exemple

Oscillation avec :

- Profondeur de passe de 20 mm au départ
- Profondeur de passe de 10 mm à la fin
- Profondeur de passe totale de 100 mm
- Course de 100 mm
- Avance 1000 mm/min
- Durée des passes à lécher d'1 seconde
- Axes géométriques standard

Code de programme

```
N10 T1 D1
N20 CYCLE4078(20,10,100,100,1000,1)
N30 M30
```

3.26.1.53 CYCLE4079 - Rectification plane avec pénétration intermittente

Syntaxe

```
CYCLE4079(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>,
<S_A2>)
```

Paramètres

N°	Paramètre	Type de données	Signification
1	<S_I>	REAL	Profondeur de passe au départ
2	<S_J>	REAL	Profondeur de passe à la fin
3	<S_K>	REAL	Profondeur de passe totale
4	<S_A>	REAL	Largeur de rectification
5	<S_R>	REAL	Avance pour passe
6	<S_F>	REAL	Avance pour passe transversale
7	<S_P>	REAL	Durée des passes à lécher
8	<S_A1>	AXIS	Axe de pénétration (facultatif)
9	<S_A2>	AXIS	Axe d'oscillation (facultatif)

Fonction

Le cycle sert à l'usinage avec une profondeur de passe totale par étapes de pénétration. Les profondeurs de passe au départ et à la fin peuvent être différentes. Un mouvement tangentiel a lieu entre chaque passe.

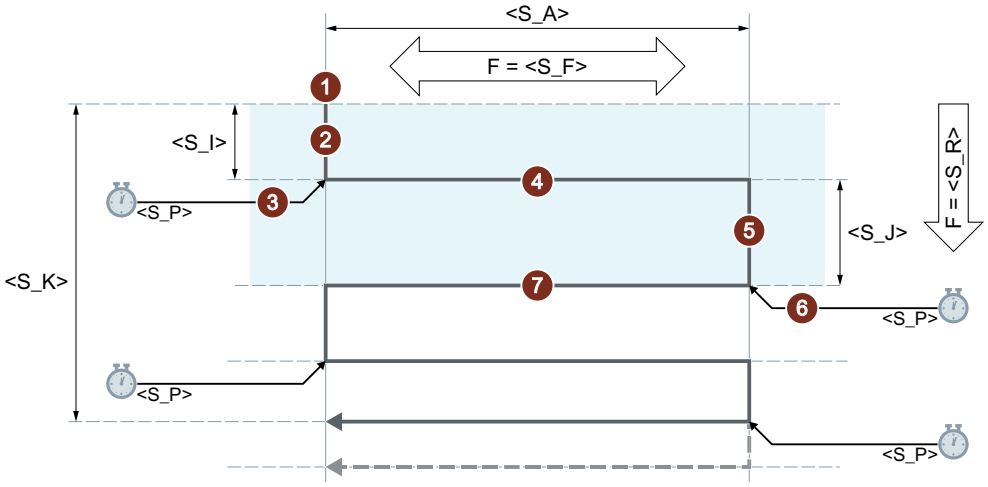
Les indications de course P1 à P4 peuvent être négatives ou positives.

L'indication de l'axe de pénétration et/ou de l'axe d'oscillation est facultative. Si l'un ou aucun des paramètres n'est indiqué, le cycle utilise les deux premiers axes géométriques du canal.

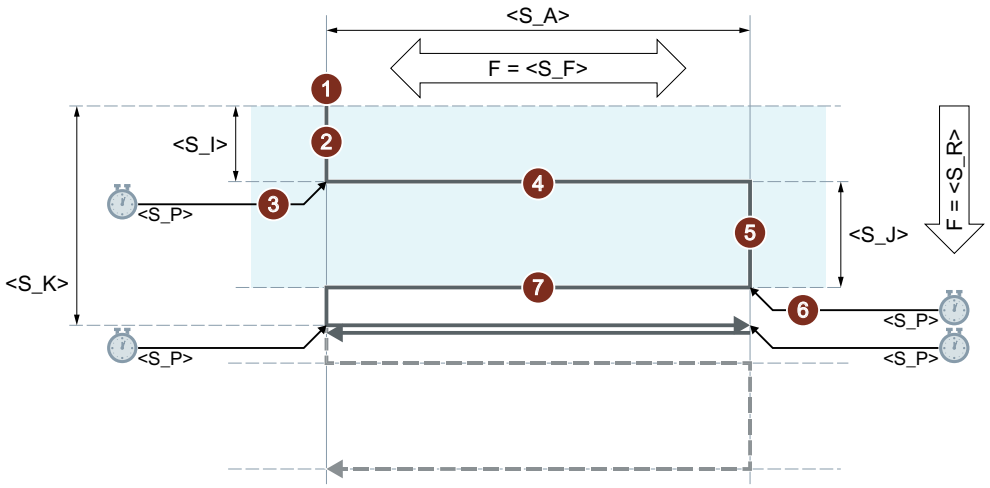
Lorsque la somme des profondeurs de passe au départ et à la fin est égale à 0 ou que la profondeur de passe totale est égale à 0, une seule course d'étincelage est exécutée.

Déroulement

Profondeur de passe totale atteinte lors de la pénétration au deuxième point d'inversion



Profondeur de passe totale atteinte lors de la pénétration au premier point d'inversion



① Démarrage du cycle dans la position actuelle de l'axe d'oscillation.

② Déplacement de l'axe de pénétration sur la profondeur de passe au départ <S_I> avec l'avance pour passe <S_R>.

③ Étincelage avec durée des passes à lécher <S_P>.

④ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_A> en tant que course et avec l'avance pour passe transversale <S_F>.

⑤ Déplacement de l'axe de pénétration sur la profondeur de passe à la fin <S_J> avec l'avance pour passe <S_R>.

⑥ Étincelage avec durée des passes à lécher <S_P>.

⑦ Déplacement de l'axe d'oscillation avec la largeur de rectification <S_A> en tant que course sur le point de départ et avec l'avance pour passe transversale <S_F>.

Indique les étapes itératives du déroulement.

Le déroulement est répété jusqu'à ce que la profondeur de passe totale <S_K> soit atteinte. La dernière course est donc décomposée de manière irrégulière.

Remarque

Le déroulement ne peut pas être interrompu en bloc par bloc.

Exemple

Oscillation avec :

- Profondeur de passe de 0,02 mm au départ
- Profondeur de passe de 0,01 mm à la fin
- Profondeur de passe totale d'1 mm
- Course de 100 mm
- Avance de passe d'1 mm/min
- Avance transversale de 1000 mm/min

- Durée des passes à lécher d'1 seconde
- Axes géométriques standard

Code de programme

```
N10 T1 D1
N20 CYCLE4079(0.02,0.01,1,100,1,1000,1)
N30 M30
```

3.26.1.54 GROUP_BEGIN - Début du bloc de programme

Syntaxe

```
GROUP_BEGIN(<_LEVEL>, <_NAME>, <_SP>, <_MODE>, <S_ICON>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification		
1		<_LEVEL>	INT	Plan		
				0 =	Plan principal	
				1 =	1er niveau inférieur	
2		<_NAME>	STRING[128]	Nom de bloc		
3		<_SP>	INT	Broche		
				0 =	Pas de broche	
				1 =	Broche principale	
				2 =	Contre-broche	
4		<_MODE>	INT	Mode		
				Bit 0	= 1	GROUP_ADDEND existe
				Bit 1	= 1	ShopTurn : retrait automatique (accostage de la position de changement d'outil)
				Bit 12	réservé	
				Bit 13	réservé	
5		<S_ICON>	STRING[32]	Nom de l'icône (uniquement pour l'interface utilisateur)		

3.26.1.55 GROUP_END - Fin du bloc de programme

Syntaxe

```
GROUP_END(<_LEVEL>, <_SP>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification	
1		<_LEVEL>	INT	Plan	
				0 =	Plan principal
				1 =	1er niveau inférieur
2		<_SP>	INT	Broche	
				0 =	Pas de broche
				1 =	Broche principale
				2 =	Contre-broche

3.26.1.56 GROUP_ADDEND - Fin du supplément de mise au point

Syntaxe

```
GROUP_ADDEND(<_LEVEL>, <_SP>)
```

Paramètres

N°	Masque de paramètre	Paramètre interne	Type de données	Signification	
1		<_LEVEL>	INT	Plan	
				0 =	Plan principal
				1 =	1er niveau inférieur
2		<_SP>	INT	Broche	
				0 =	Pas de broche
				1 =	Broche principale
				2 =	Contre-broche

3.26.1.57 Conditions marginales

Mise à l'échelle de la technologie dans les masques de cycle

Si la mise à l'échelle de la technologie est activée, il est possible de sélectionner pour différents masques de cycle la saisie simplifiée, pour laquelle seuls les paramètres de cycle les plus importants sont affichés.

La saisie simplifiée peut être sélectionnée par exemple pour les masques de cycle suivants :

Technologie	Masque de cycle
Perçage	Perçage profond
	Taraudage
Fraisage	Poche rectangulaire
	Fraisage de contour : poche

Technologie	Masque de cycle
Tournage	Filetage au tour : longitudinale
	Tournage de contour : chariotage
	Tournage de contour : plongée
	Tournage de contour : plongée G+D

Pour les masques de cycle concernés, l'interface utilisateur propose les options "Saisie : **simplifiée**" et "Saisie : **complète**".

Paramètres de cycle non affichés

Les paramètres de cycle non affichés pour la saisie simplifiée sont soit préréglés avec des valeurs fixes, technologiquement pertinentes, mais non modifiables, soit ils se voient affecter des valeurs paramétrables par les données de réglage de cycle spécifiques au canal. Voir section "Mise en service" > "Données de réglage de cycle spécifiques au canal" ci-dessous

Basculement "Saisie : complète" > "Saisie : simplifiée"

Si un masque de cycle est renseigné avec le réglage "Saisie complète" et ensuite basculé sur "Saisie simplifiée", lors de la génération de l'appel de cycle, les valeurs par défaut ou les valeurs des données de réglage seront utilisées pour les paramètres qui ne sont plus affichés.

Mise en service

Paramètres machine de configuration spécifiques à un canal

Le paramètre machine permet d'activer l'option de mise à l'échelle de la technologie dans les masques de cycle :

PM52210 $MCS_FUNCTION_MASK_DISP, bit 9 = 1 (afficher sélection "Saisie simplifiée")

Données de réglage de cycle spécifiques à un canal

Si la saisie simplifiée est activée dans les masques de cycle, les valeurs pour des paramètres de cycle déterminés peuvent être spécifiées par les données de réglage suivantes :

Numéro	Descripteur	Signification
SD55300	$SCS_EASY_SAFETY_CLEARANCE	Distance de sécurité
SD55301	$SCS_EASY_DWELL_TIME	Arrêt temporisé
SD55305	$SCS_EASY_DRILL_DEEP_FD1	Perçage profond : Pourcentage : 1. Avance
SD55306	$SCS_EASY_DRILL_DEEP_DF	Perçage profond : pourcentage : pénétration
SD55307	$SCS_EASY_DRILL_DEEP_V1	Perçage profond : profondeur de passe minimale
SD55308	$SCS_EASY_DRILL_DEEP_V2	Perçage profond : valeur de retrait
SD55309	$SCS_EASY_THREAD_RETURN_DIST	Filetage au tour : distance de retrait

3.26.2 Vue d'ensemble des paramètres des cycles de mesure

3.26.2.1 Paramètres des cycles de mesure CYCLE973

```
PROC CYCLE973(INT S_MVAR,INT S_PRNUM,INT S_CALNUM,REAL S_SETV,INT S_MA,INT S_MD,REAL
S_FA,REAL S_TSA,REAL S_VMS,INT S_NMSP,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-8 Paramètres d'appel CYCLE973 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=0012103)	
			Valeurs :	UNITES : calibrage sur une surface, une arête ou dans une rainure 0 = longueur sur surface/arête (dans le SCP) avec valeur de consigne connue 1 = rayon sur surface (dans le SCP) avec valeur de consigne connue 2 = longueur dans rainure (dans le SCM), voir S_CALNUM 3 = rayon dans rainure (dans le SCM), voir S_CALNUM
				DIZAINES : réservé 0 = 0
				CENTAINES : réservé 0 = 0
				MILLIERS : sélection axe de mesure et sens de mesure pour calibrage [2)] 0 = aucune donnée (pour calibrage surface sur fond de rainure, aucun axe de mesure ni direction de mesure sélectionné) [4)] 1 = indiquer sélection axe de mesure et direction de mesure, voir S_MA, S_MD (une direction de mesure dans un axe de mesure) 2 = indiquer sélection axe de mesure, voir S_MA (deux directions de mesure dans un axe de mesure)
				DIZAINES DE MILLIERS : détermination écart de position (inclinaison du palpeur) [2), 3)] 0 = déterminer écart de position 1 = ne pas déterminer écart de position
				CENTAINES DE MILLIERS : réservé 0 = 0
				UN MILLION :Adapter la longueur d'outil [7)] 0 = ne pas adapter la longueur d'outil (uniquement point de déclenchement) 1 = adapter la longueur d'outil
2	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
3		S_CALNUM	Numéro de la rainure de calibrage lors du calibrage sur une rainure (par défaut=1) [5)]	
4		S_SETV	Valeur de consigne lors du calibrage sur une surface	

N°	Paramètre de masque	Paramètre de cycle	Signification	
5	X0	S_MA	Axe de mesure (numéro de l'axe) [6] (par défaut=1)	
			Valeurs :	1 = 1er axe du plan (pour G18 Z) 2 = 2e axe du plan (pour G18 X) 3 = 3e axe du plan (pour G18 Y) [6]
6	+-	S_MD	Direction de mesure (par défaut=1)	
			Valeurs :	0 = direction de mesure positive 1 = direction de mesure négative
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	VMS	S_VMS	Vitesse de mesure variable lors du calibrage [2]	
10	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (par défaut=1)	
11		S_MCBIT	réservé	
12		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
13		_AMODE	Mode alternatif	

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Uniquement pertinent lors du calibrage dans deux directions d'axe

4) Uniquement axe de mesure et direction de mesure sont automatiquement déterminés à partir de la position de tranchant (SL) du palpeur. SL=8 → -X , SL=7 → -Z

5) Le numéro de la rainure de calibrage (n) renvoie aux données de réglage générales suivantes (toutes les positions dans le SCM) :
Pour position de tranchant SL=7 :
SD54615 $SNS_MEA_CAL_EDGE_BASE_AX1[n] Position du fond de la rainure dans le 1er axe du plan (pour G18 Z)
SD54621 $SNS_MEA_CAL_EDGE_PLUS_DIR_AX2[n] Position du flanc de la rainure dans le sens positif du 2e axe du plan (pour G18 X)
SD54622 $SNS_MEA_CAL_EDGE_MINUS_DIR_AX2[n] Position du flanc de la rainure dans le sens négatif du 2e axe du plan
Pour position de tranchant SL=8 :
SD54619 $SNS_MEA_CAL_EDGE_BASE_AX2[n] Position du fond de la rainure dans le 2e axe du plan
SD54620 $SNS_MEA_CAL_EDGE_UPPER_AX2[n] Position du bord supérieur de la rainure dans le 2e axe du plan (uniquement pour prépositionner le palpeur)
SD54617 $SNS_MEA_CAL_EDGE_PLUS_DIR_AX1[n] Position du flanc de la rainure dans le sens positif du 1er axe du plan
SD54618 $SNS_MEA_CAL_EDGE_MINUS_DIR_AX1[n] Position du flanc de la rainure dans le sens négatif du 1er axe du plan
Remarque :
Les valeurs de position du flanc de la rainure +- peuvent être déterminées approximativement.
La largeur de la rainure, résultant de l'écart des valeurs de position du flanc de la rainure, doit être déterminée avec précision (comparateur de précision).
Lors du calibrage dans la rainure, la longueur d'outil du palpeur de l'axe calibré est supposée être égale à 0.
Les valeurs de position du fond de la rainure doivent également être déterminées avec précision sur la machine (pas de cotes tirées d'un plan).

6) Axe de mesure S_MA=3 pour le calibrage sur une surface et sur un tour avec un 3e axe du plan (pour G18 Y).

7) Adapter la longueur d'outil pour calibrage de la longueur dans rainure ou pour des longueurs sur surface.
Le palpeur de pièce dans des tours peut être décrit avec 2 longueurs (X Z).
Palpeur Tournage type 580

Position de tranchant 7 : pour le calibrage de la longueur, la longueur Z est corrigée au choix.
Palpeur Tournage type 580
Longueur de coupe 8 : pour le calibrage de la longueur, la longueur X est corrigée au choix.
Pour les variantes de mesure Rayon sur rainure ou Rayon sur surface, aucune adaptation de la longueur d'outil n'est réalisée. Seuls les points de déclenchement correspondants sont toujours archivés.

3.26.2.2 Paramètres des cycles de mesure CYCLE974

```
PROC CYCLE974(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,INT S_MA,REAL
S_FA,REAL S_TSA,REAL S_STA1,INT S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL S_TZL,REAL
S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT
_AMODE,INT _DP)
```

Tableau 3-9 Paramètres d'appel CYCLE974[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : 0 = mesurer face frontale 1 = mesure intérieure 2 = mesure extérieure
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM)[3)] 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : réservé
				DIZAINES DE MILLIERS : mesure avec ou sans pivotement de la broche principale (broche rotative) 0 = mesure sans pivotement 1 = mesure avec pivotement

N°	Paramètre de masque	Paramètre de cycle	Signification	
2	Sélection	S_KNUM	Correction dans décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction dans DO ou DO de base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2), 4)]	
			Valeurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction entre 1 et au maximum 999 numéro D (numéro de tranchant) pour correction d'outil ; pour correction totale et en réglage, voir aussi S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction dans géométrie d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune indication (correction de l'usure de longueur d'outil) 1 = correction d'outil correction totale (CT)[5)] Valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + CT (ancienne) valeur de correction, CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale (non inversée)) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
5	X0	S_SETV	Valeur de consigne	
6	X	S_MA	Axe de mesure (numéro de l'axe) (par défaut=1)	
			Valeurs :	1 = 1er axe du plan (pour G18 Z) 2 = 2e axe du plan (pour G18 X) 3 = 3e axe du plan (pour G18 Y) [5)]
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	α	S_STA1	Angle de départ lors de la mesure avec pivotement	
10	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (par défaut=1)	
11	T	S_TNAME	Nom d'outil[2)]	
12	DL	S_DLNUM	Correction totale / en réglage numéro DL[5)]	

N°	Paramètre de masque	Paramètre de cycle	Signification	
13	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	
14	TZL	S_TZL	Correction zéro[2), 4)]	
15	DIF	S_TDIF	Contrôle d'écart dimensionnel[2), 4)]	
16	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[4)]	
17	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[4)]	
18	TMV	S_TMV	Plage de correction lors du calcul de moyenne[2)]	
19	FW	S_K	Facteur de pondération pour calcul de moyenne[2)]	
20	EVN	S_EVNUM	Numéro mémoire de moyennes empiriques[2), 7)]	
21		S_MCBIT	réservé	
22		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
23		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Correction dans DO uniquement possible avec mesure sans pivotement

4) Pour correction d'outil, tenir compte du paramètre machine spécifique au canal MD 20360 TOOL_PARAMETER_DEF_MASK bit0 et bit1

5) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR . De plus, il faut que le paramètre machine général MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK bit8=1.

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

7) Calcul de moyenne empirique uniquement possible pour correction d'outil
Plage de valeurs de la mémoire de valeurs moyennes empiriques :
De 1 à 20, numéro de la mémoire de valeurs empiriques, voir donnée de réglage propre au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
De 10 000 à 200 000, numéro de la mémoire de valeurs moyennes, voir donnée de réglage propre au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

3.26.2.3 Paramètres des cycles de mesure CYCLE994

```
PROC CYCLE994(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,INT S_MA,REAL
S_SZA,REAL S_SZO,REAL S_FA,REAL S_TSA,INT S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL
S_TZL,REAL S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT
_DMODE,INT _AMODE,INT _DP)
```

Tableau 3-10 Paramètres d'appel CYCLE994[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : mesure intérieure ou extérieure (par défaut = 1) 1 = mesure intérieure 2 = mesure extérieure
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM)[3)] 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : plage de contournement 0 = aucune plage de contournement 1 = axe de contournement du 1er axe du plan (pour G18 Z). Axe de mesure, voir S_MA. 2 = axe de contournement du 2e axe du plan (pour G18 X). Axe de mesure, voir S_MA. 3 = axe de contournement du 3e axe du plan (pour G18 Y). Axe de mesure, voir S_MA. [8)]
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2), 4)]	
			Va-leurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction De 1 à 999 max. numéro D (numéro de tranchant) pour correction d'outil Pour correction totale et en réglage, voir aussi S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction géométrie d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune indication (correction de l'usure de longueur d'outil) 1 = correction d'outil correction totale (CT)[5)] Valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + CT (ancienne) valeur de correction, CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] Valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie ;
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale, non inversée) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
5	X0	S_SETV	Valeur de consigne	
6	X	S_MA	Numéro de l'axe de mesure (par défaut=1)	
			Va-leurs :	1 = 1er axe du plan (pour G18 Z) 2 = 2e axe du plan (pour G18 X) 3 = 3e axe du plan (pour G18 Y) [8)]
7	X1	S_SZA	Distance de contournement dans l'axe de mesure	
8	Y1	S_SZO	Distance de contournement dans l'axe de contournement	
9	DFA	S_FA	Course de mesure	
10	TSA	S_TSA	Zone de fiabilité	
11	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (par défaut=1)	
12	T	S_TNAME	Nom d'outil[2)]	

N°	Paramètre de masque	Paramètre de cycle	Signification	
13	DL	S_DLNUM	Correction totale / en réglage numéro DL[5]	
14	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	
15	TZL	S_TZL	Correction zéro[2), 4)]	
16	DIF	S_TDIF	Contrôle d'écart dimensionnel[2), 4)]	
17	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[4)]	
18	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[4)]	
19	TMV	S_TMV	Plage de correction lors du calcul de moyenne[2)]	
20	FW	S_K	Facteur de pondération pour calcul de moyenne[2)]	
21	EVN	S_EVNUM	Numéro mémoire de valeurs empiriques[2), 7)]	
22		S_MCBIT	réservé	
23		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
24		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Correction dans DO uniquement possible avec mesure sans pivotement

4) Pour correction d'outil, tenir compte du paramètre machine spécifique au PM de canal 20360 TOOL_PARAMETER_DEF_MASK

5) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR . De plus, il faut que le paramètre machine général MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK bit8=1.

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

7) Calcul de moyenne empirique uniquement possible pour correction d'outil
Plage de valeurs de la mémoire des valeurs moyennes empiriques :
De 1 à 20, numéro de la mémoire des valeurs empiriques, voir donnée de réglage spécifique au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
De 10 000 à 200 000 numéro de la mémoire des valeurs moyennes, voir donnée de réglage spécifique au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

8) Si axe Y disponible sur machine

3.26.2.4 Paramètres des cycles de mesure CYCLE976

```
PROC CYCLE976(INT S_MVAR,INT S_PRNUM,REAL S_SETV,REAL S_SETV0,INT S_MA,INT S_MD,REAL
S_FA,REAL S_TSA,REAL S_VMS,REAL S_STA1,INT S_NMSP,INT S_SETV1,INT _DMODE,INT _AMODE)
```

Tableau 3-11 Paramètres d'appel CYCLE976[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=1000)	
			Valeurs :	UNITES : calibrage sur surface, sur sphère de calibrage ou dans bague de calibrage[2)] 0 = longueur sur surface avec valeur de consigne connue 1 = rayon dans bague de calibrage avec diamètre connu (valeur de consigne) et centre connu. 2 = rayon dans bague de calibrage avec diamètre connu (valeur de consigne) et centre inconnu 3 = rayon et longueur sur sphère de calibrage 4 = rayon sur arête avec valeur de consigne connue Tenir compte de la sélection axe de mesure et direction de mesure. [3)] 5 = Rayon entre deux arêtes avec consigne connue et distance par rapport aux arêtes. L'axe de mesure doit être sélectionné.
				DIZAINES : réservé 0 = 0
				CENTAINES : réservé 0 = 0
				MILLIERS : sélection axe de mesure et direction de mesure lors du calibrage 0 = aucune donnée (aucune axe de mesure ni direction de mesure sélectionné)[8)] 1 = indiquer axe de mesure et direction de mesure sélectionné, voir S_MA, S_MD (une direction de mesure dans un axe de mesure) 2 = indiquer axe de mesure sélectionné, voir S_MA (deux directions de mesure dans un axe de mesure)
				DIZAINES DE MILLIERS : détermination écart de position (inclinaison du palpeur)[2)] 0 = déterminer écart de position du palpeur[6)] 1 = ne pas déterminer d'écart de position
				CENTAINES DE MILLIERS : calibrage en parallèle à l'axe ou sous angle 0 = calibrage en parallèle à l'axe dans le SCP actif 1 = calibrage sous angle[7)]
				MILLIONS : détermination longueur d'outil lors du calibrage sur surface ou sur sphère 0 = aucune détermination de la longueur d'outil 1 = détermination de la longueur d'outil [4)] 2 = calibrage de l'axe d'approche sur la sphère, détermination de la longueur d'outil, saisie de la différence de mesure des longueurs d'outil dans les données de calibrage
2	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
3		S_SETV	Valeur de consigne	
4	Z0	S_SETV0	Valeur de consigne de la référence de longueur lors du calibrage Sphère	
5	X / Y / Z	S_MA	Axe de mesure (numéro de l'axe)[2), 6)] (par défaut=1)	
			Valeurs :	1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z)

N°	Paramètre de masque	Paramètre de cycle	Signification	
6	+ -	S_MD	Direction de mesure[2), 6)]	
			Valeurs :	0 = positif 1 = négatif
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	VMS	S_VMS	Vitesse de mesure variable lors du calibrage [2)]	
10	α	S_STA1	Angle de départ[2), 5)]	
11	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (par défaut=1)	
12	X0	S_SETV1	Point de référence d'arête pour calibrage entre 2 arêtes [3)]	
13		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
14		_AMODE	Mode alternatif	

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Pour le calibrage "Rayon dans la bague de calibrage", le diamètre et le centre de la bague doivent être connus (4 directions de mesure).
Pour le calibrage "Rayon sur deux arêtes", uniquement la distance des arêtes en direction de l'axe de mesure doit être connu (2 directions de mesure).
Pour le calibrage "Rayon sur une arête", la valeur de consigne de la surface doit être connue.

4) Variante de mesure : calibrage uniquement sur une surface (longueur sur surface), longueur d'outil corrigée résulte de S_MD et S_MA.

5) Uniquement pour variante de mesure "Bague de calibrage, ... et centre connu" (S_MVAR=1xxx02).

6) Axe de mesure uniquement pour variante de mesure S_MVAR=0 ou =xx1x01 ou =xx2x01 ou =20000
Variante de mesure : "Calibrage sur une surface" → Sélection de l'axe de mesure et du sens de la mesure
ou sur "Bague de calibrage, ... et centre connu" → Sélection d'un sens d'axe et sélection de l'axe de mesure et du sens de la mesure
ou sur "Bague de calibrage, ... et centre connu" → Sélection de deux sens d'axe et sélection de l'axe de mesure
ou "Détermination de la longueur du palpeur" → S_MA=3 → 3e axe du plan (pour G17 Z)

7) Variante de mesure uniquement calibrage dans la bague de calibrage ou sur la sphère de calibrage
Pour "Calibrage sur sphère de calibrage", la sphère est contournée à l'équateur lors de la mesure sous angle.

8) Pour calibrage "Rayon dans bague de calibrage" avec centre inconnu quatre directions de mesure dans le plan (pour G17 +-X +-Y).
Pour calibrage "Longueur sur surface" dans le sens moins de l'axe d'outil (pour G17 -Z).

3.26.2.5 Paramètres des cycles de mesure CYCLE978

```
PROC CYCLE978(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL
S_TSA,INT S_MA,INT S_MD,INT S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL S_TZL,REAL
S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT
_AMODE,INT _DP)
```

Tableau 3-12 Paramètres d'appel CYCLE978[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Va-leurs :	UNITES : élément de contour 0 = mesures surface
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM) 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : réservé
				DIZAINES DE MILLIERS : mesure avec/sans pivotement de broche ou orientation de palpeur dans le sens de commutation[9)] 0 = mesure sans pivotement de broche, sans orientation du palpeur 1 = mesure avec pivotement de broche 2 = orientation du palpeur dans le sens de commutation
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Va-leurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2)]	
			Va-leurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction entre 1 et au maximum 999 numéro D (numéro de tranchant) pour correction d'outil pour correction totale et en réglage voir également S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32 000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction dans géométrie d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction de l'usure du rayon d'outil) 1 = correction d'outil correction totale (CT)[5)] La valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + valeur de correction CT (ancienne), CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] La valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale, non inversée) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icon+quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
5	X0	S_SETV	Valeur de consigne	
6	DFA	S_FA	Course de mesure	
7	TSA	S_TSA	Zone de fiabilité	
8	X	S_MA	Numéro de l'axe de mesure[7)] (plage de valeurs de 1 à 3)	
			Va-leurs :	1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z) Mesure dans le sens de l'outil
9		S_MD	Direction de mesure de l'axe de mesure	
			Va-leurs :	1 = direction de mesure positive 2 = direction de mesure négative
10	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (plage de valeurs de 1 à 9)	

N°	Paramètre de masque	Paramètre de cycle	Signification	
11	TR	S_TNAME	Nom d'outil[3)]	
12	DL	S_DLNUM	Correction totale / en réglage numéro DL[5)]	
13	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	
14	TZL	S_TZL	Correction zéro[2), 3)]	
15	DIF	S_TDIF	Contrôle d'écart dimensionnel[2), 3)]	
16	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[3)]	
17	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[3)]	
18	TMV	S_TMV	Plage de correction lors du calcul de moyenne[2)]	
19	FW	S_K	Facteur de pondération pour calcul de moyenne[2)]	
20	EVN	S_EVNUM	Bloc de données mémoire de valeurs empiriques[2), 8)]	
21		S_MCBIT	réservé	
22		_DMODE	Mode display	
			Va-leurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
23		_AMODE	Mode alternatif	
			Va-leurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

1) Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Uniquement pour correction dans outil, autrement paramètre = ""

4) Uniquement pour correction dans outil et si utilisation d'une tolérance dimensionnelle, autrement paramètre = 0

5) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR . De plus, il faut que le paramètre machine général MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK bit8=1.

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

7) Correction dans géométrie d'outil :
- pour mesure dans le plan (S_MA=1 ou S_MA=2) correction dans rayon d'outil
- pour mesure dans le sens de l'outil (S_MA=3) Correction de longueur d'outil L1

8) Calcul de moyenne empirique possible pour correction d'outil et correction dans DO
Plage de valeurs de la mémoire des valeurs moyennes empiriques :
De 1 à 20, numéro de la mémoire des valeurs empiriques, voir donnée de réglage propre au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
De 10 000 à 200 000 numéro de la mémoire des valeurs moyennes, voir donnée de réglage propre au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

9) Lors de la mesure avec pivotement de broche, le rayon/le diamètre du palpeur doit être déterminé avec précision avec une variante de calibrage du CYCLE976 Rayon sur anneau ou sur arête ou sur sphère. Sinon le résultat de mesure est faussé.

3.26.2.6 Paramètres des cycles de mesure CYCLE998

```
PROC CYCLE998(INT S_MVAR,INT S_KNUM,INT S_RA,INT S_PRNUM,REAL S_SETV,REAL S_STA1,REAL
S_INCA,REAL S_FA,REAL S_TSA,INT S_MA,INT S_MD,REAL S_ID,REAL S_SETV0,REAL S_SETV1,REAL
S_SETV2,REAL S_SETV3,INT S_NMSP,INT S_EVNUM,INT _DMODE,INT _AMODE)
```

Tableau 3-13 Paramètres d'appel CYCLE998 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=5)	
			Valeurs :	UNITES : élément de contour 5 = mesure arête (un angle) 6 = mesurer plan (deux angles)
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer et pas de correction du DO 1 = mesure, détermination et correction du DO (voir S_KNUM)
				MILLIERS : zone de protection 0 = aucune zone de protection prise en compte 1 = prise en compte d'une zone de protection
				DIZAINES DE MILLIERS : mesurer avec pivotement de broche (mesure différentielle) 0 = mesure sans pivotement de broche 1 = mesure avec pivotement de broche
				CENTAINES DE MILLIERS : mesure sous angle ou en parallèle à l'axe 0 = mesure sous angle 1 = mesure en parallèle à l'axe
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base [3)] 0 = correction précise 1 = correction approximative

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	A, B, C	S_RA	Cible de correction Rotation de coordonnées ou axe rotatif [14]	
			Valeurs :	0 = cible de correction Rotation de coordonnées autour de l'axe résultant du paramètre S_MA [4] >0 = cible de correction Axe rotatif. Numéro de l'axe de canal de l'axe rotatif (de préférence table rotative). La correction d'angle est effectuée dans la partie de translation du DO de l'axe rotatif.
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (par défaut=1)	
5	DX / DY / DZ	S_SETV	Distance (incrémentale) de la position de départ au point de mesure P1 de l'axe de mesure (S_MA) [5]	
6	α	S_STA1	Valeur de consigne d'angle pour "Orienter arête" ou pour "Orienter plan" autour du 1er axe du plan (pour G17 X) [9]	
7	β	S_INCA	Valeur de consigne d'angle pour "Orienter plan" autour du 2e axe du plan (pour G17 Y) [9]	
8	DFA	S_FA	Course de mesure	
9	TSA	S_TSA	Zone de fiabilité Surveillance de l'écart d'angle par rapport à la valeur de consigne d'angle [degré] [6]	
10	X / Y / Z	S_MA	Axe de mesure, axe de décalage [7] (par défaut=201)	
			Valeurs :	UNITES : numéro de l'axe de mesure 1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z)
				DIZAINES : réservé
				CENTAINES : numéro de l'axe de décalage 1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z)
11	+-	S_MD	Direction de mesure de l'axe de mesure [8]	
			Valeurs :	0 = direction de mesure est déterminée à partir de la valeur de consigne et de la position réelle de l'axe de mesure (compatibilité) 1 = direction de mesure positive 2 = direction de mesure négative
12	L2	S_ID	Pour variante de mesure "Orienter arête" : Distance (incrémentale) entre les points de mesure P1 et P2 dans l'axe de décalage (valeur >0) Pour variante de mesure "Orienter plan", les paramètres valables sont énumérés ci-après.	
13	L2	S_SETV0	Distance entre les points de mesure P1 et P2 dans le 1er axe du plan [10]	
14		S_SETV1	Distance entre les points de mesure P1 et P2 dans le 2e axe du plan [11], [12]	
15	L3x	S_SETV2	Distance entre les points de mesure P1 et P3 dans le 1er axe du plan [11]	
16	L3y	S_SETV3	Distance entre les points de mesure P1 et P3 dans le 2e axe du plan [10]	
17	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (par défaut=1)	
18		S_EVNUM	Bloc de données mémoire de valeurs empiriques [2], [13]	

N°	Paramètre de masque	Paramètre de cycle	Signification	
19		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
20		_AMODE	réservé (mode alternatif)	

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) DO "de précision" uniquement si cible de correction Axe rotatif et MD 52207 $MCS_AXIS_USAGE_ATTRIB[n] bit6=1.
Si le DO n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO.

4) Exemple pour correction dans rotation de coordonnées : S_MA=102 axe de mesure Y axe de décalage X résulte en rotation de coordonnées autour de Z (pour G17)

5) Valeur pertinente uniquement si utilisation d'une zone de protection (S_MVAR Position des MILLIERS = 1)

6) Les angles des paramètres S_STA1 et S_TSA sont additionnés lors du positionnement du point de mesure P1 par rapport au point de mesure P2 dans l'axe de décalage.

7) Le numéro de l'axe de mesure ne doit pas être identique au numéro de l'axe de décalage (par exemple 101 pas autorisé)

8) Sens de mesure uniquement "Orienter arête" et "Mesurer en parallèle à l'axe (S_MVAR=10x105)

9) Plage d'angle S_STA1 ±45 degrés pour "Orienter arête"
Plage d'angle S_STA1 0 à +60 degrés et S_INCA ±30 degrés pour "Orienter plan"

10) Pour variante de mesure "Orienter plan" et "Orienter arête"

11) Pour variante de mesure "Mesurer plan" et "Mesurer en parallèle à l'axe"

12) Pas pour version de cycle de mesure SW04.04.

13) Calcul de valeur empirique pour la correction dans DO ; plage de valeurs de la mémoire des valeurs moyennes empiriques : De 1 à 20, numéro de la mémoire de valeurs empiriques, voir donnée de réglage propre au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]

14) Le paramètre S_RA n'est pertinent que pour la variante de mesure "Orienter arête" (S_MVAR xxxxx5).

3.26.2.7 Paramètres des cycles de mesure CYCLE977

```
PROC CYCLE977(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,REAL S_SETV0,REAL
S_SETV1,REAL S_FA,REAL S_TSA,REAL S_STA1,REAL S_ID,REAL S_SZA,REAL S_SZO,INT S_MA,INT
S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL S_TZL,REAL S_TDIF,REAL S_TUL,REAL S_TLL,REAL
S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT _AMODE,REAL S_XM,REAL_S_YM,INT _DP)
```

Tableau 3-14 Paramètres d'appel CYCLE977[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : élément de contour (plage de valeurs de 1 à 6) 1 = mesure alésage 2 = mesure tourillon (arbre) 3 = mesure rainure 4 = mesure âme 5 = mesure rectangle, intérieur 6 = mesure rectangle, extérieur
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou. pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM) 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : zone de protection 0 = aucune zone de protection prise en compte 1 = prise en compte d'une zone de protection
				DIZAINES DE MILLIERS : mesure avec/sans pivotement de broche (mesure différentielle) ou orientation du palpeur dans le sens de commutation 0 = mesure sans pivotement de broche, sans orientation du palpeur 1 = mesure avec pivotement de broche 2 = orientation du palpeur dans le sens de commutation
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2)]	
			Valeurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction entre 1 et au maximum 999 numéro D (numéro de tranchant) pour correction d'outil ; pour correction totale et en réglage, voir aussi S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32 000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction rayon d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction de l'usure du rayon d'outil) 1 = correction d'outil correction totale (CT)[5)] La valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + CT (ancienne) valeur de correction, CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] La valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale, non inversée) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
5	X0	S_SETV	Valeur de consigne	
6	X0	S_SETV0	Valeur de consigne pour rectangle 1er axe du plan (X pour G17)	
7	Y0	S_SETV1	Valeur de consigne pour rectangle 2e axe du plan (Y pour G17)	
8	XM	S_XM	Spécification de valeur de consigne pour le centre axe géométrique X	
9	YM	S_YM	Spécification de valeur de consigne pour le centre axe géométrique Y	
10	DFA	S_FA	Course de mesure	
11	TSA	S_TSA	Zone de fiabilité	
12	α 0	S_STA1	Angle de départ	

N°	Paramètre de masque	Paramètre de cycle	Signification	
13	DZ	S_ID	Valeur relative 1. Approche relative du 3e axe du plan (Z pour G17) Direction d'approche via signes de S_ID. Pour Mesure tourillon, âme et rectangle, la descente à la hauteur de mesure est définie avec S_ID . 2. Prise en considération d'une zone de protection Pour Mesure alésage, rainure et rectangle intérieur et avec une zone de protection, la hauteur de dépassement est définie avec S_ID .	
14	X1	S_SZA	Diamètre ou longueur (largeur) de la zone de protection[7)]	
15	Y1	S_SZO	Pour mesure rectangle : largeur de la zone de protection du 2e axe du plan	
16	X	S_MA	Numéro de l'axe de mesure[7)] (uniquement pour Mesure rainure ou âme)	
			Valeurs :	1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y)
17	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	
18	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (plage de valeurs de 1 à 9)	
19	TR	S_TNAME	Nom d'outil[2)]	
20	DL	S_DLNUM	Correction totale / en réglage numéro DL[5)]	
21	TZL	S_TZL	Correction zéro[2), 4)]	
22	DIF	S_TDIF	Contrôle d'écart dimensionnel[2), 4)]	
23	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[4)]	
24	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[4)]	
25	TMV	S_TMV	Plage de correction lors du calcul de moyenne[2)]	
26	FW	S_K	Facteur de pondération pour calcul de moyenne[2)]	
27		S_EVNUM	Bloc de données mémoire de moyennes empiriques[2), 8)]	
28		S_MCBIT	réservé	
29		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
30		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

1) Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Uniquement pour correction dans outil, autrement paramètre = ""

4) Uniquement pour correction dans outil et si utilisation d'une tolérance dimensionnelle, autrement paramètre = 0

5) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR . De plus, il faut que le paramètre machine général MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK bit8=1.

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

[7] Diamètre ou largeur de la zone de protection au sein de l'alésage ou de la rainure.
Diamètre resp. largeur de la zone de protection en dehors du tourillon ou de l'âme

[8] Calcul de moyenne empirique possible pour correction d'outil
Plage de valeurs de la mémoire de moyennes empiriques :
entre 1 et 20 numéro (n) de la mémoire de valeurs empiriques, voir donnée de réglage spécifique au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
Entre 10 000 et 200 000 numéro (n) de la mémoire de valeurs moyennes, voir donnée de réglage spécifique au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

3.26.2.8 Paramètres des cycles de mesure CYCLE961

```
PROC CYCLE961(INT S_MVAR,INT S_KNUM,INT S_PRNUM,REAL S_SETV0,REAL S_SETV1,REAL
S_SETV2,REAL S_SETV3,REAL S_SETV4,REAL S_SETV5,REAL S_SETV6,REAL S_SETV7,REAL S_SETV8,REAL
S_SETV9,REAL S_STA1,REAL S_INCA,REAL S_ID,REAL S_FA,REAL S_TSA,INT S_NMSP,INT S_MCBIT,INT
_DMODE,INT _AMODE)
```

Tableau 3-15 Paramètres d'appel CYCLE961 [1]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut ≥ 6)	
			Valeurs :	UNITES : élément de contour 5 = configuration angle intérieur droit, définition de valeur de consigne angle et distances A1 à A3 6 = configuration angle extérieur droit, définition de valeur de consigne angle et distances A1 à A3 7 = configuration angle intérieur, définition angle et distances A1 à A4 8 = configuration angle extérieur, définition angle et distances A1 à A4
				DIZAINES : définition de valeur de consigne sous forme de distance ou par quatre points 0 = valeur de consigne est définie sous forme de distance (polaire) 1 = valeur de consigne est définie par quatre points (points de mesure P1 à P4)
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou pas de correction d'outil) 1 = mesure, détermination et correction du DO, voir S_KNUM
				MILLIERS : zone de protection 0 = aucune prise en considération d'une zone de protection (obstacle) 1 = prise en considération d'une zone de protection (obstacle), voir S_ID
				DIZAINES DE MILLIERS : position de l'angle dans le SCP 0 = la position de l'angle est déterminée via le paramètre S_STA1 (compatibilité) 1 = position 1 de l'angle dans le point de départ positionné de la mesure [6] 2 = position 2 de l'angle, distances dans le 1er axe du plan (pour G17 X) sont négatives (voir S_SETV0, S_SETV1) 3 = position 3 de l'angle, distances dans le 1er et 2e axe du plan (pour G17 XY) sont négatives (voir S_SETV0 à S_SETV3) 4 = position 4 de l'angle, distances dans le 2e axe du plan (pour G17 Y) sont négatives (voir S_SETV2, S_SETV3)

N°	Paramètre de masque	Paramètre de cycle	Signification	
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2]	
			Va-leurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 9 = correction DO actif ou pour G500 dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[5] 1 = correction grossière
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
4	L1/X1	S_SETV0	Distance L1 entre le pôle et le point de mesure P1 en direction du 1er axe du plan (pour G17 X)[3] (si la distance L1=0, L1 = M_SETV1 / 2 est calculé automatiquement) ou point de départ P1x du 1er axe du plan (pour G17 X) [4]	
5	L2/Y1	S_SETV1	Distance L2 entre le pôle et le point de mesure P2 en direction du 1er axe du plan [3] ou point de départ P1y du 2e axe du plan (pour G17 Y) [4]	
6	L3/X2	S_SETV2	Distance L3 entre le pôle et le point de mesure P3 en direction du 2e axe du plan [3] (si la distance L3=0, L3 = M_SETV3 / 2 est calculé automatiquement lorsque l'angle n'est pas droit) ou point de départ P2x du 1er axe du plan [4]	
7	L4/Y2	S_SETV3	Distance L4 entre le pôle et le point de mesure P3 en direction du 2e axe du plan lorsque l'angle n'est pas droit [3] ou point de départ P2y du 2e axe du plan [4]	
8	XP/X3	S_SETV4	Position du pôle dans le 1er axe du plan [3] ou point de départ P3x du 1er axe du plan [4]	
9	XP/Y3	S_SETV5	Position du pôle dans le 2e axe du plan [3] ou point de départ P3x du 2e axe du plan [4]	
10	X4	S_SETV6	Point de départ P4x du 1er axe du plan [4]	
11	Y4	S_SETV7	Point de départ P4y du 2e axe du plan [4]	
12	X0	S_SETV8	Valeur de consigne de l'angle mesuré dans le 1er axe du plan lors de la correction dans DO	
13	Y0	S_SETV9	Valeur de consigne de l'angle mesuré dans le 2e axe du plan lors de la correction dans DO	
14	α0	S_STA1	Angle de départ depuis direction positive du 1er axe du plan vers l'arête de référence de la pièce dans le SCM (+-270 degrés)	
15	α1	S_INCA	Angle entre arêtes de référence de pièce lors de la mesure d'un angle pas droit [7]	
16	DZ	S_ID	Valeur d'approche à hauteur de mesure pour chaque point de mesure lorsqu'une zone de protection est activée (voir S_MVAR).	
17	DFA	S_FA	Course de mesure	
18	TSA	S_TSA	Zone de fiabilité Surveillance de l'écart d'angle par rapport à la valeur de consigne d'angle [degré]	

N°	Paramètre de masque	Paramètre de cycle	Signification	
19	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (plage de valeurs de 1 à 9) [2]	
20		S_MCBIT	réservé	
21		_DMODE	Mode display	
			Va-leurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
22		_AMODE	Mode alternatif	

1) Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Définition des points de mesure en coordonnées polaires tenant compte de l'angle de départ S_STA1 pour le point de mesure 3 ou 4 de l'incrément angulaire S_INCA

4) Définition des points de mesure dans le système de coordonnées cartésiennes (au moyen de 4 points)

5) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO.

6) Plage de valeurs Angle S_INCA : -180 à +180 degrés

7) Angle de départ S_STA1, plage de valeurs : angle droit : +- 90 degrés, angle quelconque : +- 45 degrés

3.26.2.9 Paramètres des cycles de mesure CYCLE979

```
PROC CYCLE979(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL
S_TSA,REAL S_CPA,REAL S_CPO,REAL S_STA1,REAL S_INCA,INT S_NMSP,STRING[32] S_TNAME,REAL
S_DLNUM,REAL S_TZL,REAL S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT
S_MCBIT,INT _DMODE,INT _AMODE,REAL S_XM,REAL S_YM,INT _DP)
```

Tableau 3-16 Paramètres d'appel CYCLE979[0)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : élément de contour 1 = mesure alésage 2 = mesure tourillon (arbre)
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou. pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM) 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : nombre de points de mesure 0 = 3 points de mesure 1 = 4 points de mesure
				DIZAINES DE MILLIERS : mesure avec/sans pivotement de broche (mesure différentielle) ou orientation du palpeur dans le sens de commutation 0 = mesure sans pivotement de broche, sans orientation du palpeur 1 = mesure avec pivotement de broche 2 = orientation du palpeur dans le sens de commutation
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2)]	
			Va-leurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction entre 1 et au maximum 999 numéro D (numéro de tranchant) pour correction d'outil ; pour correction totale et en réglage, voir aussi S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32 000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction dans rayon d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction de l'usure du rayon d'outil) 1 = correction d'outil correction totale (CT)[5)] La valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + valeur de correction CT (ancienne), CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] La valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale, non inversée) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
5	X0	S_SETV	Valeur de consigne	
6	DFA	S_FA	Course de mesure	
7	TSA	S_TSA	Zone de fiabilité	
8	X0	S_CPA	Centre du 1er axe du plan (pour G17 X)	
9	Y0	S_CPO	Centre du 2e axe du plan (pour G17 Y)	
10	XM	S_XM	Spécification de valeur de consigne pour le centre axe géométrique X	
11	YM	S_YM	Spécification de valeur de consigne pour le centre axe géométrique Y	
12	alpha 0	S_STA1	Angle de départ[7)]	
13	alpha 1	S_INCA	Incrément angulaire[8)]	
14	Mesures	S_NMSP	Nombre de mesures au même endroit [1)] (plage de valeurs de 1 à 9)	
15	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	

N°	Paramètre de masque	Paramètre de cycle	Signification	
16	T	S_TNAME	Nom d'outil[2)]	
17	DL	S_DLNUM	Correction totale / en réglage numéro DL[1), 4)]	
18	TZL	S_TZL	Correction zéro[1), 2)]	
19	DIF	S_TDIF	Contrôle d'écart dimensionnel[1), 2)]	
20	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[2)]	
21	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[2)]	
22	TMV	S_TMV	Plage de correction lors du calcul de moyenne[1)]	
23	FW	S_K	Facteur de pondération pour calcul de moyenne[1)]	
24		S_EVNUM	Bloc de données mémoire de valeurs empiriques[1), 6)]	
25		S_MCBIT	réservé	
26		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
27		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

0) Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

1) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

2) Uniquement pour correction dans outil, autrement paramètre = ""

3) Uniquement pour correction dans outil et si utilisation d'une tolérance dimensionnelle, autrement paramètre = 0

4) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR .

5) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO.

6) Calcul de moyenne empirique uniquement possible pour correction d'outil
Plage de valeurs de la mémoire de valeurs moyennes empiriques :
De 1 à 20, numéro de la mémoire de valeurs empiriques, voir donnée de réglage spécifique au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
De 10 000 à 200 000 numéro de la mémoire de valeurs moyennes, voir donnée de réglage spécifique au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

7) Plage de valeurs Angle de départ -360 à +360 degrés

8) Plage de valeur Incrément angulaire >0 à ≤90 degrés pour 4 points de mesure resp. >0 à ≤120 degrés pour 3 points de mesure.

3.26.2.10 Paramètres de cycles de mesure CYCLE997

```
PROC CYCLE997 (INT S_MVAR,INT S_KNUM,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL S_TSA,REAL
S_STA1,REAL S_INCA,REAL S_SETV0,REAL S_SETV1,REAL S_SETV2,REAL S_SETV3,REAL S_SETV4,REAL
S_SETV5,REAL S_SETV6,REAL S_SETV7,REAL S_SETV8,REAL S_TNVL,INT S_NMSP,INT S_MCBIT,INT
_DMODE,INT _AMODE)
```

Tableau 3-17 Paramètres d'appel CYCLE997 [1), 2)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=9)	
			Va-leurs :	UNITES : élément de contour 9 = mesurer sphère
				DIZAINES : répétition de la mesure 0 = sans répétition de la mesure 1 = avec répétition de la mesure
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO) 1 = mesure, détermination et correction du DO (voir S_KNUM)
				MILLIERS : stratégie de mesure 0 = mesure en parallèle à l'axe, sans angle de départ, orientation du palpeur selon SD55740, bit 1 1 = mesure le long, avec angle de départ, orientation du palpeur selon SD55740, bit 1 2 = mesure le long, avec angle de départ, orienter le palpeur dans le sens de commutation 3 = mesure en parallèle à l'axe, avec angle de départ, orientation du palpeur selon SD55740, bit 1 4 = mesure en parallèle à l'axe, avec angle de départ, orienter le palpeur dans le sens de commutation
				DIZAINES DE MILLIERS : nombre des sphères à mesurer 0 = mesurer une sphère 1 = mesurer trois sphères
				CENTAINES DE MILLIERS : nombre des points de mesure, uniquement pour mesure sous angle (tenir compte de la stratégie de mesure : position des MILLIERS > 0) 0 = trois points de mesure pour la mesure sous angle (le long de la sphère) 1 = quatre points de mesure lors de la mesure sous angle (le long de la sphère)
				MILLIONS : détermination du diamètre de consigne de la sphère 0 = aucune détermination du diamètre de consigne de la sphère 1 = détermination du diamètre de consigne de la sphère

N°	Paramètre de masque	Paramètre de cycle	Signification	
2	Sélection	S_KNUM	Correction dans décalage d'origine (DO) ou base ou référence de base [3]	
			Va-leurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et au maximum 99 numéros du décalage d'origine ou entre 1 et au maximum 16 numéros du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction dans DO ou DO de base ou référence de base 0 = correction dans DO réglable 1 = correction dans DO de base propre au canal 2 = correction dans référence de base 3 = correction dans DO de base global [7] 9 = correction dans DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction approximative ou précise dans DO ou DO de base ou référence de base 0 = correction fine[6] 1 = correction grossière
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
4		S_SETV	Diamètre de la/des sphère(s) [4]	
5	DFA	S_FA	Course de mesure	
6	TSA	S_TSA	Zone de fiabilité	
7	alpha 0	S_STA1	Angle de départ pour la mesure sous angle	
8	alpha 1	S_INCA	Incrément angulaire pour la mesure sous angle	
9	X1	S_SETV0	Position de consigne de la 1re sphère du 1er axe du plan (pour G17 X) pour la mesure de 3 sphères	
10	Y1	S_SETV1	Position de consigne de la 1re sphère du 2e axe du plan (pour G17 Y) pour la mesure de 3 sphères	
11	Z1	S_SETV2	Position de consigne de la 1re sphère du 3e axe du plan (pour G17 Z) pour la mesure de 3 sphères	
12	X2	S_SETV3	Position de consigne de la 2e sphère du 1er axe du plan pour la mesure de 3 sphères	
13	Y2	S_SETV4	Position de consigne de la 2e sphère du 2e axe du plan pour la mesure de 3 sphères	
14	Z2	S_SETV5	Position de consigne de la 2e sphère du 3e axe du plan pour la mesure de 3 sphères	
15	X3	S_SETV6	Position de consigne de la 3e sphère du 1er axe du plan pour la mesure de 3 sphères	
16	Y3	S_SETV7	Position de consigne de la 3e sphère du 2e axe du plan pour la mesure de 3 sphères	
17	Z3	S_SETV8	Position de consigne de la 3e sphère du 3e axe du plan pour la mesure de 3 sphères	
18	TVL	S_TNVL	Valeur limite pour la distorsion du triangle (somme des écarts) pour la mesure de 3 sphères [5]	
19	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (plage de valeurs de 1 à 9)	
20		S_MCBIT	réservé	

N°	Paramètre de masque	Paramètre de cycle	Signification	
21		_DMODE	Mode display	
			Va-leurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
22		_AMODE	Mode alternatif	

1) Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Positionnement intermédiaire avec rotation autour de la sphère à l'équateur

4) Mesure de 3 sphères : le même diamètre de consigne (_SETV) est valable pour toutes les sphères

5) Valeur par défaut pour S_TNVL=1.2
Correction dans DO : La correction dans DO est uniquement effectuée si la distorsion déterminée est inférieure à la valeur limite S_TNVL .

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

7) Pour la variante de mesure "Mesurer trois sphères", une correction dans un frame de base global est impossible (S_KNUM = 3001 bis 3016) étant donné que le frame ne possède pas de composants de rotation.

3.26.2.11 Paramètres du cycle de mesure CYCLE995

```
PROC CYCLE995 (INT S_MVAR,INT S_KNUM,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL S_TSA,REAL
S_STA1,REAL S_INCA,REAL S_DZ,REAL S_SETV0,REAL S_SETV1,REAL S_SETV2,REAL S_TUL,REAL
S_TZL,INT S_NMSP,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-18 Paramètres d'appel CYCLE995 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=5)	
			Valeurs :	UNITES : élément de contour 5 = Géométrie de broche (parallélisme à l'axe d'outil)
				DIZAINES : répétition de la mesure 1 = avec répétition de mesure
				CENTAINES : aucune cible de correction 0 = mesure seule
				MILLIERS : stratégie de mesure 2 = mesure sous angle, alignement du palpeur dans le sens de commutation
				DIZAINES DE MILLIERS : nombre des sphères à mesurer 0 = mesurer une sphère
				CENTAINES DE MILLIERS : nombre de points de mesure 1 = 4 points de mesure lors de la mesure sous angle (rotation autour de la sphère)
				MILLIONS : détermination du diamètre de consigne de la sphère 0 = aucune détermination du diamètre de consigne de la sphère 1 = détermination du diamètre de consigne de la sphère
2	Sélection	S_KNUM	destination de correction 0 = 0	
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
4	DM	S_SETV	Diamètre de la sphère de calibrage [4)]	
5	DFA	S_FA	Course de mesure	
6	TSA	S_TSA	Zone de fiabilité [5)]	
7	alpha 0	S_STA1	Angle de départ pour la mesure sous angle [3)]	
8		S_INCA	Incrément angulaire pour la mesure sous angle [2)]	
9	DZ	S_DZ	Écart entre la 1re mesure P1 et la 2e mesure P2 sur l'axe du palpeur	
10		S_SETV0	Position de consigne de la sphère, 1er axe du plan (pour G17 X) [2)]	
11		S_SETV1	Position de consigne de la sphère, 2ème axe du plan (pour G17 Y) [2)]	
12		S_SETV2	Position de consigne de la sphère, 3ème axe du plan (pour G17 Z) [2)]	
13	TUL	S_TUL	Limite supérieure de tolérance du décalage angulaire	
14	TZL	S_TZL	Plage de correction zéro[1), 4)]	
15	Nombre	S_NMSP	Nombre de mesures au même endroit [2)] (plage de valeurs de 1 à 9)	
16		S_MCBIT	réservé [2)]	
17		_DMODE	Mode display	
			Valeurs :	UNITES : Plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)

N°	Paramètre de masque	Paramètre de cycle	Signification	
18		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

1) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

2) Les paramètres ne sont pas utilisés actuellement et ils ne sont pas affichés dans le masque de saisie.
Le paramètre Incrément angulaire S_INCA est préréglé sur 90 degrés.

3) Plage de valeurs Angle de départ -360 à +360 degrés

4) pour tolérance dimensionnelle "oui" :
Si les angles mesurés sont inférieurs à la valeur de la plage de correction zéro TZL, les paramètres de résultat se rapportant aux angles (_OVR[2], _OVR[3]) et aux écarts (_OVR[7], _OVR[8]) sont mis à zéro.
L'affichage TZL se fait via les données de réglage générales SD54760 $SNS_MEA_FUNCTION_MASK_PIECE , bit 25=1.
(Sélection Débloquer correction zéro pour la mesure de l'angularité de broche)

5) Le paramètre TSA se rapporte à la 1ère mesure de la sphère de calibrage.

3.26.2.12 Paramètres des cycles de mesure CYCLE996

```
PROC CYCLE996(INT S_MVAR,INT S_TC,INT S_PRNUM,REAL S_SETV,REAL S_STA1,REAL S_SETV0,REAL
S_SETV1,REAL S_SETV2,REAL S_SETV3,REAL S_SETV4,REAL S_SETV5,REAL S_TNVL,REAL S_FA,REAL
S_TSA,INT S_NMSP,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-19 Paramètres d'appel CYCLE996 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=1)	
			Valeurs :	UNITES : ordre des points de mesure 0 = calcul cinématique (sélection avec : Affichage du résultat, journal, modification du bloc de données d'orientation, acquittement éventuel de l'utilisateur), voir _AMODE 1 = 1re mesure 2 = 2e mesure 3 = 3e mesure
				DIZAINES : réservé 0 = 0
				CENTAINES : Variante de mesure pour 1re à 3e mesure 0 = mesure de la sphère de calibrage en parallèle à l'axe 1 = mesure de la sphère de calibrage sous angle et pas de poursuite de la broche [3)] 2 = mesure de la sphère de calibrage et poursuite de la broche dans le sens de commutation du palpeur [3)] 3 = mesure en parallèle à l'axe, avec angle de départ [8)] 4 = mesure en parallèle à l'axe, poursuite de la broche dans le sens de commutation du palpeur [8)]
				MILLIERS : destination de correction pour le calcul de la cinématique [4)] 0 = mesure seule. Les blocs de données d'orientation sont calculés, mais restent inchangés. 1 = calcul du bloc de données d'orientation. Les blocs de données d'orientation sont modifiés, le cas échéant après acquittement par l'utilisateur. [4)]
				DIZAINES DE MILLIERS : axe de mesure (axe rotatif 1 ou 2) ou résultat vectoriel ouvert ou fermé pour Calcul de la cinématique 0 = résultat vectoriel fermé (uniquement pour Calcul de la cinématique) 1 = axe rotatif 1 (uniquement pour 1re à 3e mesure) 2 = axe rotatif 2 (uniquement pour 1re à 3e mesure) [5)] 3 = résultat vectoriel ouvert (uniquement pour Calcul de la cinématique)
				CENTAINES DE MILLIERS : normalisation axe rotatif 1 pour Calcul de la cinématique 0 = pas de normalisation axe rotatif 1 1 = normalisation dans la direction du 1er axe du plan (pour G17 X) 2 = Normalisation dans la direction du 2e axe du plan (pour G17 Y) 3 = Normalisation dans la direction du 3e axe du plan (pour G17 Z)
				MILLIONS : normalisation axe rotatif 2 pour Calcul de la cinématique [5)] 0 = pas de normalisation axe rotatif 2 1 = normalisation dans la direction du 1er axe du plan (pour G17 X) 2 = Normalisation dans la direction du 2e axe du plan (pour G17 Y) 3 = Normalisation dans la direction du 3e axe du plan (pour G17 Z)
				DIZAINES DE MILLIONS : fichiers journaux 0 = pas de fichier journal 1 = fichier journal contenant les vecteurs calculés (Toolcarrier) et la 1re transformation dynamique 5 axes (TRAORI(1)), si cela est configuré dans les MD.
2		S_TC	Numéro du bloc de données d'orientation (Toolcarrier)	

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
4		S_SETV	Diamètre de la sphère de calibrage	
5	alpha 0	S_STA1	Angle de départ pour la mesure sous angle	
6	alpha 0	S_SETV0	Valeur de position axe rotatif 1 (si axe rotatif manuel ou semi-automatique)	
7	alpha 1	S_SETV1	Valeur de position axe rotatif 2 (si axe rotatif manuel ou semi-automatique) [6]	
8	XN	S_SETV2	Valeur de position pour normalisation axe rotatif 1	
9	XN	S_SETV3	Valeur de position pour normalisation axe rotatif 2 [6]	
10	delta	S_SETV4	Valeur de tolérance des vecteurs offset I1 à I4	
11	delta	S_SETV5	Valeur de tolérance des vecteurs d'axe rotatif V1 et V2	
12	TVL	S_TNVL	Valeur limite du segment angulaire de l'axe rotatif (plage de valeurs 1 à 60 degrés) (par défaut=20) [7]	
13	DFA	S_FA	Course de mesure	
14	TSA	S_TSA	Zone de fiabilité	
15	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (par défaut=1)	
16		S_MCBIT	réservé	
17		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
18		_AMODE	Mode alternatif	
			Valeurs :	UNITES : contrôle de tolérance oui/non 0 = non 1 = oui : Evaluation des valeurs de tolérance des vecteurs S_SETV4, S_SETV5
				DIZAINES : acquittement de l'utilisateur pour l'inscription des vecteurs calculés dans le bloc de données d'orientation [4] 0 = oui : l'utilisateur doit acquitter la modification. 1 = non : les vecteurs calculés sont immédiatement inscrits (uniquement lorsque les positions des CENTAINES et des MILLIERS = 0).
				CENTAINES : affichage du résultat [5] 0 = non 1 = oui
				MILLIERS : modification possible de l'affichage du résultat 0 = non 1 = oui, et modifiable (uniquement lorsque la position des CENTAINES = 1)

[1] Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

[2] Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

[3] Cette variante permet par exemple d'effectuer des mesures sur la sphère de calibrage lorsque la cinématique est positionnée à 90 degrés, sans collision avec la tige de fixation de la sphère de calibrage. Un angle de départ S_STA1 peut être défini de 0 à 360 degrés. L'incrément angulaire de rotation autour de la sphère est égal à 90 degrés.
La donnée de réglage spécifique au canal SD55634 $SCS_MEA_FEED_PLANE_VALUE est utilisée comme avance sur la trajectoire circulaire.

[4] une interrogation de l'utilisateur avec M0 a lieu avant l'inscription. Les vecteurs sont uniquement inscrits avec démarrage commande numérique.
Si le programme de mesure est interrompu avec RESET , aucun vecteur calculé n'est inscrit.
Les vecteurs sont uniquement inscrits si la tolérance des vecteurs offset n'est pas dépassée lors du calcul.

[5] Affichage du résultat uniquement avec la variante de mesure Calcul cinématique.
Si le résultat de la mesure doit également être affiché après les mesures 1 à 3, la donnée de réglage propre au canal SD55613 $SCS_MEA_RESULT_DISPLAY doit être activée.

[6] axe rotatif 2 uniquement pour cinématiques avec deux axes rotatifs

[7] Valeur limite segment angulaire de l'axe rotatif. Plage des valeurs de `S_TNVL` de 20 à 60 degrés. Les valeurs de `S_TNVL` < 20 degrés peuvent conduire à des imprécisions dues aux imprécisions de mesure du palpeur, de l'ordre de quelques micromètres. Un dépassement de la valeur limite déclenche le message d'erreur 61430, avec affichage de la valeur minimale pour la valeur limite.

[8] Poursuite de la broche dans le sens de commutation du palpeur si SD54760 bit 17 = 1

3.26.2.13 Paramètres de cycle de mesure CYCLE9960

```
PROC CYCLE9960(INT S_MVAR,STRING[40] S_TNAME,INT S_PRNUM,REAL S_SETV,REAL S_SETV1,REAL
S_START_RA1,REAL S_END_RA1,INT S_CMEA_RA1,REAL S_POS_RA2,REAL S_SETV2,REAL
S_START_RA2,REAL S_END_RA2,INT S_CMEA_RA2,REAL S_POS_RA1,REAL S_SETV4,REAL S_FA,REAL
S_TSA,INT S_NMSP,INT S_DMODE,INT S_AMODE,INT S_KNUM)
```

Tableau 3-20 Paramètres d'appel CYCLE9960 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1	Sélection	S_MVAR	Variante de mesure (par défaut=1)	
			Valeurs :	UNITES : Variante de mesure 0 = calcul et mesure de la cinématique (sélection avec : affichage du résultat, protocole, modification du bloc de données d'orientation (voir milliers S_MVAR)) 1 = mesurer la tête de référence 2 = adapter la tête à la tête de référence 3 = mesurer et calculer les points intermédiaires (E996)
				DIZAINES : Réservé
				CENTAINES : Variante de mesure avec la mesure de la sphère 2 = mesure de la sphère de calibrage et poursuite de la broche dans le sens de commutation du palpeur 4 = mesure en parallèle à l'axe, avec angle de départ, poursuite de la broche dans le sens de commutation du palpeur [2)]
				MILLIERS : Destination de correction pour le calcul de la cinématique [3)] 0 = mesurer et calculer. Les blocs de données sont calculés et restent inchangés 1 = les blocs de données calculés sont modifiés, le cas échéant après acquittement par l'utilisateur 2 = la cinématique déjà mesurée est calculée et modifiée, le cas échéant après acquittement par l'utilisateur
				DIZAINES DE MILLIERS : axe de mesure (axe rotatif 1 ou 2) 1 = mesurer et calculer tous les axes rotatifs présents 4 = mesurer et calculer uniquement l'axe rotatif 1 5 = mesurer et calculer uniquement l'axe rotatif 2
2		S_TNAME	Nom de la transformation (bloc de données de pivotement ou transformation sur la base de chaînes cinématiques)	
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
4		S_SETV	Diamètre de la sphère de calibrage	
5	alpha 1	S_SETV1	Angle de départ pour la mesure sous angle pour le 1er axe rotatif	
6			S_START_RA1 : angle de départ du 1er axe rotatif	
7			S_END_RA1 : angle final du 1er axe rotatif	
8			S_CMEA_RA1 : nombre de mesures du 1er axe rotatif, 3 pour Mesurer et calculer la cinématique. Pour les points intermédiaires, jusqu'à 12 mesures possibles.	
9			S_POS_RA2 : position du 2e axe rotatif lors de la mesure du 1er axe rotatif [4)]	
10	alpha 2	S_SETV2	Angle de départ pour la mesure sous angle pour le 2e axe rotatif [4)]	
11			S_START_RA2 : angle de départ du 2e axe rotatif [4)]	
12			S_END_RA2 : angle final du 2e axe rotatif [4)]	
13			S_CMEA_RA2 : nombre de mesures du 2e axe rotatif, 3 pour Mesurer et calculer la cinématique. Pour les points intermédiaires, jusqu'à 12 mesures possibles.[4)]	
14			S_POS_RA1 : position du 1er axe rotatif lors de la mesure du 2e axe rotatif	
15	delta	S_SETV4	Valeur de tolérance des vecteurs offsets	
16	DFA	S_FA	Course de mesure	

N°	Paramètre de masque	Paramètre de cycle	Signification	
17	TSA	S_TSA	Zone de fiabilité	
18	Nombre	S_NMSP	Nombre de mesures au même endroit [5] (par défaut=1)	
19		_DMODE	Mode display	
			Valeurs :	UNITES : Plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
20		_AMODE	Mode alternatif	
			Valeurs :	UNITÉS : Contrôle de tolérance oui/non 0 = non 1 = oui : évaluation des valeurs de tolérance des vecteurs -> S_SETV4, S_SETV5
				DIZAINES : Calibrage automatique 0 = sans calibrage automatique 1 = calibrage automatique
				CENTAINES : Automatisme angle de départ 0 = l'angle de départ réglé est utilisé 1 = calcul automatisé de l'angle de départ, pour chaque point de mesure
21		S_KNUM	Numéro du DO à corriger pour mesurer la tête de référence	
			Valeurs	UNITES : la correction intervient toujours dans le décalage grossier, le décalage fin est effacé
				DIZAINES : 1..99 numéro du DO réglable (1=G54)
				MILLIERS : 9 correction DO réglable actif

1) Toutes les valeurs par défaut = 0 ou marquées comme par défaut = xx

2) Poursuite de la broche dans le sens de commutation du palpeur si SD54760 bit 17 = 1

3) Une interrogation de l'utilisateur avec M0 a lieu avant l'inscription. Les vecteurs sont uniquement inscrits avec démarrage CN. Si le programme de mesure est interrompu avec RESET, aucun vecteur calculé n'est inscrit. Les vecteurs sont uniquement inscrits si la tolérance des vecteurs offsets n'est pas dépassée lors du calcul.

4) Axe rotatif 2 uniquement pour cinématiques avec 2 axes rotatifs

5) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE.

3.26.2.14 Paramètres des cycles de mesure CYCLE982

```
PROC CYCLE982(INT S_MVAR,INT S_KNUM,INT S_PRNUM,INT S_MA,INT S_MD,REAL S_ID,REAL S_FA,REAL
S_TSA,REAL S_VMS,REAL S_STA1,REAL S_CORA,REAL S_TZL,REAL S_TDIF,INT S_NMSP,INT S_EVNUM,INT
S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-21 Paramètres d'appel CYCLE982 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : calibrage / mesure 0 = calibrage du palpeur d'outil 1 = mesure individuelle de l'outil[3)] 2 = mesure multiple de l'outil, détermination de la longueur et du rayon d'outil (pour outils de fraisage)
				DIZAINES : calibrage ou mesure dans le SCM ou SCP 0 = avec référencement dans le SCM[4)] 1 = avec référencement dans le SCP
				CENTAINES : mesure avec ou sans pivotement pour des outils de fraisage 0 = mesure sans pivotement 1 = mesure avec pivotement
				MILLIERS : cible de correction pour outils de fraisage 0 = détermination de la longueur, ou de la longueur et du rayon (voir S_MVAR position des unités) 1 = détermination du rayon lorsque S_MVAR position des unités = 1 2 = détermination de la longueur et du rayon (face frontale) lorsque S_MVAR position des unités = 1 ou 2 3 = fraise trois tailles tranchant supérieur (face arrière) et détermination de la longueur et du rayon [5)]
				DIZAINES DE MILLIERS : position de l'outil de fraisage ou du foret 0 = position axiale de l'outil de fraisage ou du foret, rayon dans le 2e axe du plan (pour G18 X) [7)] 1 = position radiale de l'outil de fraisage ou du foret, rayon dans le 1er axe du plan (pour G18 Z) [7)]
				CENTAINES DE MILLIERS : calibrage incrémental ou mesure 0 = aucune donnée 1 = calibrage incrémental ou mesure
				MILLIONS : positionner la broche sur angle de départ S_STA1 (uniquement pour mesure outils de fraisage) 0 = ne pas positionner la broche 1 = positionner la broche sur angle de départ S_STA1
2	Sélection	S_KNUM	Variante de correction [2)]	
			Valeurs :	UNITES : correction d'outil 0 = aucune donnée (correction d'outil en géométrie) 1 = correction d'outil en usure
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
4	X0	S_MA	Axe de mesure	
			Valeurs :	1 = 1er axe du plan (pour G18 Z) 2 = 2e axe du plan (pour G18 X)

N°	Paramètre de masque	Paramètre de cycle	Signification	
5	+-	S_MD	Sens de la mesure	
			Valeurs :	0 = aucune sélection (direction de mesure est déterminée à partir de la valeur réelle) 1 = positif 2 = négatif
6	Z2	S_ID	Décalage	
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	VMS	S_VMS	Vitesse de mesure variable lors du calibrage [2]	
10	alpha1	S_STA1	Angle de départ pour mesure d'outils de fraisage	
11	alpha2	S_CORA	Angle de correction pour mesure avec pivotement d'outils de fraisage [8]	
12	TZL	S_TZL	Correction zéro lors de la mesure d'outils. Lors du calibrage S_TZL = 0	
13	DIF	S_TDIF	Contrôle d'écart dimensionnel	
14	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (par défaut=1)	
15	EVN	S_EVNUM	Numéro mémoire de moyennes empiriques [2], [9]	
16		S_MCBIT	réservé	
17		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
				DIZAINES : position de coupe pour outils de tournage et de fraisage (uniquement pour affichage dans les masques de saisie 1 à 9)
				CENTAINES : type d'outil 0 = outil de tournage 1 = fraise 2 = foret
				MILLIERS : stratégie d'accostage par rapport au palpeur d'outil 0 = PLUS [X/Z] ; X pour position axiale de l'outil, Z pour position radiale de l'outil 1 = MOINS [X/Z] ; X pour position axiale de l'outil, Z pour position radiale de l'outil
18		_AMODE	Mode alternatif	
			Valeurs :	UNITES : réservé
				DIZAINES : réservé
				CENTAINES : réservé
				MILLIERS : Accostage de la position de départ après le processus de mesure en cas de calibrage et de mesure individuelle (voir S_MVAR - UNITES) 0 = L'outil se trouve décalé de DFA par rapport à l'arête du palpeur 1 = Accostage de la position de départ

[1] Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

[2] Affichage dépendant de la donnée de réglage générale SD54762 _MEA_FUNCTION_MASK_TOOL

[3] Mesure de l'outil de tournage ou de la fraise ou du foret. Axe de mesure dans paramètre S_MA
Outils de tournage spécifiés par positions de tranchant 1...8, outils de fraisage spécifiés par positions des CENTAINES et des MILLIERS dans paramètre S_MVAR.

[4] La mesure et le calibrage s'effectuent dans le système de coordonnées de base (SCM avec transformation cinématique désactivée).

[5] Pas pour mesure incrémentale.

[6] Uniquement pour mesure multiple `S_MVAR=x2x02` ou `x3x02` (exemple : fraise trois tailles ou fraise à rainurer)

[7] Lorsque la donnée de réglage propre au canal SD42950 $SC_TOOL_LENGTH_TYPE = 2, les composantes de longueur d'outil sont affectées comme pour les outils de tournage

[8] Uniquement pour mesure avec pivotement `S_MVAR=xx1x1`

[9] Calcul de valeur empirique
Plage de valeur de la mémoire de valeurs empiriques : de 1 à 20, numéro de la mémoire de valeurs empiriques, voir la donnée de réglage propre au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1].

3.26.2.15 Paramètres des cycles de mesure CYCLE971

```
PROC CYCLE971(INT S_MVAR,INT S_KNUM,INT S_PRNUM,INT S_MA,INT S_MD,REAL S_ID,REAL S_FA,REAL
S_TSA,REAL S_VMS,REAL S_TZL,REAL S_TDIF,INT S_NMSP,REAL S_F1,REAL S_S1,REAL S_F2,REAL
S_S2,REAL S_F3,REAL S_S3,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-22 Paramètres d'appel CYCLE971 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : 0 = calibrage du palpeur d'outil 1 = mesure de l'outil avec broche à l'arrêt (longueur resp. rayon) 2 = mesure de l'outil avec broche en rotation (longueur resp. rayon), voir paramètre S_F1 à S_S4
				DIZAINES : mesure dans le SCM ou le SCP 0 = mesure dans le SCM (avec référencement dans le SCM), mesure de l'outil ou calibrage du palpeur d'outil 1 = mesure dans le SCP (avec référencement dans le SCP), mesure de l'outil ou calibrage du palpeur d'outil
				CENTAINES : Contrôle individuel des dents 0 = non 1 = oui
				MILLIERS : 0 = 0
				DIZAINES DE MILLIERS : Calibrage incrémental ou mesure 0 = aucune donnée 1 = calibrage incrémental ou mesure
				CENTAINES DE MILLIERS : calibrage automatique du palpeur d'outil 0 = ne pas calibrer automatiquement le palpeur d'outil 1 = calibrer automatiquement le palpeur d'outil
				MILLIONS : Calibrage dans le plan avec pivotement de broche 0 = Calibrage dans le plan sans pivotement de broche 1 = Calibrage dans le plan avec pivotement de broche
2	Sélection	S_KNUM	Variante de correction [2)]	
			Valeurs :	UNITES : correction d'outil 0 = aucune donnée (correction d'outil en géométrie) 1 = correction d'outil en usure
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur)	
4	X0	S_MA	Axe de mesure, axe de décalage [4)]	
			Valeurs :	UNITES : numéro de l'axe de mesure 1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z)
				DIZAINES : 0 = 0
				CENTAINES : numéro de l'axe de décalage 0 = aucun axe de décalage 1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y)

N°	Paramètre de masque	Paramètre de cycle	Signification	
5	+-	S_MD	Sens de la mesure	
			Valeurs :	0 = aucune sélection (direction de mesure est déterminée à partir de la valeur réelle) 1 = positif 2 = négatif
6	V	S_ID	Décalage	
			Valeurs :	0 = pour outils sans décalage >0 = • Calibrage : le décalage agit sur le 3e axe du plan (pour G17 Z) si le diamètre de l'outil de calibrage est supérieur au diamètre supérieur du palpeur. L'outil est alors décalé à partir du centre du palpeur d'une distance correspondant au rayon de l'outil, après soustraction de la valeur de S_ID. L'axe de décalage est en outre indiqué dans S_MA . • Mesure : pour les outils à plusieurs arêtes tranchantes, indiquez le décalage entre la longueur d'outil et le point le plus haut de l'arête tranchante, pour la mesure du rayon, ainsi que le décalage du rayon d'outil par rapport au point le plus haut de l'arête tranchante, pour la mesure de la longueur d'outil.
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	VMS	S_VMS	Vitesse de mesure variable lors du calibrage [2)]	
10	TZL	S_TZL	Correction zéro (uniquement pour mesure outil)	
11	DIF	S_TDIF	Contrôle de l'écart dimensionnel pour mesure de l'outil (S_MVAR=xx1 ou S_MVAR=xx2)	
12	Mesures	S_NMSP	Nombre de mesures au même endroit [2)]	
13	F1	S_F1	1re avance pour palpage avec broche en rotation [2)]	
14	S1	S_S1	1re vitesse de rotation pour palpage avec broche en rotation [2)]	
15	F2	S_F2	2e avance pour palpage avec broche en rotation [2)]	
16	S2	S_S2	2e vitesse de rotation pour palpage avec broche en rotation [2)]	
17	F3	S_F3	3e avance pour palpage avec broche en rotation [2)]	
18	S3	S_S3	3e vitesse de rotation pour palpage avec broche en rotation [2)]	
19	EVN	S_EVNUM	Numéro mémoire de valeurs empiriques [2)]	
20		S_MCBIT	Masque des _CBIT resp. _CHBIT	
21		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)

N°	Paramètre de masque	Paramètre de cycle	Signification	
22		_AMODE	Mode alternatif	
			Valeurs :	UNITES : Décalage d'outil pour mesure de rayon 1 = non 2 = oui
				DIZAINES : Sens du décalage d'outil pour mesure de rayon dans le 3e axe du plan (pour G17 Z) 1 = positif 2 = négatif
				CENTAINES : Décalage d'outil pour mesure de longueur ou pour calibrage du palpeur dans le 3e axe 0 = compatibilité, auto 1 = non 2 = oui
				MILLIERS : Sens du décalage d'outil pour mesure de longueur dans l'axe de décalage (voir S_MA CENTAINES) 1 = positif 2 = négatif

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54762 MEA_FUNCTION_MASK_TOOL

3) Uniquement pour correction dans outil et si utilisation d'une tolérance dimensionnelle, autrement paramètre = 0

4) Pour la mesure automatique (S_MVAR=1x00xx) aucun affichage axe de mesure, axe de décalage ⇒ S_MA=0.

3.26.2.16 Paramètres de cycle de mesure CYCLE150

```
PROC CYCLE150(INT S_PICT,INT S_PROT,STRING[160] S_PATH) SAVE
ACTBLOCNO DISPLOF
```

Tableau 3-23 Paramètres d'appel CYCLE150

N°	Paramètre de masque	Paramètre de cycle	Signification	
1	Image des résultats de mesure	S_PICT	Sélection de l'affichage des résultats (par défaut = 0)	
			Valeurs :	UNITES : 0 = Image des résultats de mesure ARRET 1 = Image des résultats de mesure MARCHE
				Dizaines : Sélection du mode d'affichage (valeurs comme SD55613) 1 = afficher l'image des résultats de mesure - Désélection automatique au bout de 8 s 3 = afficher l'image des résultats de mesure - Acquittement avec NC-Start 4 = afficher l'image des résultats de mesure - uniquement pour les alarmes (61303 ... 61306)
2		S_PROT	Sélection de la journalisation (par défaut = 0)	

N°	Paramètre de masque	Paramètre de cycle	Signification	
	Procès-verbal		Valeurs :	UNITES : Sélection Procès-verbal off/on dernière mesure 0 = Procès-verbal ARRET 1 = Procès-verbal MARCHE 2 = Consignation de la dernière mesure
	Type du PV			DIZAINES : Sélection du type de procès-verbal 0 = Procès-verbal standard 1 = Procès-verbal utilisateur (librement définissable)
	Format du procès-verbal			CENTAINES : Sélection du format de procès-verbal 0 = Format texte 1 = Format tableau (pour importation Excel)
	Données du procès-verbal			MILLIERS : Réécrire la sélection ou l'accrocher 0 = nouveau 1 = accrocher
	Procès-verbal Archivage			DIZAINES DE MILLIERS : Sélection de l'emplacement des procès-verbaux 0 = comme le programme pièces 1 = Répertoire
3		S_PATH	Chemin pour le fichier de procès-verbal en fonction de la sélection de l'emplacement des procès-verbaux (indication complète du chemin ou nom de fichier uniquement, par ex. : "//NC:/WKS.DIR/NAME.WPD" ou "MESSPROTOKOLL.TXT"	

3.26.3 Cycles de mesure

Les cycles de mesure sont des sous-programmes spéciaux fournis par Siemens et dédiés à des tâches de mesure. Comme cela est généralement le cas pour les cycles, les cycles de mesure peuvent aussi être adaptés au cas de figure concret par l'intermédiaire de paramètres.

Des cycles de mesure sont disponibles pour les mesures dans les domaines et technologies suivants :

- Mesures d'outil tournage/fraisage
- Mesures de pièce tournage/fraisage

Informations complémentaires

Manuel de programmation Cycles de mesure

3.26.3.1 Vue d'ensemble des paramètres des cycles de mesure

Paramètres des cycles de mesure CYCLE973

```
PROC CYCLE973(INT S_MVAR,INT S_PRNUM,INT S_CALNUM,REAL S_SETV,INT S_MA,INT S_MD,REAL
S_FA,REAL S_TSA,REAL S_VMS,INT S_NMSP,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-24 Paramètres d'appel CYCLE973 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=0012103)	
			Valeurs :	UNITES : calibrage sur une surface, une arête ou dans une rainure 0 = longueur sur surface/arête (dans le SCP) avec valeur de consigne connue 1 = rayon sur surface (dans le SCP) avec valeur de consigne connue 2 = longueur dans rainure (dans le SCM), voir S_CALNUM 3 = rayon dans rainure (dans le SCM), voir S_CALNUM
				DIZAINES : réservé 0 = 0
				CENTAINES : réservé 0 = 0
				MILLIERS : sélection axe de mesure et sens de mesure pour calibrage [2)] 0 = aucune donnée (pour calibrage surface sur fond de rainure, aucun axe de mesure ni direction de mesure sélectionné) [4)] 1 = indiquer sélection axe de mesure et direction de mesure, voir S_MA, S_MD (une direction de mesure dans un axe de mesure) 2 = indiquer sélection axe de mesure, voir S_MA (deux directions de mesure dans un axe de mesure)
				DIZAINES DE MILLIERS : détermination écart de position (inclinaison du palpeur) [2), 3)] 0 = déterminer écart de position 1 = ne pas déterminer écart de position
				CENTAINES DE MILLIERS : réservé 0 = 0
				UN MILLION :Adapter la longueur d'outil [7)] 0 = ne pas adapter la longueur d'outil (uniquement point de déclenchement) 1 = adapter la longueur d'outil
2	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
3		S_CALNUM	Numéro de la rainure de calibrage lors du calibrage sur une rainure (par défaut=1) [5)]	
4		S_SETV	Valeur de consigne lors du calibrage sur une surface	
5	X0	S_MA	Axe de mesure (numéro de l'axe) [6)] (par défaut=1)	
			Valeurs :	1 = 1er axe du plan (pour G18 Z) 2 = 2e axe du plan (pour G18 X) 3 = 3e axe du plan (pour G18 Y) [6)]
6	+-	S_MD	Direction de mesure (par défaut=1)	
			Valeurs :	0 = direction de mesure positive 1 = direction de mesure négative
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	VMS	S_VMS	Vitesse de mesure variable lors du calibrage [2)]	
10	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (par défaut=1)	

N°	Paramètre de masque	Paramètre de cycle	Signification	
11		S_MCBIT	réservé	
12		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
13		_AMODE	Mode alternatif	

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Uniquement pertinent lors du calibrage dans deux directions d'axe

4) Uniquement axe de mesure et direction de mesure sont automatiquement déterminés à partir de la position de tranchant (SL) du palpeur. SL=8 → -X , SL=7 → -Z

5) Le numéro de la rainure de calibrage (n) renvoie aux données de réglage générales suivantes (toutes les positions dans le SCM) :
Pour position de tranchant SL=7 :
SD54615 $SNS_MEA_CAL_EDGE_BASE_AX1[n] Position du fond de la rainure dans le 1er axe du plan (pour G18 Z)
SD54621 $SNS_MEA_CAL_EDGE_PLUS_DIR_AX2[n] Position du flanc de la rainure dans le sens positif du 2e axe du plan (pour G18 X)
SD54622 $SNS_MEA_CAL_EDGE_MINUS_DIR_AX2[n] Position du flanc de la rainure dans le sens négatif du 2e axe du plan
Pour position de tranchant SL=8 :
SD54619 $SNS_MEA_CAL_EDGE_BASE_AX2[n] Position du fond de la rainure dans le 2e axe du plan
SD54620 $SNS_MEA_CAL_EDGE_UPPER_AX2[n] Position du bord supérieur de la rainure dans le 2e axe du plan (uniquement pour prépositionner le palpeur)
SD54617 $SNS_MEA_CAL_EDGE_PLUS_DIR_AX1[n] Position du flanc de la rainure dans le sens positif du 1er axe du plan
SD54618 $SNS_MEA_CAL_EDGE_MINUS_DIR_AX1[n] Position du flanc de la rainure dans le sens négatif du 1er axe du plan
Remarque :
Les valeurs de position du flanc de la rainure +- peuvent être déterminées approximativement.
La largeur de la rainure, résultant de l'écart des valeurs de position du flanc de la rainure, doit être déterminée avec précision (comparateur de précision).
Lors du calibrage dans la rainure, la longueur d'outil du palpeur de l'axe calibré est supposée être égale à 0.
Les valeurs de position du fond de la rainure doivent également être déterminées avec précision sur la machine (pas de cotes tirées d'un plan).

6) Axe de mesure S_MA=3 pour le calibrage sur une surface et sur un tour avec un 3e axe du plan (pour G18 Y).

7) Adapter la longueur d'outil pour calibrage de la longueur dans rainure ou pour des longueurs sur surface.
Le palpeur de pièce dans des tours peut être décrit avec 2 longueurs (X Z).
Palpeur Tournage type 580
Position de tranchant 7 : pour le calibrage de la longueur, la longueur Z est corrigée au choix.
Palpeur Tournage type 580
Longueur de coupe 8 : pour le calibrage de la longueur, la longueur X est corrigée au choix.
Pour les variantes de mesure Rayon sur rainure ou Rayon sur surface, aucune adaptation de la longueur d'outil n'est réalisée. Seuls les points de déclenchement correspondants sont toujours archivés.

Paramètres des cycles de mesure CYCLE974

```
PROC CYCLE974(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,INT S_MA,REAL
S_FA,REAL S_TSA,REAL S_STA1,INT S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL S_TZL,REAL
S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT
_AMODE,INT _DP)
```

Tableau 3-25 Paramètres d'appel CYCLE974[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : 0 = mesurer face frontale 1 = mesure intérieure 2 = mesure extérieure
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM)[3)] 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : réservé
				DIZAINES DE MILLIERS : mesure avec ou sans pivotement de la broche principale (broche rotative) 0 = mesure sans pivotement 1 = mesure avec pivotement
2	Sélection	S_KNUM	Correction dans décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction dans DO ou DO de base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2), 4)]	
			Valeurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction entre 1 et au maximum 999 numéro D (numéro de tranchant) pour correction d'outil ; pour correction totale et en réglage, voir aussi S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction dans géométrie d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune indication (correction de l'usure de longueur d'outil) 1 = correction d'outil correction totale (CT)[5)] Valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + CT (ancienne) valeur de correction, CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale (non inversée)) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
5	X0	S_SETV	Valeur de consigne	
6	X	S_MA	Axe de mesure (numéro de l'axe) (par défaut=1)	
			Valeurs :	1 = 1er axe du plan (pour G18 Z) 2 = 2e axe du plan (pour G18 X) 3 = 3e axe du plan (pour G18 Y) [5)]
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	α	S_STA1	Angle de départ lors de la mesure avec pivotement	
10	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (par défaut=1)	
11	T	S_TNAME	Nom d'outil[2)]	
12	DL	S_DLNUM	Correction totale / en réglage numéro DL[5)]	

N°	Paramètre de masque	Paramètre de cycle	Signification	
13	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	
14	TZL	S_TZL	Correction zéro[2), 4)]	
15	DIF	S_TDIF	Contrôle d'écart dimensionnel[2), 4)]	
16	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[4)]	
17	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[4)]	
18	TMV	S_TMV	Plage de correction lors du calcul de moyenne[2)]	
19	FW	S_K	Facteur de pondération pour calcul de moyenne[2)]	
20	EVN	S_EVNUM	Numéro mémoire de moyennes empiriques[2), 7)]	
21		S_MCBIT	réservé	
22		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
23		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Correction dans DO uniquement possible avec mesure sans pivotement

4) Pour correction d'outil, tenir compte du paramètre machine spécifique au canal MD 20360 TOOL_PARAMETER_DEF_MASK bit0 et bit1

5) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR . De plus, il faut que le paramètre machine général MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK bit8=1.

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

7) Calcul de moyenne empirique uniquement possible pour correction d'outil
Plage de valeurs de la mémoire de valeurs moyennes empiriques :
De 1 à 20, numéro de la mémoire de valeurs empiriques, voir donnée de réglage propre au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
De 10 000 à 200 000, numéro de la mémoire de valeurs moyennes, voir donnée de réglage propre au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

Paramètres des cycles de mesure CYCLE994

```
PROC CYCLE994(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,INT S_MA,REAL
S_SZA,REAL S_SZO,REAL S_FA,REAL S_TSA,INT S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL
S_TZL,REAL S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT
_DMODE,INT _AMODE,INT _DP)
```

Tableau 3-26 Paramètres d'appel CYCLE994[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : mesure intérieure ou extérieure (par défaut = 1) 1 = mesure intérieure 2 = mesure extérieure
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM)[3)] 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : plage de contournement 0 = aucune plage de contournement 1 = axe de contournement du 1er axe du plan (pour G18 Z). Axe de mesure, voir S_MA. 2 = axe de contournement du 2e axe du plan (pour G18 X). Axe de mesure, voir S_MA. 3 = axe de contournement du 3e axe du plan (pour G18 Y). Axe de mesure, voir S_MA. [8)]
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2), 4)]	
			Va-leurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction De 1 à 999 max. numéro D (numéro de tranchant) pour correction d'outil Pour correction totale et en réglage, voir aussi S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction géométrie d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune indication (correction de l'usure de longueur d'outil) 1 = correction d'outil correction totale (CT)[5)] Valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + CT (ancienne) valeur de correction, CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] Valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie ;
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale, non inversée) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
5	X0	S_SETV	Valeur de consigne	
6	X	S_MA	Numéro de l'axe de mesure (par défaut=1)	
			Va-leurs :	1 = 1er axe du plan (pour G18 Z) 2 = 2e axe du plan (pour G18 X) 3 = 3e axe du plan (pour G18 Y) [8)]
7	X1	S_SZA	Distance de contournement dans l'axe de mesure	
8	Y1	S_SZO	Distance de contournement dans l'axe de contournement	
9	DFA	S_FA	Course de mesure	
10	TSA	S_TSA	Zone de fiabilité	
11	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (par défaut=1)	
12	T	S_TNAME	Nom d'outil[2)]	

N°	Paramètre de masque	Paramètre de cycle	Signification	
13	DL	S_DLNUM	Correction totale / en réglage numéro DL[5]	
14	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	
15	TZL	S_TZL	Correction zéro[2), 4)]	
16	DIF	S_TDIF	Contrôle d'écart dimensionnel[2), 4)]	
17	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[4)]	
18	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[4)]	
19	TMV	S_TMV	Plage de correction lors du calcul de moyenne[2)]	
20	FW	S_K	Facteur de pondération pour calcul de moyenne[2)]	
21	EVN	S_EVNUM	Numéro mémoire de valeurs empiriques[2), 7)]	
22		S_MCBIT	réservé	
23		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
24		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Correction dans DO uniquement possible avec mesure sans pivotement

4) Pour correction d'outil, tenir compte du paramètre machine spécifique au PM de canal 20360 TOOL_PARAMETER_DEF_MASK

5) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR . De plus, il faut que le paramètre machine général MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK bit8=1.

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

7) Calcul de moyenne empirique uniquement possible pour correction d'outil
Plage de valeurs de la mémoire des valeurs moyennes empiriques :
De 1 à 20, numéro de la mémoire des valeurs empiriques, voir donnée de réglage spécifique au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
De 10 000 à 200 000 numéro de la mémoire des valeurs moyennes, voir donnée de réglage spécifique au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

8) Si axe Y disponible sur machine

Paramètres des cycles de mesure CYCLE976

```
PROC CYCLE976(INT S_MVAR,INT S_PRNUM,REAL S_SETV,REAL S_SETV0,INT S_MA,INT S_MD,REAL
S_FA,REAL S_TSA,REAL S_VMS,REAL S_STA1,INT S_NMSP,INT S_SETV1,INT _DMODE,INT _AMODE)
```

Tableau 3-27 Paramètres d'appel CYCLE976[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=1000)	
			Valeurs :	UNITES : calibrage sur surface, sur sphère de calibrage ou dans bague de calibrage[2)] 0 = longueur sur surface avec valeur de consigne connue 1 = rayon dans bague de calibrage avec diamètre connu (valeur de consigne) et centre connu. 2 = rayon dans bague de calibrage avec diamètre connu (valeur de consigne) et centre inconnu 3 = rayon et longueur sur sphère de calibrage 4 = rayon sur arête avec valeur de consigne connue Tenir compte de la sélection axe de mesure et direction de mesure. [3)] 5 = Rayon entre deux arêtes avec consigne connue et distance par rapport aux arêtes. L'axe de mesure doit être sélectionné.
				DIZAINES : réservé 0 = 0
				CENTAINES : réservé 0 = 0
				MILLIERS : sélection axe de mesure et direction de mesure lors du calibrage 0 = aucune donnée (aucune axe de mesure ni direction de mesure sélectionné)[8)] 1 = indiquer axe de mesure et direction de mesure sélectionné, voir S_MA, S_MD (une direction de mesure dans un axe de mesure) 2 = indiquer axe de mesure sélectionné, voir S_MA (deux directions de mesure dans un axe de mesure)
				DIZAINES DE MILLIERS : détermination écart de position (inclinaison du palpeur)[2)] 0 = déterminer écart de position du palpeur[6)] 1 = ne pas déterminer d'écart de position
				CENTAINES DE MILLIERS : calibrage en parallèle à l'axe ou sous angle 0 = calibrage en parallèle à l'axe dans le SCP actif 1 = calibrage sous angle[7)]
				MILLIONS : détermination longueur d'outil lors du calibrage sur surface ou sur sphère 0 = aucune détermination de la longueur d'outil 1 = détermination de la longueur d'outil [4)] 2 = calibrage de l'axe d'approche sur la sphère, détermination de la longueur d'outil, saisie de la différence de mesure des longueurs d'outil dans les données de calibrage
2	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
3		S_SETV	Valeur de consigne	
4	Z0	S_SETV0	Valeur de consigne de la référence de longueur lors du calibrage Sphère	
5	X / Y / Z	S_MA	Axe de mesure (numéro de l'axe)[2), 6)] (par défaut=1)	
			Valeurs :	1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z)

N°	Paramètre de masque	Paramètre de cycle	Signification	
6	+ -	S_MD	Direction de mesure[2), 6)]	
			Valeurs :	0 = positif 1 = négatif
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	VMS	S_VMS	Vitesse de mesure variable lors du calibrage [2)]	
10	α	S_STA1	Angle de départ[2), 5)]	
11	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (par défaut=1)	
12	X0	S_SETV1	Point de référence d'arête pour calibrage entre 2 arêtes [3)]	
13		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
14		_AMODE	Mode alternatif	

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Pour le calibrage "Rayon dans la bague de calibrage", le diamètre et le centre de la bague doivent être connus (4 directions de mesure).
Pour le calibrage "Rayon sur deux arêtes", uniquement la distance des arêtes en direction de l'axe de mesure doit être connu (2 directions de mesure).
Pour le calibrage "Rayon sur une arête", la valeur de consigne de la surface doit être connue.

4) Variante de mesure : calibrage uniquement sur une surface (longueur sur surface), longueur d'outil corrigée résulte de S_MD et S_MA.

5) Uniquement pour variante de mesure "Bague de calibrage, ... et centre connu" (S_MVAR=1xxx02).

6) Axe de mesure uniquement pour variante de mesure S_MVAR=0 ou =xx1x01 ou =xx2x01 ou =20000
Variante de mesure : "Calibrage sur une surface" → Sélection de l'axe de mesure et du sens de la mesure
ou sur "Bague de calibrage, ... et centre connu" → Sélection d'un sens d'axe et sélection de l'axe de mesure et du sens de la mesure
ou sur "Bague de calibrage, ... et centre connu" → Sélection de deux sens d'axe et sélection de l'axe de mesure
ou "Détermination de la longueur du palpeur" → S_MA=3 → 3e axe du plan (pour G17 Z)

7) Variante de mesure uniquement calibrage dans la bague de calibrage ou sur la sphère de calibrage
Pour "Calibrage sur sphère de calibrage", la sphère est contournée à l'équateur lors de la mesure sous angle.

8) Pour calibrage "Rayon dans bague de calibrage" avec centre inconnu quatre directions de mesure dans le plan (pour G17 +-X +-Y).
Pour calibrage "Longueur sur surface" dans le sens moins de l'axe d'outil (pour G17 -Z).

Paramètres des cycles de mesure CYCLE978

```
PROC CYCLE978(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL
S_TSA,INT S_MA,INT S_MD,INT S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL S_TZL,REAL
S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT
_AMODE,INT _DP)
```

Tableau 3-28 Paramètres d'appel CYCLE978[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Va-leurs :	UNITES : élément de contour 0 = mesures surface
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM) 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : réservé
				DIZAINES DE MILLIERS : mesure avec/sans pivotement de broche ou orientation de palpeur dans le sens de commutation[9)] 0 = mesure sans pivotement de broche, sans orientation du palpeur 1 = mesure avec pivotement de broche 2 = orientation du palpeur dans le sens de commutation
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Va-leurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2)]	
			Va-leurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction entre 1 et au maximum 999 numéro D (numéro de tranchant) pour correction d'outil pour correction totale et en réglage voir également S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32 000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction dans géométrie d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction de l'usure du rayon d'outil) 1 = correction d'outil correction totale (CT)[5)] La valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + valeur de correction CT (ancienne), CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] La valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale, non inversée) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icon+quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
5	X0	S_SETV	Valeur de consigne	
6	DFA	S_FA	Course de mesure	
7	TSA	S_TSA	Zone de fiabilité	
8	X	S_MA	Numéro de l'axe de mesure[7)] (plage de valeurs de 1 à 3)	
			Va-leurs :	1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z) Mesure dans le sens de l'outil
9		S_MD	Direction de mesure de l'axe de mesure	
			Va-leurs :	1 = direction de mesure positive 2 = direction de mesure négative
10	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (plage de valeurs de 1 à 9)	

N°	Paramètre de masque	Paramètre de cycle	Signification	
11	TR	S_TNAME	Nom d'outil[3)]	
12	DL	S_DLNUM	Correction totale / en réglage numéro DL[5)]	
13	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	
14	TZL	S_TZL	Correction zéro[2), 3)]	
15	DIF	S_TDIF	Contrôle d'écart dimensionnel[2), 3)]	
16	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[3)]	
17	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[3)]	
18	TMV	S_TMV	Plage de correction lors du calcul de moyenne[2)]	
19	FW	S_K	Facteur de pondération pour calcul de moyenne[2)]	
20	EVN	S_EVNUM	Bloc de données mémoire de valeurs empiriques[2), 8)]	
21		S_MCBIT	réservé	
22		_DMODE	Mode display	
			Va-leurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
23		_AMODE	Mode alternatif	
			Va-leurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

1) Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Uniquement pour correction dans outil, autrement paramètre = ""

4) Uniquement pour correction dans outil et si utilisation d'une tolérance dimensionnelle, autrement paramètre = 0

5) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR . De plus, il faut que le paramètre machine général MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK bit8=1.

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

7) Correction dans géométrie d'outil :
- pour mesure dans le plan (S_MA=1 ou S_MA=2) correction dans rayon d'outil
- pour mesure dans le sens de l'outil (S_MA=3) Correction de longueur d'outil L1

8) Calcul de moyenne empirique possible pour correction d'outil et correction dans DO
Plage de valeurs de la mémoire des valeurs moyennes empiriques :
De 1 à 20, numéro de la mémoire des valeurs empiriques, voir donnée de réglage propre au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
De 10 000 à 200 000 numéro de la mémoire des valeurs moyennes, voir donnée de réglage propre au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

9) Lors de la mesure avec pivotement de broche, le rayon/le diamètre du palpeur doit être déterminé avec précision avec une variante de calibrage du CYCLE976 Rayon sur anneau ou sur arête ou sur sphère. Sinon le résultat de mesure est faussé.

Paramètres des cycles de mesure CYCLE998

```
PROC CYCLE998(INT S_MVAR,INT S_KNUM,INT S_RA,INT S_PRNUM,REAL S_SETV,REAL S_STA1,REAL
S_INCA,REAL S_FA,REAL S_TSA,INT S_MA,INT S_MD,REAL S_ID,REAL S_SETV0,REAL S_SETV1,REAL
S_SETV2,REAL S_SETV3,INT S_NMSP,INT S_EVNUM,INT _DMODE,INT _AMODE)
```

Tableau 3-29 Paramètres d'appel CYCLE998 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=5)	
			Valeurs :	UNITES : élément de contour 5 = mesure arête (un angle) 6 = mesurer plan (deux angles)
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer et pas de correction du DO 1 = mesure, détermination et correction du DO (voir S_KNUM)
				MILLIERS : zone de protection 0 = aucune zone de protection prise en compte 1 = prise en compte d'une zone de protection
				DIZAINES DE MILLIERS : mesurer avec pivotement de broche (mesure différentielle) 0 = mesure sans pivotement de broche 1 = mesure avec pivotement de broche
				CENTAINES DE MILLIERS : mesure sous angle ou en parallèle à l'axe 0 = mesure sous angle 1 = mesure en parallèle à l'axe
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base [3)] 0 = correction précise 1 = correction approximative

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	A, B, C	S_RA	Cible de correction Rotation de coordonnées ou axe rotatif [14]	
			Valeurs :	0 = cible de correction Rotation de coordonnées autour de l'axe résultant du paramètre S_MA [4] >0 = cible de correction Axe rotatif. Numéro de l'axe de canal de l'axe rotatif (de préférence table rotative). La correction d'angle est effectuée dans la partie de translation du DO de l'axe rotatif.
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (par défaut=1)	
5	DX / DY / DZ	S_SETV	Distance (incrémentale) de la position de départ au point de mesure P1 de l'axe de mesure (S_MA) [5]	
6	α	S_STA1	Valeur de consigne d'angle pour "Orienter arête" ou pour "Orienter plan" autour du 1er axe du plan (pour G17 X) [9]	
7	β	S_INCA	Valeur de consigne d'angle pour "Orienter plan" autour du 2e axe du plan (pour G17 Y) [9]	
8	DFA	S_FA	Course de mesure	
9	TSA	S_TSA	Zone de fiabilité Surveillance de l'écart d'angle par rapport à la valeur de consigne d'angle [degré] [6]	
10	X / Y / Z	S_MA	Axe de mesure, axe de décalage [7] (par défaut=201)	
			Valeurs :	UNITES : numéro de l'axe de mesure 1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z)
				DIZAINES : réservé
				CENTAINES : numéro de l'axe de décalage 1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z)
11	+-	S_MD	Direction de mesure de l'axe de mesure [8]	
			Valeurs :	0 = direction de mesure est déterminée à partir de la valeur de consigne et de la position réelle de l'axe de mesure (compatibilité) 1 = direction de mesure positive 2 = direction de mesure négative
12	L2	S_ID	Pour variante de mesure "Orienter arête" : Distance (incrémentale) entre les points de mesure P1 et P2 dans l'axe de décalage (valeur >0) Pour variante de mesure "Orienter plan", les paramètres valables sont énumérés ci-après.	
13	L2	S_SETV0	Distance entre les points de mesure P1 et P2 dans le 1er axe du plan [10]	
14		S_SETV1	Distance entre les points de mesure P1 et P2 dans le 2e axe du plan [11], [12]	
15	L3x	S_SETV2	Distance entre les points de mesure P1 et P3 dans le 1er axe du plan [11]	
16	L3y	S_SETV3	Distance entre les points de mesure P1 et P3 dans le 2e axe du plan [10]	
17	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (par défaut=1)	
18		S_EVNUM	Bloc de données mémoire de valeurs empiriques [2], [13]	

N°	Paramètre de masque	Paramètre de cycle	Signification	
19		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
20		_AMODE	réservé (mode alternatif)	

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) DO "de précision" uniquement si cible de correction Axe rotatif et MD 52207 $MCS_AXIS_USAGE_ATTRIB[n] bit6=1.
Si le DO n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO.

4) Exemple pour correction dans rotation de coordonnées : S_MA=102 axe de mesure Y axe de décalage X résulte en rotation de coordonnées autour de Z (pour G17)

5) Valeur pertinente uniquement si utilisation d'une zone de protection (S_MVAR Position des MILLIERS = 1)

6) Les angles des paramètres S_STA1 et S_TSA sont additionnés lors du positionnement du point de mesure P1 par rapport au point de mesure P2 dans l'axe de décalage.

7) Le numéro de l'axe de mesure ne doit pas être identique au numéro de l'axe de décalage (par exemple 101 pas autorisé)

8) Sens de mesure uniquement "Orienter arête" et "Mesurer en parallèle à l'axe (S_MVAR=10x105)

9) Plage d'angle S_STA1 ±45 degrés pour "Orienter arête"
Plage d'angle S_STA1 0 à +60 degrés et S_INCA ±30 degrés pour "Orienter plan"

10) Pour variante de mesure "Orienter plan" et "Orienter arête"

11) Pour variante de mesure "Mesurer plan" et "Mesurer en parallèle à l'axe"

12) Pas pour version de cycle de mesure SW04.04.

13) Calcul de valeur empirique pour la correction dans DO ; plage de valeurs de la mémoire des valeurs moyennes empiriques : De 1 à 20, numéro de la mémoire de valeurs empiriques, voir donnée de réglage propre au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]

14) Le paramètre S_RA n'est pertinent que pour la variante de mesure "Orienter arête" (S_MVAR xxxxx5).

Paramètres des cycles de mesure CYCLE977

```
PROC CYCLE977(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,REAL S_SETV0,REAL
S_SETV1,REAL S_FA,REAL S_TSA,REAL S_STA1,REAL S_ID,REAL S_SZA,REAL S_SZO,INT S_MA,INT
S_NMSP,STRING[32] S_TNAME,INT S_DLNUM,REAL S_TZL,REAL S_TDIF,REAL S_TUL,REAL S_TLL,REAL
S_TMV,INT S_K,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT _AMODE,REAL S_XM,REAL_S_YM,INT _DP)
```

Tableau 3-30 Paramètres d'appel CYCLE977[1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : élément de contour (plage de valeurs de 1 à 6) 1 = mesure alésage 2 = mesure tourillon (arbre) 3 = mesure rainure 4 = mesure âme 5 = mesure rectangle, intérieur 6 = mesure rectangle, extérieur
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou. pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM) 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : zone de protection 0 = aucune zone de protection prise en compte 1 = prise en compte d'une zone de protection
				DIZAINES DE MILLIERS : mesure avec/sans pivotement de broche (mesure différentielle) ou orientation du palpeur dans le sens de commutation 0 = mesure sans pivotement de broche, sans orientation du palpeur 1 = mesure avec pivotement de broche 2 = orientation du palpeur dans le sens de commutation
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2)]	
			Valeurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction entre 1 et au maximum 999 numéro D (numéro de tranchant) pour correction d'outil ; pour correction totale et en réglage, voir aussi S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32 000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction rayon d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction de l'usure du rayon d'outil) 1 = correction d'outil correction totale (CT)[5)] La valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + CT (ancienne) valeur de correction, CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] La valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale, non inversée) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
5	X0	S_SETV	Valeur de consigne	
6	X0	S_SETV0	Valeur de consigne pour rectangle 1er axe du plan (X pour G17)	
7	Y0	S_SETV1	Valeur de consigne pour rectangle 2e axe du plan (Y pour G17)	
8	XM	S_XM	Spécification de valeur de consigne pour le centre axe géométrique X	
9	YM	S_YM	Spécification de valeur de consigne pour le centre axe géométrique Y	
10	DFA	S_FA	Course de mesure	
11	TSA	S_TSA	Zone de fiabilité	
12	α 0	S_STA1	Angle de départ	

N°	Paramètre de masque	Paramètre de cycle	Signification	
13	DZ	S_ID	Valeur relative 1. Approche relative du 3e axe du plan (Z pour G17) Direction d'approche via signes de S_ID. Pour Mesure tourillon, âme et rectangle, la descente à la hauteur de mesure est définie avec S_ID . 2. Prise en considération d'une zone de protection Pour Mesure alésage, rainure et rectangle intérieur et avec une zone de protection, la hauteur de dépassement est définie avec S_ID .	
14	X1	S_SZA	Diamètre ou longueur (largeur) de la zone de protection[7)]	
15	Y1	S_SZO	Pour mesure rectangle : largeur de la zone de protection du 2e axe du plan	
16	X	S_MA	Numéro de l'axe de mesure[7)] (uniquement pour Mesure rainure ou âme)	
			Valeurs :	1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y)
17	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	
18	Mesures	S_NMSP	Nombre de mesures au même endroit [2)] (plage de valeurs de 1 à 9)	
19	TR	S_TNAME	Nom d'outil[2)]	
20	DL	S_DLNUM	Correction totale / en réglage numéro DL[5)]	
21	TZL	S_TZL	Correction zéro[2), 4)]	
22	DIF	S_TDIF	Contrôle d'écart dimensionnel[2), 4)]	
23	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[4)]	
24	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[4)]	
25	TMV	S_TMV	Plage de correction lors du calcul de moyenne[2)]	
26	FW	S_K	Facteur de pondération pour calcul de moyenne[2)]	
27		S_EVNUM	Bloc de données mémoire de moyennes empiriques[2), 8)]	
28		S_MCBIT	réservé	
29		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
30		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

1) Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Uniquement pour correction dans outil, autrement paramètre = ""

4) Uniquement pour correction dans outil et si utilisation d'une tolérance dimensionnelle, autrement paramètre = 0

5) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR . De plus, il faut que le paramètre machine général MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK bit8=1.

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

[7] Diamètre ou largeur de la zone de protection au sein de l'alésage ou de la rainure.
Diamètre resp. largeur de la zone de protection en dehors du tourillon ou de l'âme

[8] Calcul de moyenne empirique possible pour correction d'outil
Plage de valeurs de la mémoire de moyennes empiriques :
entre 1 et 20 numéro (n) de la mémoire de valeurs empiriques, voir donnée de réglage spécifique au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
Entre 10 000 et 200 000 numéro (n) de la mémoire de valeurs moyennes, voir donnée de réglage spécifique au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

Paramètres des cycles de mesure CYCLE961

```
PROC CYCLE961(INT S_MVAR,INT S_KNUM,INT S_PRNUM,REAL S_SETV0,REAL S_SETV1,REAL
S_SETV2,REAL S_SETV3,REAL S_SETV4,REAL S_SETV5,REAL S_SETV6,REAL S_SETV7,REAL S_SETV8,REAL
S_SETV9,REAL S_STA1,REAL S_INCA,REAL S_ID,REAL S_FA,REAL S_TSA,INT S_NMSP,INT S_MCBIT,INT
_DMODE,INT _AMODE)
```

Tableau 3-31 Paramètres d'appel CYCLE961 [1]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut ≥ 6)	
			Valeurs :	UNITES : élément de contour 5 = configuration angle intérieur droit, définition de valeur de consigne angle et distances A1 à A3 6 = configuration angle extérieur droit, définition de valeur de consigne angle et distances A1 à A3 7 = configuration angle intérieur, définition angle et distances A1 à A4 8 = configuration angle extérieur, définition angle et distances A1 à A4
				DIZAINES : définition de valeur de consigne sous forme de distance ou par quatre points 0 = valeur de consigne est définie sous forme de distance (polaire) 1 = valeur de consigne est définie par quatre points (points de mesure P1 à P4)
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou pas de correction d'outil) 1 = mesure, détermination et correction du DO, voir S_KNUM
				MILLIERS : zone de protection 0 = aucune prise en considération d'une zone de protection (obstacle) 1 = prise en considération d'une zone de protection (obstacle), voir S_ID
				DIZAINES DE MILLIERS : position de l'angle dans le SCP 0 = la position de l'angle est déterminée via le paramètre S_STA1 (compatibilité) 1 = position 1 de l'angle dans le point de départ positionné de la mesure [6] 2 = position 2 de l'angle, distances dans le 1er axe du plan (pour G17 X) sont négatives (voir S_SETV0, S_SETV1) 3 = position 3 de l'angle, distances dans le 1er et 2e axe du plan (pour G17 XY) sont négatives (voir S_SETV0 à S_SETV3) 4 = position 4 de l'angle, distances dans le 2e axe du plan (pour G17 Y) sont négatives (voir S_SETV2, S_SETV3)

N°	Paramètre de masque	Paramètre de cycle	Signification	
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2]	
			Va-leurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 9 = correction DO actif ou pour G500 dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[5] 1 = correction grossière
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
4	L1/X1	S_SETV0	Distance L1 entre le pôle et le point de mesure P1 en direction du 1er axe du plan (pour G17 X)[3] (si la distance L1=0, L1 = M_SETV1 / 2 est calculé automatiquement) ou point de départ P1x du 1er axe du plan (pour G17 X) [4]	
5	L2/Y1	S_SETV1	Distance L2 entre le pôle et le point de mesure P2 en direction du 1er axe du plan [3] ou point de départ P1y du 2e axe du plan (pour G17 Y) [4]	
6	L3/X2	S_SETV2	Distance L3 entre le pôle et le point de mesure P3 en direction du 2e axe du plan [3] (si la distance L3=0, L3 = M_SETV3 / 2 est calculé automatiquement lorsque l'angle n'est pas droit) ou point de départ P2x du 1er axe du plan [4]	
7	L4/Y2	S_SETV3	Distance L4 entre le pôle et le point de mesure P3 en direction du 2e axe du plan lorsque l'angle n'est pas droit [3] ou point de départ P2y du 2e axe du plan [4]	
8	XP/X3	S_SETV4	Position du pôle dans le 1er axe du plan [3] ou point de départ P3x du 1er axe du plan [4]	
9	XP/Y3	S_SETV5	Position du pôle dans le 2e axe du plan [3] ou point de départ P3x du 2e axe du plan [4]	
10	X4	S_SETV6	Point de départ P4x du 1er axe du plan [4]	
11	Y4	S_SETV7	Point de départ P4y du 2e axe du plan [4]	
12	X0	S_SETV8	Valeur de consigne de l'angle mesuré dans le 1er axe du plan lors de la correction dans DO	
13	Y0	S_SETV9	Valeur de consigne de l'angle mesuré dans le 2e axe du plan lors de la correction dans DO	
14	α0	S_STA1	Angle de départ depuis direction positive du 1er axe du plan vers l'arête de référence de la pièce dans le SCM (+-270 degrés)	
15	α1	S_INCA	Angle entre arêtes de référence de pièce lors de la mesure d'un angle pas droit [7]	
16	DZ	S_ID	Valeur d'approche à hauteur de mesure pour chaque point de mesure lorsqu'une zone de protection est activée (voir S_MVAR).	
17	DFA	S_FA	Course de mesure	
18	TSA	S_TSA	Zone de fiabilité Surveillance de l'écart d'angle par rapport à la valeur de consigne d'angle [degré]	

N°	Paramètre de masque	Paramètre de cycle	Signification	
19	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (plage de valeurs de 1 à 9) [2]	
20		S_MCBIT	réservé	
21		_DMODE	Mode display	
			Va-leurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
22		_AMODE	Mode alternatif	

[1] Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

[2] Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

[3] Définition des points de mesure en coordonnées polaires tenant compte de l'angle de départ S_STA1 pour le point de mesure 3 ou 4 de l'incrément angulaire S_INCA

[4] Définition des points de mesure dans le système de coordonnées cartésiennes (au moyen de 4 points)

[5] Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO.

[6] Plage de valeurs Angle S_INCA : -180 à +180 degrés

[7] Angle de départ S_STA1, plage de valeurs : angle droit : +- 90 degrés, angle quelconque : +- 45 degrés

Paramètres des cycles de mesure CYCLE979

```
PROC CYCLE979(INT S_MVAR,INT S_KNUM,INT S_KNUM1,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL
S_TSA,REAL S_CPA,REAL S_CPO,REAL S_STA1,REAL S_INCA,INT S_NMSP,STRING[32] S_TNAME,REAL
S_DLNUM,REAL S_TZL,REAL S_TDIF,REAL S_TUL,REAL S_TLL,REAL S_TMV,INT S_K,INT S_EVNUM,INT
S_MCBIT,INT _DMODE,INT _AMODE,REAL S_XM,REAL S_YM,INT _DP)
```

Tableau 3-32 Paramètres d'appel CYCLE979[0)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : élément de contour 1 = mesure alésage 2 = mesure tourillon (arbre)
				DIZAINES : réservé
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO ou. pas de correction d'outil) 1 = mesure, détermination et correction du DO (voir S_KNUM) 2 = mesure et correction d'outil (voir S_KNUM1)
				MILLIERS : nombre de points de mesure 0 = 3 points de mesure 1 = 4 points de mesure
				DIZAINES DE MILLIERS : mesure avec/sans pivotement de broche (mesure différentielle) ou orientation du palpeur dans le sens de commutation 0 = mesure sans pivotement de broche, sans orientation du palpeur 1 = mesure avec pivotement de broche 2 = orientation du palpeur dans le sens de commutation
2	Sélection	S_KNUM	Correction décalage d'origine (DO) ou DO de base ou référence de base [2)]	
			Valeurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et 99 max., numéro du décalage d'origine ou entre 1 et 16 au maximum, numéro du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction DO ou base ou référence de base 0 = correction DO réglable 1 = correction DO de base propre au canal 2 = correction référence de base 3 = correction DO de base global 9 = correction DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction grossière ou fine dans DO, DO de base ou référence de base 0 = correction fine[6)] 1 = correction grossière

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	Sélection	S_KNUM1	Correction dans correction d'outil[2)]	
			Va-leurs :	UNITES :
				DIZAINES :
				CENTAINES : 0 = aucune correction entre 1 et au maximum 999 numéro D (numéro de tranchant) pour correction d'outil ; pour correction totale et en réglage, voir aussi S_DLNUM
				MILLIERS : 0 ou numéros D univoques
				DIZAINES DE MILLIERS : 0 ou numéros D univoques 1 jusqu'à 32 000 max., lorsque des numéros D univoques sont configurés dans les PM
				CENTAINES DE MILLIERS : correction d'outil[2)] 0 = aucune donnée (correction dans rayon d'outil) 1 = correction longueur L1 2 = correction longueur L2 3 = correction longueur L3 4 = correction rayon
				MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction de l'usure du rayon d'outil) 1 = correction d'outil correction totale (CT)[5)] La valeur de correction d'outil est ajoutée à la CT existante 2 = correction d'outil correction en réglage (CR)[5)] CR (nouvelle) = CR (ancienne) + valeur de correction CT (ancienne), CT (nouvelle) = 0 3 = correction d'outil correction en réglage (CR)[5)] La valeur de correction d'outil est ajoutée à la CR existante 4 = correction d'outil géométrie
				DIZAINES DE MILLIONS : correction d'outil[2)] 0 = aucune donnée (correction dans la géométrie d'outil normale, non inversée) 1 = correction inversée
				CENTAINES DE MILLIONS : correction d'outil 0 = correction d'outil sans outil frère 1 = correction d'outil dans l'outil frère (_DP)
4	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
5	X0	S_SETV	Valeur de consigne	
6	DFA	S_FA	Course de mesure	
7	TSA	S_TSA	Zone de fiabilité	
8	X0	S_CPA	Centre du 1er axe du plan (pour G17 X)	
9	Y0	S_CPO	Centre du 2e axe du plan (pour G17 Y)	
10	XM	S_XM	Spécification de valeur de consigne pour le centre axe géométrique X	
11	YM	S_YM	Spécification de valeur de consigne pour le centre axe géométrique Y	
12	alpha 0	S_STA1	Angle de départ[7)]	
13	alpha 1	S_INCA	Incrément angulaire[8)]	
14	Mesures	S_NMSP	Nombre de mesures au même endroit [1)] (plage de valeurs de 1 à 9)	
15	ST	_DP	Numéro de l'outil frère à corriger (numéro frère)	

N°	Paramètre de masque	Paramètre de cycle	Signification	
16	T	S_TNAME	Nom d'outil[2)]	
17	DL	S_DLNUM	Correction totale / en réglage numéro DL[1), 4)]	
18	TZL	S_TZL	Correction zéro[1), 2)]	
19	DIF	S_TDIF	Contrôle d'écart dimensionnel[1), 2)]	
20	TUL	S_TUL	Limite supérieure de tolérance (relative à la consigne)[2)]	
21	TLL	S_TLL	Limite inférieure de tolérance (relative à la consigne)[2)]	
22	TMV	S_TMV	Plage de correction lors du calcul de moyenne[1)]	
23	FW	S_K	Facteur de pondération pour calcul de moyenne[1)]	
24		S_EVNUM	Bloc de données mémoire de valeurs empiriques[1), 6)]	
25		S_MCBIT	réservé	
26		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
27		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

0) Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

1) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

2) Uniquement pour correction dans outil, autrement paramètre = ""

3) Uniquement pour correction dans outil et si utilisation d'une tolérance dimensionnelle, autrement paramètre = 0

4) Uniquement si la fonction "Correction totale / en réglage" est configurée dans le paramètre machine général MD 18108 $MN_MM_NUM_SUMCORR .

5) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO.

6) Calcul de moyenne empirique uniquement possible pour correction d'outil
Plage de valeurs de la mémoire de valeurs moyennes empiriques :
De 1 à 20, numéro de la mémoire de valeurs empiriques, voir donnée de réglage spécifique au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
De 10 000 à 200 000 numéro de la mémoire de valeurs moyennes, voir donnée de réglage spécifique au canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

7) Plage de valeurs Angle de départ -360 à +360 degrés

8) Plage de valeur Incrément angulaire >0 à ≤90 degrés pour 4 points de mesure resp. >0 à ≤120 degrés pour 3 points de mesure.

Paramètres de cycles de mesure CYCLE997

```
PROC CYCLE997 (INT S_MVAR,INT S_KNUM,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL S_TSA,REAL
S_STA1,REAL S_INCA,REAL S_SETV0,REAL S_SETV1,REAL S_SETV2,REAL S_SETV3,REAL S_SETV4,REAL
S_SETV5,REAL S_SETV6,REAL S_SETV7,REAL S_SETV8,REAL S_TNVL,INT S_NMSP,INT S_MCBIT,INT
_DMODE,INT _AMODE)
```

Tableau 3-33 Paramètres d'appel CYCLE997 [1), 2)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=9)	
			Va-leurs :	UNITES : élément de contour 9 = mesurer sphère
				DIZAINES : répétition de la mesure 0 = sans répétition de la mesure 1 = avec répétition de la mesure
				CENTAINES : destination de correction 0 = seulement mesurer (pas de correction du DO) 1 = mesure, détermination et correction du DO (voir S_KNUM)
				MILLIERS : stratégie de mesure 0 = mesure en parallèle à l'axe, sans angle de départ, orientation du palpeur selon SD55740, bit 1 1 = mesure le long, avec angle de départ, orientation du palpeur selon SD55740, bit 1 2 = mesure le long, avec angle de départ, orienter le palpeur dans le sens de commutation 3 = mesure en parallèle à l'axe, avec angle de départ, orientation du palpeur selon SD55740, bit 1 4 = mesure en parallèle à l'axe, avec angle de départ, orienter le palpeur dans le sens de commutation
				DIZAINES DE MILLIERS : nombre des sphères à mesurer 0 = mesurer une sphère 1 = mesurer trois sphères
				CENTAINES DE MILLIERS : nombre des points de mesure, uniquement pour mesure sous angle (tenir compte de la stratégie de mesure : position des MILLIERS > 0) 0 = trois points de mesure pour la mesure sous angle (le long de la sphère) 1 = quatre points de mesure lors de la mesure sous angle (le long de la sphère)
				MILLIONS : détermination du diamètre de consigne de la sphère 0 = aucune détermination du diamètre de consigne de la sphère 1 = détermination du diamètre de consigne de la sphère

N°	Paramètre de masque	Paramètre de cycle	Signification	
2	Sélection	S_KNUM	Correction dans décalage d'origine (DO) ou base ou référence de base [3]	
			Va-leurs :	UNITES :
				DIZAINES : 0 = aucune correction entre 1 et au maximum 99 numéros du décalage d'origine ou entre 1 et au maximum 16 numéros du décalage de base
				CENTAINES : réservé
				MILLIERS : Correction dans DO ou DO de base ou référence de base 0 = correction dans DO réglable 1 = correction dans DO de base propre au canal 2 = correction dans référence de base 3 = correction dans DO de base global [7] 9 = correction dans DO actif ou pour G500 dans le dernier DO de base propre au canal actif
				DIZAINES DE MILLIERS : Correction approximative ou précise dans DO ou DO de base ou référence de base 0 = correction fine[6] 1 = correction grossière
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
4		S_SETV	Diamètre de la/des sphère(s) [4]	
5	DFA	S_FA	Course de mesure	
6	TSA	S_TSA	Zone de fiabilité	
7	alpha 0	S_STA1	Angle de départ pour la mesure sous angle	
8	alpha 1	S_INCA	Incrément angulaire pour la mesure sous angle	
9	X1	S_SETV0	Position de consigne de la 1re sphère du 1er axe du plan (pour G17 X) pour la mesure de 3 sphères	
10	Y1	S_SETV1	Position de consigne de la 1re sphère du 2e axe du plan (pour G17 Y) pour la mesure de 3 sphères	
11	Z1	S_SETV2	Position de consigne de la 1re sphère du 3e axe du plan (pour G17 Z) pour la mesure de 3 sphères	
12	X2	S_SETV3	Position de consigne de la 2e sphère du 1er axe du plan pour la mesure de 3 sphères	
13	Y2	S_SETV4	Position de consigne de la 2e sphère du 2e axe du plan pour la mesure de 3 sphères	
14	Z2	S_SETV5	Position de consigne de la 2e sphère du 3e axe du plan pour la mesure de 3 sphères	
15	X3	S_SETV6	Position de consigne de la 3e sphère du 1er axe du plan pour la mesure de 3 sphères	
16	Y3	S_SETV7	Position de consigne de la 3e sphère du 2e axe du plan pour la mesure de 3 sphères	
17	Z3	S_SETV8	Position de consigne de la 3e sphère du 3e axe du plan pour la mesure de 3 sphères	
18	TVL	S_TNVL	Valeur limite pour la distorsion du triangle (somme des écarts) pour la mesure de 3 sphères [5]	
19	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (plage de valeurs de 1 à 9)	
20		S_MCBIT	réservé	

N°	Paramètre de masque	Paramètre de cycle	Signification	
21		_DMODE	Mode display	
			Va-leurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
22		_AMODE	Mode alternatif	

1) Toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

2) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Positionnement intermédiaire avec rotation autour de la sphère à l'équateur

4) Mesure de 3 sphères : le même diamètre de consigne (_SETV) est valable pour toutes les sphères

5) Valeur par défaut pour S_TNVL=1.2
Correction dans DO : La correction dans DO est uniquement effectuée si la distorsion déterminée est inférieure à la valeur limite S_TNVL .

6) Si le DO "de précision" n'est pas configuré dans les PM, la correction est effectuée "approximativement" après le DO

7) Pour la variante de mesure "Mesurer trois sphères", une correction dans un frame de base global est impossible (S_KNUM = 3001 bis 3016) étant donné que le frame ne possède pas de composants de rotation.

Paramètres du cycle de mesure CYCLE995

```
PROC CYCLE995 (INT S_MVAR,INT S_KNUM,INT S_PRNUM,REAL S_SETV,REAL S_FA,REAL S_TSA,REAL
S_STA1,REAL S_INCA,REAL S_DZ,REAL S_SETV0,REAL S_SETV1,REAL S_SETV2,REAL S_TUL,REAL
S_TZL,INT S_NMSP,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-34 Paramètres d'appel CYCLE995 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=5)	
			Valeurs :	UNITES : élément de contour 5 = Géométrie de broche (parallélisme à l'axe d'outil)
				DIZAINES : répétition de la mesure 1 = avec répétition de mesure
				CENTAINES : aucune cible de correction 0 = mesure seule
				MILLIERS : stratégie de mesure 2 = mesure sous angle, alignement du palpeur dans le sens de commutation
				DIZAINES DE MILLIERS : nombre des sphères à mesurer 0 = mesurer une sphère
				CENTAINES DE MILLIERS : nombre de points de mesure 1 = 4 points de mesure lors de la mesure sous angle (rotation autour de la sphère)
				MILLIONS : détermination du diamètre de consigne de la sphère 0 = aucune détermination du diamètre de consigne de la sphère 1 = détermination du diamètre de consigne de la sphère
2	Sélection	S_KNUM	destination de correction 0 = 0	
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (plage de valeurs de 1 à 40)	
4	DM	S_SETV	Diamètre de la sphère de calibrage [4)]	
5	DFA	S_FA	Course de mesure	
6	TSA	S_TSA	Zone de fiabilité [5)]	
7	alpha 0	S_STA1	Angle de départ pour la mesure sous angle [3)]	
8		S_INCA	Incrément angulaire pour la mesure sous angle [2)]	
9	DZ	S_DZ	Écart entre la 1re mesure P1 et la 2e mesure P2 sur l'axe du palpeur	
10		S_SETV0	Position de consigne de la sphère, 1er axe du plan (pour G17 X) [2)]	
11		S_SETV1	Position de consigne de la sphère, 2ème axe du plan (pour G17 Y) [2)]	
12		S_SETV2	Position de consigne de la sphère, 3ème axe du plan (pour G17 Z) [2)]	
13	TUL	S_TUL	Limite supérieure de tolérance du décalage angulaire	
14	TZL	S_TZL	Plage de correction zéro[1), 4)]	
15	Nombre	S_NMSP	Nombre de mesures au même endroit [2)] (plage de valeurs de 1 à 9)	
16		S_MCBIT	réservé [2)]	
17		_DMODE	Mode display	
			Valeurs :	UNITES : Plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)

N°	Paramètre de masque	Paramètre de cycle	Signification	
18		_AMODE	Mode alternatif	
			Valeurs :	UNITES : tolérance dimensionnelle oui/non 0 = non 1 = oui

toutes les valeurs par défaut = 0 ou caractérisées comme plage de valeur de a à b

1) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

2) Les paramètres ne sont pas utilisés actuellement et ils ne sont pas affichés dans le masque de saisie.
Le paramètre Incrément angulaire S_INCA est préréglé sur 90 degrés.

3) Plage de valeurs Angle de départ -360 à +360 degrés

4) pour tolérance dimensionnelle "oui" :
Si les angles mesurés sont inférieurs à la valeur de la plage de correction zéro TZL, les paramètres de résultat se rapportant aux angles (_OVR[2], _OVR[3]) et aux écarts (_OVR[7], _OVR[8]) sont mis à zéro.
L'affichage TZL se fait via les données de réglage générales SD54760 $SNS_MEA_FUNCTION_MASK_PIECE , bit 25=1.
(Sélection Débloquer correction zéro pour la mesure de l'angularité de broche)

5) Le paramètre TSA se rapporte à la 1ère mesure de la sphère de calibrage.

Paramètres des cycles de mesure CYCLE996

```
PROC CYCLE996(INT S_MVAR,INT S_TC,INT S_PRNUM,REAL S_SETV,REAL S_STA1,REAL S_SETV0,REAL
S_SETV1,REAL S_SETV2,REAL S_SETV3,REAL S_SETV4,REAL S_SETV5,REAL S_TNVL,REAL S_FA,REAL
S_TSA,INT S_NMSP,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-35 Paramètres d'appel CYCLE996 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure (par défaut=1)	
			Valeurs :	UNITES : ordre des points de mesure 0 = calcul cinématique (sélection avec : Affichage du résultat, journal, modification du bloc de données d'orientation, acquittement éventuel de l'utilisateur), voir _AMODE 1 = 1re mesure 2 = 2e mesure 3 = 3e mesure
				DIZAINES : réservé 0 = 0
				CENTAINES : Variante de mesure pour 1re à 3e mesure 0 = mesure de la sphère de calibrage en parallèle à l'axe 1 = mesure de la sphère de calibrage sous angle et pas de poursuite de la broche [3)] 2 = mesure de la sphère de calibrage et poursuite de la broche dans le sens de commutation du palpeur [3)] 3 = mesure en parallèle à l'axe, avec angle de départ [8)] 4 = mesure en parallèle à l'axe, poursuite de la broche dans le sens de commutation du palpeur [8)]
				MILLIERS : destination de correction pour le calcul de la cinématique [4)] 0 = mesure seule. Les blocs de données d'orientation sont calculés, mais restent inchangés. 1 = calcul du bloc de données d'orientation. Les blocs de données d'orientation sont modifiés, le cas échéant après acquittement par l'utilisateur. [4)]
				DIZAINES DE MILLIERS : axe de mesure (axe rotatif 1 ou 2) ou résultat vectoriel ouvert ou fermé pour Calcul de la cinématique 0 = résultat vectoriel fermé (uniquement pour Calcul de la cinématique) 1 = axe rotatif 1 (uniquement pour 1re à 3e mesure) 2 = axe rotatif 2 (uniquement pour 1re à 3e mesure) [5)] 3 = résultat vectoriel ouvert (uniquement pour Calcul de la cinématique)
				CENTAINES DE MILLIERS : normalisation axe rotatif 1 pour Calcul de la cinématique 0 = pas de normalisation axe rotatif 1 1 = normalisation dans la direction du 1er axe du plan (pour G17 X) 2 = Normalisation dans la direction du 2e axe du plan (pour G17 Y) 3 = Normalisation dans la direction du 3e axe du plan (pour G17 Z)
				MILLIONS : normalisation axe rotatif 2 pour Calcul de la cinématique [5)] 0 = pas de normalisation axe rotatif 2 1 = normalisation dans la direction du 1er axe du plan (pour G17 X) 2 = Normalisation dans la direction du 2e axe du plan (pour G17 Y) 3 = Normalisation dans la direction du 3e axe du plan (pour G17 Z)
				DIZAINES DE MILLIONS : fichiers journaux 0 = pas de fichier journal 1 = fichier journal contenant les vecteurs calculés (Toolcarrier) et la 1re transformation dynamique 5 axes (TRAORI(1)), si cela est configuré dans les MD.
2		S_TC	Numéro du bloc de données d'orientation (Toolcarrier)	

N°	Paramètre de masque	Paramètre de cycle	Signification	
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
4		S_SETV	Diamètre de la sphère de calibrage	
5	alpha 0	S_STA1	Angle de départ pour la mesure sous angle	
6	alpha 0	S_SETV0	Valeur de position axe rotatif 1 (si axe rotatif manuel ou semi-automatique)	
7	alpha 1	S_SETV1	Valeur de position axe rotatif 2 (si axe rotatif manuel ou semi-automatique) [6]	
8	XN	S_SETV2	Valeur de position pour normalisation axe rotatif 1	
9	XN	S_SETV3	Valeur de position pour normalisation axe rotatif 2 [6]	
10	delta	S_SETV4	Valeur de tolérance des vecteurs offset I1 à I4	
11	delta	S_SETV5	Valeur de tolérance des vecteurs d'axe rotatif V1 et V2	
12	TVL	S_TNVL	Valeur limite du segment angulaire de l'axe rotatif (plage de valeurs 1 à 60 degrés) (par défaut=20) [7]	
13	DFA	S_FA	Course de mesure	
14	TSA	S_TSA	Zone de fiabilité	
15	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (par défaut=1)	
16		S_MCBIT	réservé	
17		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
18		_AMODE	Mode alternatif	
			Valeurs :	UNITES : contrôle de tolérance oui/non 0 = non 1 = oui : Evaluation des valeurs de tolérance des vecteurs S_SETV4, S_SETV5
				DIZAINES : acquittement de l'utilisateur pour l'inscription des vecteurs calculés dans le bloc de données d'orientation [4] 0 = oui : l'utilisateur doit acquitter la modification. 1 = non : les vecteurs calculés sont immédiatement inscrits (uniquement lorsque les positions des CENTAINES et des MILLIERS = 0).
				CENTAINES : affichage du résultat [5] 0 = non 1 = oui
				MILLIERS : modification possible de l'affichage du résultat 0 = non 1 = oui, et modifiable (uniquement lorsque la position des CENTAINES = 1)

[1] Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

[2] Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

[3] Cette variante permet par exemple d'effectuer des mesures sur la sphère de calibrage lorsque la cinématique est positionnée à 90 degrés, sans collision avec la tige de fixation de la sphère de calibrage. Un angle de départ S_STA1 peut être défini de 0 à 360 degrés. L'incrément angulaire de rotation autour de la sphère est égal à 90 degrés.
La donnée de réglage spécifique au canal SD55634 $SCS_MEA_FEED_PLANE_VALUE est utilisée comme avance sur la trajectoire circulaire.

[4] une interrogation de l'utilisateur avec M0 a lieu avant l'inscription. Les vecteurs sont uniquement inscrits avec démarrage commande numérique.
Si le programme de mesure est interrompu avec RESET , aucun vecteur calculé n'est inscrit.
Les vecteurs sont uniquement inscrits si la tolérance des vecteurs offset n'est pas dépassée lors du calcul.

[5] Affichage du résultat uniquement avec la variante de mesure Calcul cinématique.
Si le résultat de la mesure doit également être affiché après les mesures 1 à 3, la donnée de réglage propre au canal SD55613 $SCS_MEA_RESULT_DISPLAY doit être activée.

[6] axe rotatif 2 uniquement pour cinématiques avec deux axes rotatifs

[7] Valeur limite segment angulaire de l'axe rotatif. Plage des valeurs de `S_TNVL` de 20 à 60 degrés. Les valeurs de `S_TNVL` < 20 degrés peuvent conduire à des imprécisions dues aux imprécisions de mesure du palpeur, de l'ordre de quelques micromètres. Un dépassement de la valeur limite déclenche le message d'erreur 61430, avec affichage de la valeur minimale pour la valeur limite.

[8] Poursuite de la broche dans le sens de commutation du palpeur si SD54760 bit 17 = 1

Paramètres de cycle de mesure CYCLE9960

```
PROC CYCLE9960(INT S_MVAR,STRING[40] S_TNAME,INT S_PRNUM,REAL S_SETV,REAL S_SETV1,REAL
S_START_RA1,REAL S_END_RA1,INT S_CMEA_RA1,REAL S_POS_RA2,REAL S_SETV2,REAL
S_START_RA2,REAL S_END_RA2,INT S_CMEA_RA2,REAL S_POS_RA1,REAL S_SETV4,REAL S_FA,REAL
S_TSA,INT S_NMSP,INT S_DMODE,INT S_AMODE,INT S_KNUM)
```

Tableau 3-36 Paramètres d'appel CYCLE9960 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1	Sélection	S_MVAR	Variante de mesure (par défaut=1)	
			Valeurs :	UNITES : Variante de mesure 0 = calcul et mesure de la cinématique (sélection avec : affichage du résultat, protocole, modification du bloc de données d'orientation (voir milliers S_MVAR)) 1 = mesurer la tête de référence 2 = adapter la tête à la tête de référence 3 = mesurer et calculer les points intermédiaires (E996)
				DIZAINES : Réservé
				CENTAINES : Variante de mesure avec la mesure de la sphère 2 = mesure de la sphère de calibrage et poursuite de la broche dans le sens de commutation du palpeur 4 = mesure en parallèle à l'axe, avec angle de départ, poursuite de la broche dans le sens de commutation du palpeur [2)]
				MILLIERS : Destination de correction pour le calcul de la cinématique [3)] 0 = mesurer et calculer. Les blocs de données sont calculés et restent inchangés 1 = les blocs de données calculés sont modifiés, le cas échéant après acquittement par l'utilisateur 2 = la cinématique déjà mesurée est calculée et modifiée, le cas échéant après acquittement par l'utilisateur
				DIZAINES DE MILLIERS : axe de mesure (axe rotatif 1 ou 2) 1 = mesurer et calculer tous les axes rotatifs présents 4 = mesurer et calculer uniquement l'axe rotatif 1 5 = mesurer et calculer uniquement l'axe rotatif 2
2		S_TNAME	Nom de la transformation (bloc de données de pivotement ou transformation sur la base de chaînes cinématiques)	
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
4		S_SETV	Diamètre de la sphère de calibrage	
5	alpha 1	S_SETV1	Angle de départ pour la mesure sous angle pour le 1er axe rotatif	
6			S_START_RA1 : angle de départ du 1er axe rotatif	
7			S_END_RA1 : angle final du 1er axe rotatif	
8			S_CMEA_RA1 : nombre de mesures du 1er axe rotatif, 3 pour Mesurer et calculer la cinématique. Pour les points intermédiaires, jusqu'à 12 mesures possibles.	
9			S_POS_RA2 : position du 2e axe rotatif lors de la mesure du 1er axe rotatif [4)]	
10	alpha 2	S_SETV2	Angle de départ pour la mesure sous angle pour le 2e axe rotatif [4)]	
11			S_START_RA2 : angle de départ du 2e axe rotatif [4)]	
12			S_END_RA2 : angle final du 2e axe rotatif [4)]	
13			S_CMEA_RA2 : nombre de mesures du 2e axe rotatif, 3 pour Mesurer et calculer la cinématique. Pour les points intermédiaires, jusqu'à 12 mesures possibles.[4)]	
14			S_POS_RA1 : position du 1er axe rotatif lors de la mesure du 2e axe rotatif	
15	delta	S_SETV4	Valeur de tolérance des vecteurs offsets	
16	DFA	S_FA	Course de mesure	

N°	Paramètre de masque	Paramètre de cycle	Signification	
17	TSA	S_TSA	Zone de fiabilité	
18	Nombre	S_NMSP	Nombre de mesures au même endroit [5] (par défaut=1)	
19		_DMODE	Mode display	
			Valeurs :	UNITES : Plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
20		_AMODE	Mode alternatif	
			Valeurs :	UNITÉS : Contrôle de tolérance oui/non 0 = non 1 = oui : évaluation des valeurs de tolérance des vecteurs -> S_SETV4, S_SETV5
				DIZAINES : Calibrage automatique 0 = sans calibrage automatique 1 = calibrage automatique
				CENTAINES : Automatisme angle de départ 0 = l'angle de départ réglé est utilisé 1 = calcul automatisé de l'angle de départ, pour chaque point de mesure
21		S_KNUM	Numéro du DO à corriger pour mesurer la tête de référence	
			Valeurs	UNITES : la correction intervient toujours dans le décalage grossier, le décalage fin est effacé
				DIZAINES : 1..99 numéro du DO réglable (1=G54)
				MILLIERS : 9 correction DO réglable actif

1) Toutes les valeurs par défaut = 0 ou marquées comme par défaut = xx

2) Poursuite de la broche dans le sens de commutation du palpeur si SD54760 bit 17 = 1

3) Une interrogation de l'utilisateur avec M0 a lieu avant l'inscription. Les vecteurs sont uniquement inscrits avec démarrage CN. Si le programme de mesure est interrompu avec RESET, aucun vecteur calculé n'est inscrit. Les vecteurs sont uniquement inscrits si la tolérance des vecteurs offsets n'est pas dépassée lors du calcul.

4) Axe rotatif 2 uniquement pour cinématiques avec 2 axes rotatifs

5) Affichage dépendant de la donnée de réglage générale SD54760 $SNS_MEA_FUNCTION_MASK_PIECE.

Paramètres des cycles de mesure CYCLE982

```
PROC CYCLE982(INT S_MVAR,INT S_KNUM,INT S_PRNUM,INT S_MA,INT S_MD,REAL S_ID,REAL S_FA,REAL
S_TSA,REAL S_VMS,REAL S_STA1,REAL S_CORA,REAL S_TZL,REAL S_TDIF,INT S_NMSP,INT S_EVNUM,INT
S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-37 Paramètres d'appel CYCLE982 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : calibrage / mesure 0 = calibrage du palpeur d'outil 1 = mesure individuelle de l'outil[3)] 2 = mesure multiple de l'outil, détermination de la longueur et du rayon d'outil (pour outils de fraisage)
				DIZAINES : calibrage ou mesure dans le SCM ou SCP 0 = avec référencement dans le SCM[4)] 1 = avec référencement dans le SCP
				CENTAINES : mesure avec ou sans pivotement pour des outils de fraisage 0 = mesure sans pivotement 1 = mesure avec pivotement
				MILLIERS : cible de correction pour outils de fraisage 0 = détermination de la longueur, ou de la longueur et du rayon (voir S_MVAR position des unités) 1 = détermination du rayon lorsque S_MVAR position des unités = 1 2 = détermination de la longueur et du rayon (face frontale) lorsque S_MVAR position des unités = 1 ou 2 3 = fraise trois tailles tranchant supérieur (face arrière) et détermination de la longueur et du rayon [5)]
				DIZAINES DE MILLIERS : position de l'outil de fraisage ou du foret 0 = position axiale de l'outil de fraisage ou du foret, rayon dans le 2e axe du plan (pour G18 X) [7)] 1 = position radiale de l'outil de fraisage ou du foret, rayon dans le 1er axe du plan (pour G18 Z) [7)]
				CENTAINES DE MILLIERS : calibrage incrémental ou mesure 0 = aucune donnée 1 = calibrage incrémental ou mesure
				MILLIONS : positionner la broche sur angle de départ S_STA1 (uniquement pour mesure outils de fraisage) 0 = ne pas positionner la broche 1 = positionner la broche sur angle de départ S_STA1
2	Sélection	S_KNUM	Variante de correction [2)]	
			Valeurs :	UNITES : correction d'outil 0 = aucune donnée (correction d'outil en géométrie) 1 = correction d'outil en usure
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur) (par défaut=1)	
4	X0	S_MA	Axe de mesure	
			Valeurs :	1 = 1er axe du plan (pour G18 Z) 2 = 2e axe du plan (pour G18 X)

N°	Paramètre de masque	Paramètre de cycle	Signification	
5	+-	S_MD	Sens de la mesure	
			Valeurs :	0 = aucune sélection (direction de mesure est déterminée à partir de la valeur réelle) 1 = positif 2 = négatif
6	Z2	S_ID	Décalage	
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	VMS	S_VMS	Vitesse de mesure variable lors du calibrage [2]	
10	alpha1	S_STA1	Angle de départ pour mesure d'outils de fraisage	
11	alpha2	S_CORA	Angle de correction pour mesure avec pivotement d'outils de fraisage [8]	
12	TZL	S_TZL	Correction zéro lors de la mesure d'outils. Lors du calibrage S_TZL = 0	
13	DIF	S_TDIF	Contrôle d'écart dimensionnel	
14	Mesures	S_NMSP	Nombre de mesures au même endroit [2] (par défaut=1)	
15	EVN	S_EVNUM	Numéro mémoire de moyennes empiriques [2, 9]	
16		S_MCBIT	réservé	
17		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)
				DIZAINES : position de coupe pour outils de tournage et de fraisage (uniquement pour affichage dans les masques de saisie 1 à 9)
				CENTAINES : type d'outil 0 = outil de tournage 1 = fraise 2 = foret
				MILLIERS : stratégie d'accostage par rapport au palpeur d'outil 0 = PLUS [X/Z] ; X pour position axiale de l'outil, Z pour position radiale de l'outil 1 = MOINS [X/Z] ; X pour position axiale de l'outil, Z pour position radiale de l'outil
18		_AMODE	Mode alternatif	
			Valeurs :	UNITES : réservé
				DIZAINES : réservé
				CENTAINES : réservé
				MILLIERS : Accostage de la position de départ après le processus de mesure en cas de calibrage et de mesure individuelle (voir S_MVAR - UNITES) 0 = L'outil se trouve décalé de DFA par rapport à l'arête du palpeur 1 = Accostage de la position de départ

[1] Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

[2] Affichage dépendant de la donnée de réglage générale SD54762 _MEA_FUNCTION_MASK_TOOL

[3] Mesure de l'outil de tournage ou de la fraise ou du foret. Axe de mesure dans paramètre S_MA
Outils de tournage spécifiés par positions de tranchant 1...8, outils de fraisage spécifiés par positions des CENTAINES et des MILLIERS dans paramètre S_MVAR.

[4] La mesure et le calibrage s'effectuent dans le système de coordonnées de base (SCM avec transformation cinématique désactivée).

[5] Pas pour mesure incrémentale.

[6] Uniquement pour mesure multiple `S_MVAR=x2x02` ou `x3x02` (exemple : fraise trois tailles ou fraise à rainurer)

[7] Lorsque la donnée de réglage propre au canal SD42950 $SC_TOOL_LENGTH_TYPE = 2, les composantes de longueur d'outil sont affectées comme pour les outils de tournage

[8] Uniquement pour mesure avec pivotement `S_MVAR=xx1x1`

[9] Calcul de valeur empirique
Plage de valeur de la mémoire de valeurs empiriques : de 1 à 20, numéro de la mémoire de valeurs empiriques, voir la donnée de réglage propre au canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1].

Paramètres des cycles de mesure CYCLE971

```
PROC CYCLE971(INT S_MVAR,INT S_KNUM,INT S_PRNUM,INT S_MA,INT S_MD,REAL S_ID,REAL S_FA,REAL
S_TSA,REAL S_VMS,REAL S_TZL,REAL S_TDIF,INT S_NMSP,REAL S_F1,REAL S_S1,REAL S_F2,REAL
S_S2,REAL S_F3,REAL S_S3,INT S_EVNUM,INT S_MCBIT,INT _DMODE,INT _AMODE)
```

Tableau 3-38 Paramètres d'appel CYCLE971 [1)]

N°	Paramètre de masque	Paramètre de cycle	Signification	
1		S_MVAR	Variante de mesure	
			Valeurs :	UNITES : 0 = calibrage du palpeur d'outil 1 = mesure de l'outil avec broche à l'arrêt (longueur resp. rayon) 2 = mesure de l'outil avec broche en rotation (longueur resp. rayon), voir paramètre S_F1 à S_S4
				DIZAINES : mesure dans le SCM ou le SCP 0 = mesure dans le SCM (avec référencement dans le SCM), mesure de l'outil ou calibrage du palpeur d'outil 1 = mesure dans le SCP (avec référencement dans le SCP), mesure de l'outil ou calibrage du palpeur d'outil
				CENTAINES : Contrôle individuel des dents 0 = non 1 = oui
				MILLIERS : 0 = 0
				DIZAINES DE MILLIERS : Calibrage incrémental ou mesure 0 = aucune donnée 1 = calibrage incrémental ou mesure
				CENTAINES DE MILLIERS : calibrage automatique du palpeur d'outil 0 = ne pas calibrer automatiquement le palpeur d'outil 1 = calibrer automatiquement le palpeur d'outil
				MILLIONS : Calibrage dans le plan avec pivotement de broche 0 = Calibrage dans le plan sans pivotement de broche 1 = Calibrage dans le plan avec pivotement de broche
2	Sélection	S_KNUM	Variante de correction [2)]	
			Valeurs :	UNITES : correction d'outil 0 = aucune donnée (correction d'outil en géométrie) 1 = correction d'outil en usure
3	icône+ quantité	S_PRNUM	Numéro du champ du paramètre de palpeur (pas le numéro de palpeur)	
4	X0	S_MA	Axe de mesure, axe de décalage [4)]	
			Valeurs :	UNITES : numéro de l'axe de mesure 1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y) 3 = 3e axe du plan (pour G17 Z)
				DIZAINES : 0 = 0
				CENTAINES : numéro de l'axe de décalage 0 = aucun axe de décalage 1 = 1er axe du plan (pour G17 X) 2 = 2e axe du plan (pour G17 Y)

N°	Paramètre de masque	Paramètre de cycle	Signification	
5	+-	S_MD	Sens de la mesure	
			Valeurs :	0 = aucune sélection (direction de mesure est déterminée à partir de la valeur réelle) 1 = positif 2 = négatif
6	V	S_ID	Décalage	
			Valeurs :	0 = pour outils sans décalage >0 = • Calibrage : le décalage agit sur le 3e axe du plan (pour G17 Z) si le diamètre de l'outil de calibrage est supérieur au diamètre supérieur du palpeur. L'outil est alors décalé à partir du centre du palpeur d'une distance correspondant au rayon de l'outil, après soustraction de la valeur de S_ID. L'axe de décalage est en outre indiqué dans S_MA . • Mesure : pour les outils à plusieurs arêtes tranchantes, indiquez le décalage entre la longueur d'outil et le point le plus haut de l'arête tranchante, pour la mesure du rayon, ainsi que le décalage du rayon d'outil par rapport au point le plus haut de l'arête tranchante, pour la mesure de la longueur d'outil.
7	DFA	S_FA	Course de mesure	
8	TSA	S_TSA	Zone de fiabilité	
9	VMS	S_VMS	Vitesse de mesure variable lors du calibrage [2)]	
10	TZL	S_TZL	Correction zéro (uniquement pour mesure outil)	
11	DIF	S_TDIF	Contrôle de l'écart dimensionnel pour mesure de l'outil (S_MVAR=xx1 ou S_MVAR=xx2)	
12	Mesures	S_NMSP	Nombre de mesures au même endroit [2)]	
13	F1	S_F1	1re avance pour palpage avec broche en rotation [2)]	
14	S1	S_S1	1re vitesse de rotation pour palpage avec broche en rotation [2)]	
15	F2	S_F2	2e avance pour palpage avec broche en rotation [2)]	
16	S2	S_S2	2e vitesse de rotation pour palpage avec broche en rotation [2)]	
17	F3	S_F3	3e avance pour palpage avec broche en rotation [2)]	
18	S3	S_S3	3e vitesse de rotation pour palpage avec broche en rotation [2)]	
19	EVN	S_EVNUM	Numéro mémoire de valeurs empiriques [2)]	
20		S_MCBIT	Masque des _CBIT resp. _CHBIT	
21		_DMODE	Mode display	
			Valeurs :	UNITES : plan d'usinage G17/G18/G19 0 = compatibilité, le plan effectif avant l'appel de cycle reste actif 1 = G17 (uniquement actif dans le cycle) 2 = G18 (uniquement actif dans le cycle) 3 = G19 (uniquement actif dans le cycle)

N°	Paramètre de masque	Paramètre de cycle	Signification	
22		_AMODE	Mode alternatif	
			Valeurs :	UNITES : Décalage d'outil pour mesure de rayon 1 = non 2 = oui
				DIZAINES : Sens du décalage d'outil pour mesure de rayon dans le 3e axe du plan (pour G17 Z) 1 = positif 2 = négatif
				CENTAINES : Décalage d'outil pour mesure de longueur ou pour calibrage du palpeur dans le 3e axe 0 = compatibilité, auto 1 = non 2 = oui
				MILLIERS : Sens du décalage d'outil pour mesure de longueur dans l'axe de décalage (voir S_MA CENTAINES) 1 = positif 2 = négatif

1) Toutes les valeurs par défaut = 0 ou caractérisées comme par défaut=x

2) Affichage dépendant de la donnée de réglage générale SD54762 MEA_FUNCTION_MASK_TOOL

3) Uniquement pour correction dans outil et si utilisation d'une tolérance dimensionnelle, autrement paramètre = 0

4) Pour la mesure automatique (S_MVAR=1x00xx) aucun affichage axe de mesure, axe de décalage ⇒ S_MA=0.

Paramètres de cycle de mesure CYCLE150

```
PROC CYCLE150(INT S_PICT,INT S_PROT,STRING[160] S_PATH) SAVE
ACTBLOCNO DISPLOF
```

Tableau 3-39 Paramètres d'appel CYCLE150

N°	Paramètre de masque	Paramètre de cycle	Signification	
1	Image des résultats de mesure	S_PICT	Sélection de l'affichage des résultats (par défaut = 0)	
			Valeurs :	UNITES : 0 = Image des résultats de mesure ARRET 1 = Image des résultats de mesure MARCHE
				Dizaines : Sélection du mode d'affichage (valeurs comme SD55613) 1 = afficher l'image des résultats de mesure - Désélection automatique au bout de 8 s 3 = afficher l'image des résultats de mesure - Acquittement avec NC-Start 4 = afficher l'image des résultats de mesure - uniquement pour les alarmes (61303 ... 61306)
2		S_PROT	Sélection de la journalisation (par défaut = 0)	

N°	Paramètre de masque	Paramètre de cycle	Signification	
	Procès-verbal		Valeurs :	UNITES : Sélection Procès-verbal off/on dernière mesure 0 = Procès-verbal ARRET 1 = Procès-verbal MARCHE 2 = Consignation de la dernière mesure
	Type du PV			DIZAINES : Sélection du type de procès-verbal 0 = Procès-verbal standard 1 = Procès-verbal utilisateur (librement définissable)
	Format du procès-verbal			CENTAINES : Sélection du format de procès-verbal 0 = Format texte 1 = Format tableau (pour importation Excel)
	Données du procès-verbal			MILLIERS : Réécrire la sélection ou l'accrocher 0 = nouveau 1 = accrocher
	Procès-verbal Archivage			DIZAINES DE MILLIERS : Sélection de l'emplacement des procès-verbaux 0 = comme le programme pièces 1 = Répertoire
3		S_PATH	Chemin pour le fichier de procès-verbal en fonction de la sélection de l'emplacement des procès-verbaux (indication complète du chemin ou nom de fichier uniquement, par ex. : "//NC:/WKS.DIR/NAME.WPD" ou "MESSPROTOKOLL.TXT"	

Tableaux

4

4.1 Instructions

Remarque

Cycles

La liste des instructions contient tous les cycles pouvant exister dans un programme CN (code G), c.-à-d. pouvant être programmés dans l'éditeur de programmes à l'aide de masques ou qui doivent être programmés pour la rectification sans aide à la programmation. Il n'est pas tenu compte des cycles pouvant être présents dans la commande pour des raisons de compatibilité, mais qui ne peuvent plus être modifiés à l'aide de l'éditeur de programmes de SINUMERIK Operate ("cycles de compatibilité").

Instructions A ... C

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
:	O	Numéro du bloc principal de la CN, fin des repères de saut, opérateur de concaténation		+		PM-NC
*	O	Opérateur de multiplication		+		PM-NC
+	O	Opérateur d'addition		+		PM-NC
-	O	Opérateur de soustraction		+		PM-NC
<	O	Opérateur relationnel, inférieur à		+		PM-NC
<<	O	Opérateur de concaténation pour chaînes		+		PM-NC
<=	O	Opérateur relationnel, inférieur ou égal à		+		PM-NC
=	O	Opérateur d'affectation		+		PM-NC
>=	O	Opérateur relationnel, supérieur ou égal à		+		PM-NC
/	O	Opérateur de division		+		PM-NC
/0 ... ... /7		Le bloc sera sauté (1er niveau)° ... ... Le bloc sera sauté (8e niveau)		+		PM-NC
A	A	Nom d'axe	m/b	+		PM-NC
A2	A	Orientation de l'outil : angle RPY ou angle d'Euler	b	+		PM-NC
A3	A	Orientation de l'outil : 1re composante du vecteur de direction	b	+		PM-NC
A4	A	Orientation de l'outil : 1re composante du vecteur normal à la surface en début de bloc	b	+		PM-NC
A5	A	Orientation de l'outil : 1re composante du vecteur normal à la surface en fin de bloc	b	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
A6	A	Orientation de l'outil : 1re composante du vecteur de direction pour l'axe rotatif du cône	b	+		PM-NC
A7	A	Orientation de l'outil : 1re composante du vecteur pour l'orientation intermédiaire sur l'enveloppe du cône	b	+		PM-NC
ABS	F	Valeur absolue (montant)		+	+	PM-NC
AC	K	Cotation absolue de coordonnées/positions	b	+		PM-NC
ACC	K	Influence de l'accélération axiale actuelle	m	+	+	PM-NC
ACCLIMA	K	Influence de l'accélération axiale maximale actuelle	m	+	+	PM-NC
ACN	K	Cotation absolue pour axes rotatifs, accostage de position dans le sens négatif	b	+		PM-NC
ACOS	F	Arc cosinus (fonction trigonométrique)		+	+	PM-NC
ACP	K	Cotation absolue pour axes rotatifs, accostage de position dans le sens positif	b	+		PM-NC
ACTBLOCNO	P	Affichage du numéro courant d'un bloc d'alarme, même si "Inhibition de l'affichage du bloc courant" (DISPLOF) est actif !		+		PM-NC
ADDFRAME	F	Prise en compte et éventuellement activation d'un frame mesuré		+	-	PM-NC, FM-B
ADIS	A	Distance de transition entre blocs pour fonctions d'interpolation G1, G2, G3, ...	m	+		PM-NC
ADISPOS	A	Distance de transition entre blocs pour le rapide G0	m	+		PM-NC
ADISPOSA	P	Dimension de la fenêtre de tolérance pour IPOBRKA	m	+	+	PM-NC
ALF	A	Angle de relèvement rapide	m	+		PM-NC
AMIRROR	G	Fonction miroir programmable	b	+		PM-NC
AND	K	ET logique		+		PM-NC
ANG	A	Angle d'élément de contour	b	+		PM-NC
AP	A	Angle polaire	m/b	+		PM-NC
APR	K	Droit d'accès en lecture / à l'affichage		+		PM-NC
APRB	K	Droit d'accès en lecture, OPI		+		PM-NC
APRP	K	Droit d'accès en lecture, programme pièce		+		PM-NC
APW	K	Droit d'accès en écriture		+		PM-NC
APWB	K	Droit d'accès en écriture, OPI		+		PM-NC
APWP	K	Droit d'accès en écriture, programme pièce		+		PM-NC
APX	K	Définition de la protection d'accès pour l'exécution de l'élément de langage indiqué		+		PM-NC
AR	A	Angle au centre	m/b	+		PM-NC
AROT	G	Rotation programmable	b	+		PM-NC
AROTS	G	Programmation de rotations de frames avec des angles solides	b	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
AS	K	Définition d'une macro-instruction		+		PM-NC
ASCALE	G	Facteur d'échelle programmable	b	+		PM-NC
ASIN	F	Fonction de calcul, arc sinus		+	+	PM-NC
ASPLINE	G	Akima Spline	m	+		PM-NC
ATAN2	F	Arc tangente 2		+	+	PM-NC
ATOL	A	Tolérance spécifique à l'axe pour fonctions de compresseur, lissage de l'orientation et modes d'arrondissement	m	+		PM-NC
ATRANS	G	Décalage d'origine additif programmable	b	+		PM-NC
AUXFUDEL	P	Suppression pour un canal d'une fonction auxiliaire de la liste globale		+	-	FM-B
AUXFUDELG	P	Suppression pour un canal de toutes les fonctions auxiliaires d'un groupe de fonctions auxiliaires de la liste globale		+	-	FM-B
AUXFUMSEQ	P	Détermination de l'ordre de génération des fonctions auxiliaires M		+	-	FM-B
AUXFUSYNC	P	Génération, sous forme de chaîne de caractères, d'un bloc de programme pièce complet pour le SERUPRO-Fin-ASUP spécifique à un canal à partir de la liste globale des fonctions auxiliaires		+	-	FM-B
AX	K	Descripteur d'axe variable	m/b	+		PM-NC
AXCTSWE	P	Rotation d'un conteneur d'axes		+	-	PM-NC
AXCTSWEC	P	Suppression du déblocage de la rotation du conteneur d'axes		+	+	PM-NC
AXCTSWED	P	Rotation d'un conteneur d'axes (variante d'instruction pour la mise en service !)		+	-	PM-NC
AXIS	K	Descripteur d'axe, adresse d'axe		+		PM-NC
AXNAME	F	Conversion de la chaîne d'entrée en descripteur d'axe		+	-	PM-NC
AXSTRING	F	Conversion de la chaîne en numéro de broche		+	-	PM-NC
AXTOCHAN	P	Demande d'axe pour un canal donné Est possible depuis le programme CN ou à partir d'une action synchrone.		+	+	PM-NC
AXTOSPI	F	Conversion du descripteur d'axe en indice de broche		+	-	PM-NC
B	A	Nom d'axe	m/b	+		PM-NC
B2	A	Orientation de l'outil : angle RPY ou angle d'Euler	b	+		PM-NC
B3	A	Orientation de l'outil : composante du vecteur de direction	b	+		PM-NC
B4	A	Orientation de l'outil : 2e composante du vecteur normal à la surface en début de bloc	b	+		PM-NC
B5	A	Orientation de l'outil : 2e composante du vecteur normal à la surface en fin de bloc	b	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
B6	A	Orientation de l'outil : 2e composante du vecteur de direction pour l'axe rotatif du cône	b	+		PM-NC
B7	A	Orientation de l'outil : 2e composante du vecteur pour l'orientation intermédiaire sur l'enveloppe du cône	b	+		PM-NC
B_AND	O	ET sur bits		+		PM-NC
B_OR	O	OU sur bits		+		PM-NC
B_NOT	O	Négation sur bits		+		PM-NC
B_XOR	O	OU exclusif sur bits		+		PM-NC
BAUTO	G	Définition de la première section spline par les 3 points suivants	m	+		PM-NC
BLOC	K	Définition, en association avec le mot-clé TO, de la section de programme à exécuter dans un sous-programme indirect		+		PM-NC
BLSYNC	K	L'exécution de la routine d'interruption ne doit commencer qu'au changement de bloc suivant		+		PM-NC
BNAT[6]	G	Raccordement naturel avec le premier bloc spline	m	+		PM-NC
BOOL	K	Type de données : valeurs booléennes TRUE / FALSE ou 1/0		+		PM-NC
BOUND	F	Vérifie si la valeur se trouve la plage de valeurs définie. Égalité retourne la valeur de test.		+	+	PM-NC
BRISK[6]	G	Accélération résultante sous forme d'échelon	m	+		PM-NC
BRISKA	P	Activer l'accélération résultante par échelon pour les axes programmés		+	-	PM-NC
BSPLINE	G	Spline B	m	+		PM-NC
BTAN	G	Raccordement tangentiel avec le premier bloc spline	m	+		PM-NC
C	A	Nom d'axe	m/b	+		PM-NC
C2	A	Orientation de l'outil : angle RPY ou angle d'Euler	b	+		PM-NC
C3	A	Orientation de l'outil : 3e composante du vecteur de direction	b	+		PM-NC
C4	A	Orientation de l'outil : 3e composante du vecteur normal à la surface en début de bloc	b	+		PM-NC
C5	A	Orientation de l'outil : 3e composante du vecteur normal à la surface en fin de bloc	b	+		PM-NC
C6	A	Orientation de l'outil : 3e composante du vecteur de direction pour l'axe rotatif du cône	b	+		PM-NC
C7	A	Orientation de l'outil : 3e composante du vecteur pour l'orientation intermédiaire sur l'enveloppe du cône	b	+		PM-NC
CAC	K	Accostage absolu d'une position		+		PM-NC
CACN	K	Valeur rangée dans table est accostée de façon absolue en sens négatif		+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
CACP	K	Valeur rangée dans table est accostée de façon absolue en sens positif		+		PM-NC
CADAPTOF	P	Désactiver l'adaptation de la charge		+	-	PM-NC
CADAPTON	P	Activer l'adaptation de la charge		+	-	PM-NC
CALCDAT	F	Calcul du rayon et du centre d'un cercle passant par 3 ou 4 points		+	-	PM-NC
CALCFIR	P	Adaptation du filtre d'à-coup FIR au mode dynamique		+	-	PM-NC
CALCPOSI	F	Contrôle du dépassement de la zone de protection, de la limitation de la zone de travail et des limites logicielles		+	-	PM-NC
CALL	K	Appel indirect de sous-programme		+		PM-NC
CALLPATH	P	Chemin de recherche programmable pour l'appel de sous-programmes		+	-	PM-NC
CANCEL	P	Annuler une action synchrone modale		+	-	FM-SA
CANCELSUB	P	Annuler le niveau de sous-programme actuel		+	+	FM-SA
CASE	K	Branchement conditionnel		+		PM-NC
CDC	K	Accostage direct d'une position		+		PM-NC
CDOF[6]	G	Désactiver la surveillance de collision	m	+		PM-NC
CDOF2	G	Désactiver la surveillance de collision, fraisage périphérique 3D	m	+		PM-NC
CDON	G	Activer la surveillance de collision	m	+		PM-NC
CFC[6]	G	Avance constante au niveau du contour	m	+		PM-NC
CFIN	G	Avance constante uniquement pour courbure concave, pas pour courbure convexe	m	+		PM-NC
CFINE	F	Affectation d'un décalage fin à une variable FRAME		+	-	PM-NC
CFTCP	G	Avance constante au point de référence de l'arête tranchante de l'outil, trajectoire centrale	m	+		PM-NC
CHAN	K	Spécification du domaine de validité de données		+		PM-NC
CHANDATA	P	Réglage du numéro de canal pour des accès à des données de canal		+	-	PM-NC
CHAR	K	Type de données : caractère ASCII		+		PM-NC
CHF	A	Chanfrein ; valeur = longueur du chanfrein	b	+		PM-NC
CHKDM	F	Contrôle d'univocité dans un magasin		+	-	FM-TM
CHKDNO	F	Contrôle d'univocité des numéros D		+	-	PM-NC
CHR	A	Chanfrein ; valeur = longueur du chanfrein dans la direction du déplacement		+		PM-NC
CIC	K	Accostage relatif d'une position		+		PM-NC
CIP	G	Interpolation circulaire avec point intermédiaire	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
CLEARM	P	Effacement d'un / de plusieurs repères de coordination entre canaux		+	+	PM-NC
CLRINT	P	Annuler une interruption		+	-	PM-NC
CMIRROR	F	fonction miroir par rapport à un axe de coordonnées		+	-	PM-NC
COARSEA	K	Fin du déplacement lors de l'atteinte de "Arrêt précis grossier"	m	+		PM-NC
COLLPAIR	F	Vérification de l'appartenance à une paire de collision		+		PM-NC
COMPCAD	G	Activation de la fonction compacteur COMPCAD	m	+		PM-NC
COMPCURV	G	Activation de la fonction compacteur COMPCURV	m	+		PM-NC
COMPLETE		Instruction de commande pour la lecture et l'écriture de données		+		PM-NC
COMPOF[6]	G	Activation du compactage de bloc CN	m	+		PM-NC
COMPON	G	Activation de la fonction compacteur COMPON	m	+		PM-NC
COMPSURF	G	Activation de la fonction compacteur COMSURF	m	+		PM-NC
CONTDCON	P	Activation du décodage du contour, sous forme tabulaire		+	-	PM-NC
CONTPRON	P	Activation de la préparation des références		+	-	PM-NC
CORROF	P	Désélection de tous les déplacements forcés actifs.		+	-	PM-NC
CORRTC	F	Modification des vecteurs d'offset ou des vecteurs de direction des organes porte-outil orientables après mesurage de la machine.		+	-	PM-NC
CORRTRAFO	F	Modification des vecteurs d'offset ou des vecteurs de direction des axes d'orientation dans le modèle cinématique de la machine		+	-	PM-NC
COS	F	Cosinus (fonction trigonométrique)		+	+	PM-NC
COUPDEF	P	Définition groupe réducteurs électroniques / groupe broches synchrones		+	-	PM-NC
COUPDEL	P	Suppression groupe réducteurs électroniques		+	-	PM-NC
COUPOF	P	Désactivation du groupe réducteurs électroniques / de la paire de broches synchrones		+	-	PM-NC
COUPOFS	P	Désactivation du groupe réducteurs électroniques / de la paire de broches synchrones avec arrêt de la broche asservie		+	-	PM-NC
COUPON	P	Activation du groupe réducteurs électroniques / de la paire de broches synchrones		+	-	PM-NC
COUPONC	P	Activation du groupe réducteurs électroniques / de la paire de broches synchrones avec prise en charge de la programmation précédente		+	-	PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
COUPRES	P	Réinitialisation groupe réducteurs électroniques		+	-	PM-NC
CP[6]	G	Déplacement avec interpolation	m	+		PM-NC
CPBC	K	Couplage générique : critère de changement de bloc		+	+	FM-A
CPDEF	K	Couplage générique : création d'un module de couplage		+	+	FM-A
CPDEL	K	Couplage générique : suppression d'un module de couplage		+	+	FM-A
CPFMOF	K	Couplage générique : comportement de l'axe asservi lors d'une désactivation complète		+	+	FM-A
CPFMON	K	Couplage générique : comportement de l'axe asservi lors de l'activation		+	+	FM-A
CPFMSON	K	Couplage générique : mode de synchronisation		+	+	FM-A
CPFPOS	K	Couplage générique : position synchrone de l'axe asservi		+	+	FM-A
CPFRS	K	Couplage générique : système de référence des coordonnées		+	+	FM-A
CPLA	K	Couplage générique : définition d'un axe pilote		+	-	FM-A
CPLCTID	K	Couplage générique : numéro de la table de courbes		+	+	FM-A
CPLDEF	K	Couplage générique : définition d'un axe pilote et création d'un module de couplage		+	+	FM-A
CPLDEL	K	Couplage générique : suppression d'un axe pilote d'un module de couplage		+	+	FM-A
CPLDEN	K	Couplage générique : dénominateur du facteur de couplage		+	+	FM-A
CPLINSC	K	Couplage générique : facteur d'échelle pour la valeur d'entrée d'un axe pilote		+	+	FM-A
CPLINTR	K	Couplage générique : valeur de décalage pour la valeur d'entrée d'un axe pilote		+	+	FM-A
CPLNUM	K	Couplage générique : numérateur du facteur de couplage		+	+	FM-A
CPLOF	K	Couplage générique : désactivation d'un axe pilote d'un module de couplage		+	+	FM-A
CPLON	K	Couplage générique : activation d'un axe pilote d'un module de couplage		+	+	FM-A
CPLOUTSC	K	Couplage générique : facteur d'échelle pour la valeur de sortie d'un couplage		+	+	FM-A
CPLOUTTR	K	Couplage générique : valeur de décalage pour la valeur de sortie d'un couplage		+	+	FM-A
CPLPOS	K	Couplage générique : position synchrone de l'axe pilote		+	+	FM-A
CPLSETVAL	K	Couplage générique : référence du couplage		+	+	FM-A

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
CPMALARM	K	Couplage générique : inhibition des générations d'alarme spéciales relatives au couplage		+	+	FM-A
CPMBRAKE	K	Couplage générique : comportement de l'axe asservi lors de certains signaux et ordres d'arrêt		+	-	FM-A
CPMPRT	K	Couplage générique : Comportement du couplage au lancement du programme pièce lors d'une recherche en mode test du programme		+	+	FM-A
CPMRESET	K	Couplage générique : comportement du couplage lors d'un RESET		+	+	FM-A
CPMSTART	K	Couplage générique : comportement du couplage au lancement du programme pièce		+	+	FM-A
CPMVDI	K	Couplage générique : comportement de l'axe asservi en réaction à certains signaux d'interface CN/AP		+	+	FM-A
CPOF	K	Couplage générique : désactivation d'un module de couplage		+	+	FM-A
CPON	K	Couplage générique : activation d'un module de couplage		+	+	FM-A
CPRECOF[6]	G	Désactiver la précision de contour programmable	m	+		PM-NC
CPRECON	G	Activer la précision de contour programmable	m	+		PM-NC
CPRES	K	Couplage générique : activation des données configurées du couplage de broches synchrones		+	-	FM-A
CPROT	P	Activation/désactivation de zones de protection spécifiques au canal		+	-	PM-NC
CPROTDEF	P	Définition d'une zone de protection spécifique au canal		+	-	PM-NC
CPSETTYPE	K	Couplage générique : type de couplage		+	+	FM-A
CPSYNCOP	K	Couplage générique : valeur de seuil pour la synchronisme "grossière" de position		+	+	FM-A
CPSYNCOP2	K	Couplage générique : valeur de seuil pour le synchronisme de position "Grossier" 2		+	+	FM-A
CPSYNCOV	K	Couplage générique : valeur de seuil pour la synchronisation "grossière" de la vitesse		+	+	FM-A
CPSYNFIP	K	Couplage générique : valeur de seuil pour la synchronisation "fine" de position		+	+	FM-A
CPSYNFIP2	K	Couplage générique : valeur de seuil pour le synchronisme de position "Fin" 2		+	+	FM-A
CPSYNFIV	K	Couplage générique : valeur de seuil pour la synchronisation "fine" de la vitesse		+	+	FM-A
CR	A	Rayon du cercle	b	+		PM-NC
CROT	F	rotation du système de coordonnées actuel		+	-	PM-NC
CROTS	F	Rotations de frames programmables avec angles solides (rotation dans les axes indiqués)	b	+	-	PM-NC
CRPL	F	Rotation de frame dans un plan quelconque		+	-	FM-B

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
CSCALE	F	Facteur d'échelle pour plusieurs axes		+	-	PM-NC
CSPLINE	F	Spline cubique	m	+		PM-NC
CT	G	Cercle avec transition tangentielle	m	+		PM-NC
CTAB	F	Déterminer position axe asservi correspondant à position axe pilote d'après table de courbes		+	+	PM-NC
CTABDEF	P	Activer la définition d'une table		+	-	PM-NC
CTABDEL	P	Suppression d'une table de courbes		+	-	PM-NC
CTABEND	P	Désactiver la définition d'une table		+	-	PM-NC
CTABEXISTS	F	Vérifie la table de courbes ayant le numéro n		+	+	PM-NC
CTABFNO	F	Nombre des tables de courbes encore possibles dans la mémoire		+	+	PM-NC
CTABFPOL	F	Nombre des polynômes encore possibles dans la mémoire		+	+	PM-NC
CTABFSEG	F	Nombre des segments de courbes encore possibles dans la mémoire		+	+	PM-NC
CTABID	F	Indication du numéro de table de la n-ième table de courbes		+	+	PM-NC
CTABINV	F	Déterminer position axe pilote correspondant à position axe asservi d'après table de courbes		+	+	PM-NC
CTABISLOCK	F	Retourne l'état de blocage de la table de courbes ayant le numéro n		+	+	PM-NC
CTABLOCK	P	Verrouillage de la suppression et de l'écrasement		+	+	PM-NC
CTABMEMTYP	F	Retourne la mémoire dans laquelle se trouve la table de courbes ayant le numéro n.		+	+	PM-NC
CTABMPOL	F	Nombre maximum des polynômes possibles dans la mémoire		+	+	PM-NC
CTABMSEG	F	Nombre maximum des segments de courbes possibles dans la mémoire		+	+	PM-NC
CTABNO	F	Nombre de tables de courbes définies dans SRAM ou DRAM		+	+	FM-A
CTABNOMEM	F	Nombre de tables de courbes définies dans SRAM ou DRAM		+	+	PM-NC
CTABPERIOD	F	Restitue la périodicité de la table de courbes de numéro n		+	+	PM-NC
CTABPOL	F	Nombre des polynômes déjà utilisés dans la mémoire		+	+	PM-NC
CTABPOLID	F	Nombre des polynômes de courbes de la table de courbes ayant le numéro n		+	+	PM-NC
CTABSEG	F	Nombre des segments de courbes déjà utilisés dans la mémoire		+	+	PM-NC
CTABSEGID	F	Nombre des segments de courbes de la table de courbes ayant le numéro n		+	+	PM-NC
CTABSEV	F	Indique la valeur de fin de l'axe asservi du segment de la table de courbes		+	+	PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
CTABSSV	F	Indique la valeur de départ de l'axe asservi du segment de la table de courbes		+	+	PM-NC
CTABTEP	F	Indique la valeur de l'axe pilote à la fin de la table de courbes		+	+	PM-NC
CTABTEV	F	Indique la valeur de l'axe asservi à la fin de la table de courbes		+	+	PM-NC
CTABTMAX	F	Indique la valeur maximale de l'axe asservi de la table de courbes		+	+	PM-NC
CTABTMIN	F	Indique la valeur minimale de l'axe asservi de la table de courbes		+	+	PM-NC
CTABTSP	F	Indique la valeur de l'axe pilote au début de la table de courbes.		+	+	PM-NC
CTABTSV	F	Indique la valeur de l'axe asservi au début de la table de courbes		+	+	PM-NC
CTABUNLOCK	P	Annulation du verrouillage de la suppression et de l'écrasement		+	+	PM-NC
CTOL	A	Tolérance de contour pour fonctions de compresseur, lissage de l'orientation et modes d'arrondissement	m	+	-	PM-NC
CTOLG0	A	Tolérance de contour pour les déplacements à vitesse rapide	m	+	-	PM-NC
CTRANS	F	Décalage d'origine pour plusieurs axes		+	-	PM-NC
CUT2D[6]	G	2D-WRK	m	+		PM-NC
CUT2DD	G	CRO 2½D rapportée à un outil différentiel	m	+		PM-NC
CUT2DF	G	CRO 2D, relative au frame actuel (plan incliné)	m	+		PM-NC
CUT2DFD	G	CRO 2½D rapportée à un outil différentiel, relative au frame actuel (plan incliné)	m	+		PM-NC
CUT3DC	G	CRO 3D pour le fraisage périphérique	m	+		PM-NC
CUT3DCC	G	CRO 3D pour le fraisage périphérique en tenant compte d'une surface de délimitation avec correction de rayon 3D : contour de la surface d'usinage	m	+		PM-NC
CUT3DCCD	G	CRO 3D pour le fraisage périphérique en tenant compte d'une surface de délimitation avec outil différentiel sur la trajectoire du centre de l'outil : déplacement perpendiculaire jusqu'à la surface de délimitation	m	+		PM-NC
CUT3DCD	G	CRO 3D rapportée à un outil différentiel pour le fraisage périphérique	m	+		PM-NC
CUT3DF	G	CRO 3D pour le fraisage en bout avec modification de l'orientation.	m	+		PM-NC
CUT3DFD	G	CRO 3D rapportée à un outil différentiel pour le fraisage en bout avec modification d'orientation	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
CUT3DFF	G	CRO 3D pour le fraisage en bout avec orientation constante. L'orientation de l'outil est définie par G17 - G19 et, le cas échéant, par la composante rotationnelle d'un frame.	m	+		PM-NC
CUT3DFS	G	CRO 3D pour le fraisage en bout avec orientation constante. L'orientation de l'outil est définie par G17 - G19 et n'est pas influencée par les frames.	m	+		PM-NC
CUTCONOF[6]	G	Désactiver la correction de rayon constante	m	+		PM-NC
CUTCONON	G	Activer la correction de rayon constante	m	+		PM-NC
CUTMOD	A	Activer la modification des données de correction pour outils indexables (en liaison avec des porte-outils orientables)	m	+		PM-NC
CUTMODK	A	Activer la modification des données de correction pour outils indexables (en liaison avec des transformations d'orientation ayant été définies au moyen de chaînes cinématiques)	m	+		PM-NC
CYCLE60	C (T)	Cycle de gravure		+		PM-NC
CYCLE61	C (T)	Surfaçage		+		PM-NC
CYCLE62	C (T)	Appel de contour		+		PM-NC
CYCLE63	C (T)	Fraisage de poche de contour		+		PM-NC
CYCLE64	C (T)	Perçage des avant-trous d'une poche de contour		+		PM-NC
CYCLE70	C (T)	Fraisage de filetages		+		PM-NC
CYCLE72	C (T)	Fraisage en contournage		+		PM-NC
CYCLE76	C (T)	Fraisage d'un tourillon rectangulaire		+		PM-NC
CYCLE77	C (T)	Fraisage d'un tourillon circulaire		+		PM-NC
CYCLE78	C (T)	Fraisage de filetages		+		PM-NC
CYCLE79	C (T)	Polyèdre		+		PM-NC
CYCLE81	C (T)	Perçage, centrage		+		PM-NC
CYCLE82	C (T)	Perçage, lamage		+		PM-NC
CYCLE83	C (T)	Perçage de trous profonds		+		PM-NC
CYCLE84	C (T)	Taraudage sans porte-taraud compensateur		+		PM-NC
CYCLE85	C (T)	Alésage à l'alésoir		+		PM-NC
CYCLE86	C (T)	Alésage au tour		+		PM-NC
CYCLE92	C (T)	Tronçonnage		+		PM-NC
CYCLE95	C (T)	Chariotage du contour		+		PM-NC
CYCLE98	C (T)	Concaténation de filetages		+		PM-NC
CYCLE99	C (T)	Filetage au tour		+		PM-NC
CYCLE116	C (M)	Calcul du centre et du rayon d'un cercle		+		PM-MC
CYCLE119	C (M)	Détermination de la position spatiale		+		PM-MC
CYCLE150	C (M)	Affichage/journalisation des résultats de mesure		+		PM-MC
CYCLE435	C (T)	Calculer la position de dressage		+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
CYCLE495	C (T)	Profilage		+		PM-NC
CYCLE750	C (A)	Cycle de travail interne pour CYCLE751 ... CYCLE759 (contient l'instruction MMC d'appel de la fonction)		-		FM-A
CYCLE751	C (A)	Ouverture / exécution / fermeture d'une session d'optimisation		M		FM-A
CYCLE752	C (A)	Ajout d'un axe à une session d'optimisation		M		FM-A
CYCLE753	C (A)	Sélection du mode d'optimisation		M		FM-A
CYCLE754	C (A)	Ajout / suppression d'un bloc de données		M		FM-A
CYCLE755	C (A)	Sauvegarde / restauration d'un bloc de données		M		FM-A
CYCLE756	C (A)	Activation des résultats d'optimisation		M		FM-A
CYCLE757	C (A)	Sauvegarde des données d'optimisation		M		FM-A
CYCLE758	C (A)	Modification de la valeur d'un paramètre		M		FM-A
CYCLE759	C (A)	Lecture de la valeur d'un paramètre		M		FM-A
CYCLE782	C (T)	Adapter à la charge		+		PM-NC
CYCLE800	C (T)	Orientation		+		PM-NC
CYCLE801	C (T)	Réseau de trous ou cadre		+		PM-NC
CYCLE802	C (T)	Positions quelconques		+		PM-NC
CYCLE806	C (T)	Tournage par interpolation		+		PM-NC
CYCLE830	C (T)	Perçage de trous profonds 2		+		PM-NC
CYCLE832	C (T)	High Speed Settings		+		PM-NC
CYCLE840	C (T)	Taraudage avec porte-taraud compensateur		+		PM-NC
CYCLE899	C (T)	Fraisage de rainure borgne		+		PM-NC
CYCLE930	C (T)	Gorge		+		PM-NC
CYCLE940	C (T)	Dégagement de formes		+		PM-NC
CYCLE951	C (T)	Chariotage		+		PM-NC
CYCLE952	C (T)	Plongée de contour		+		PM-NC
CYCLE961	C (M)	Permet de déterminer la position d'un coin (intérieur ou extérieur) sur une pièce et de l'utiliser pour le décalage de l'origine		+		PM-MC
CYCLE971	C (M)	Permet d'étalonner un palpeur, de mesurer la longueur et/ou du rayon d'un outil (uniquement pour la technologie fraisage)		+		PM-MC
CYCLE973	C (M)	Permet d'étalonner un palpeur de pièce sur une surface de la pièce ou dans une rainure (uniquement pour la technologie tournage)		+		PM-MC
CYCLE974	C (M)	Permet de déterminer l'origine pièce dans l'axe de mesure sélectionné ou une correction d'outil avec une mesure en 1 point (uniquement pour la technologie tournage)		+		PM-MC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
CYCLE976	C (M)	Permet d'étalonner entièrement un palpeur d'outil pour un axe et une direction donnés dans le plan de travail ou sur une arête (dans une bague étalon ou sur une sphère de calibrage)		+		PM-MC
CYCLE977	C (M)	Permet de déterminer le centre dans le plan, ainsi que la largeur ou le diamètre		+		PM-MC
CYCLE978	C (M)	Permet de mesurer la position d'une arête dans le système de coordonnées pièce		+		PM-MC
CYCLE979	C (M)	Permet de déterminer le centre dans le plan, ainsi que de mesurer le rayon de segments circulaires		+		PM-MC
CYCLE982	C (M)	Permet d'étalonner un palpeur d'outil, de mesurer les outils de tournage, de perçage et de fraisage (uniquement pour la technologie tournage)		+		PM-MC
CYCLE994	C (M)	Permet de déterminer l'origine pièce dans l'axe de mesure sélectionné avec une mesure en 2 points (uniquement pour la technologie tournage)		+		PM-MC
CYCLE995	C (M)	Permet de mesurer la déviation angulaire de la broche sur une machine-outil.		+		PM-MC
CYCLE996	C (M)	Permet de déterminer des données de transformation pertinentes pour les transformations cinématiques contenant des axes rotatifs.		+		PM-MC
CYCLE997	C (M)	Permet de déterminer le centre et le diamètre d'une sphère ainsi que les centres de trois sphères réparties		+		PM-MC
CYCLE998	C (M)	Permet de déterminer la position angulaire d'une surface (plan) par rapport au plan de travail, ainsi que l'angle des arêtes dans le système de coordonnées pièce.		+		PM-MC
CYCLE4071	C (T)	Rectification longitudinale avec pénétration au point d'inversion		+		PM-NC
CYCLE4072	C (T)	Rectification longitudinale avec pénétration au point d'inversion et signal d'abandon		+		PM-NC
CYCLE4073	C (T)	Rectification longitudinale avec pénétration continue		+		PM-NC
CYCLE4074	C (T)	Rectification longitudinale avec pénétration continue et signal d'abandon		+		PM-NC
CYCLE4075	C (T)	Rectification plane avec pénétration au point d'inversion		+		PM-NC
CYCLE4077	C (T)	Rectification plane avec pénétration au point d'inversion et signal d'abandon		+		PM-NC
CYCLE4078	C (T)	Rectification plane avec pénétration continue		+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
CYCLE4079	C (T)	Rectification plane avec pénétration intermittente		+		PM-NC
CYCLE9960	C (M)	Mesure complète de la cinématique		+		PM-MC

Instructions D ... F

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1][2][3][4][5] **Explications voir Légende (Page 1281).**						
D	A	Numéro de correction d'outil		+		PM-NC
D0	A	Pour D0, les corrections seront inopérantes pour l'outil en question		+		PM-NC
DAC	K	Programmation au diamètre spécifique à l'axe non modale absolue	b	+		PM-NC
DC	K	Cotation absolue pour axes rotatifs, accoster directement la position	b	+		PM-NC
DCI	K	Affecter la classe de données I (= Individual) (SINUMERIK 828D uniquement !)		+		PM-NC
DCM	K	Affecter la classe de données M (= Manufacturer) (SINUMERIK 828D uniquement !)		+		PM-NC
DCU	K	Affecter la classe de données U (= User) (SINUMERIK 828D uniquement !)		+		PM-NC
DEF	K	Définition de variables		+		PM-NC
DEFAULT	K	Branche d'un branchement CASE		+		PM-NC
DEFINE	K	Mot-clé pour les définitions de macros		+		PM-NC
DELAYFSTOF	P	Définition de la fin d'une plage d'arrêt temporisé	m	+	-	PM-NC
DELAYFSTON	P	Définition du début d'une plage d'arrêt temporisé	m	+	-	PM-NC
DELDL	F	Suppression des corrections additives		+	-	PM-NC
DELDTG	P	Effacement de la distance restant à parcourir		-	+	FM-SA
DELETE	P	Effacement du fichier indiqué. Le nom du fichier peut être indiqué avec chemin d'accès et extension de fichier.		+	-	PM-NC
DELMLOWNER	F	Suppression de l'emplacement de magasin propriétaire de l'outil		+	-	FM-TM
DELMLRES	F	Suppression de la réservation d'emplacement de magasin		+	-	FM-TM
DELMT	P	Suppression d'un multitool		+	-	FM-TM
DELOBJ	F	Suppression d'éléments de chaînes cinématiques, de zones de protection, d'éléments de zones de protection, de paires de collision et de données de transformation		+		PM-NC
DELT	P	Suppression d'un outil		+	-	FM-TM

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1][2][3][4][5] **Explications voir Légende (Page 1281).**						
DELTC	P	Suppression d'un bloc de données de l'organe porte-outil		+	-	FM-TM
DELTOOLENV	F	Suppression d'enregistrements de description de contournage d'outil		+	-	PM-NC
DIACYCOFA	K	Programmation au diamètre modale spécifique à l'axe : désactivée dans les cycles	m	+		FM-A
DIAM90	G	Programmation au diamètre pour G90 ; programmation au rayon pour G91	m	+		PM-NC
DIAM90A	K	Programmation au diamètre modale spécifique à l'axe pour G90 et AC, programmation au rayon pour G91 et IC	m	+		PM-NC
DIAMCHAN	K	Prise en compte de tous les axes des fonctions d'axe PM dans l'état du canal de la programmation au diamètre.		+		PM-NC
DIAMCHANA	K	prise en compte de l'état de canal de la programmation de diamètre		+		PM-NC
DIAMCYCOF	G	Programmation du diamètre spécifique au canal : désactivée dans les cycles	m	+		FM-A
DIAMOF[6]	G	Programmation du diamètre : désactivée état initial, voir le constructeur de machines	m	+		PM-NC
DIAMOFA	K	Programmation modale du diamètre spécifique à l'axe : désactivée état initial, voir le constructeur de machines	m	+		PM-NC
DIAMON	G	Programmation du diamètre : activé	m	+		PM-NC
DIAMONA	K	Programmation modale du diamètre spécifique à l'axe : activée validation, voir le constructeur de machines	m	+		PM-NC
DIC	K	Programmation au diamètre spécifique à l'axe non modale relative	b	+		PM-NC
DILF	A	Distance de retrait (longueur)	m	+		PM-NC
DISABLE	P	Désactivation d'une interruption		+	-	PM-NC
DISC	A	Surhaussement arc de raccordement correction du rayon d'outil	m	+		PM-NC
DISCL	A	Distance entre le point final du mouvement de pénétration rapide et le plan d'usinage		+		PM-NC
DISPLOF	PA	Inhibition de l'affichage du bloc courant		+		PM-NC
DISPLON	PA	Annulation de l'inhibition d'affichage du bloc courant		+		PM-NC
DISPR	A	Différence sur le contour pour repositionnement	b	+		PM-NC
DISR	A	Distance de repositionnement	b	+		PM-NC
DISRP	A	Distance entre le plan de retrait et le plan d'usinage lors d'un accostage et retrait en douceur		+		PM-NC
DITE	A	Course en sortie de filetage	m	+		PM-NC
DITS	A	Course en entrée de filetage	m	+		PM-NC
DIV	K	Division avec résultat entier		+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1][2][3][4][5] **Explications voir Légende (Page 1281).**						
DL	A	Sélection de la correction d'outil additive fonction de l'endroit (DL, correction de la somme, mise en place)	m	+		PM-NC
DO	K	Action synchrone : déclenchement d'actions lorsque la condition est remplie		-	+	FM-SA
DRFOF	P	Désactivation (effacement) des décalages par manivelle (DRF)	m	+	-	PM-NC
DRIVE	G	Accélération tangentielle fonction de la vitesse	m	+		PM-NC
DRIVEA	P	Activation de la caractéristique d'accélération avec coude pour les axes programmés		+	-	PM-NC
DYNFINISH	G	Dynamique pour finition	m	+		PM-NC
DYNNORM[6]	G	Dynamique normale	m	+		PM-NC
DYNPOS	G	Dynamique pour mode positionnement, taraudage	m	+		PM-NC
DYNPREC	G	Dynamique pour superfinition	m	+		PM-NC
DYNROUGH	G	Dynamique pour ébauche	m	+		PM-NC
DYNSEMIFIN	G	Dynamique pour la préfinition	m	+		PM-NC
DZERO	P	Déclare tous les numéros D de l'unité TO non valides		+	-	PM-NC
EAUTO	G	Détermination de la dernière section spline avec les 3 derniers points	m	+		PM-NC
EGDEF	P	Définition d'un réducteur électronique		+	-	PM-NC
EGDEL	P	Suppression de la définition de couplage pour l'axe asservi		+	-	PM-NC
EGOFC	P	Désactivation continue de l'entraînement électronique		+	-	PM-NC
EGOFS	P	Désactivation sélective de l'entraînement électronique		+	-	PM-NC
EGON	P	Activation du réducteur électronique		+	-	PM-NC
EGONSYN	P	Activation du réducteur électronique		+	-	PM-NC
EGONSYNE	P	Activation de l'entraînement électronique, avec prédéfinition du mode d'accostage		+	-	PM-NC
ELSE	K	Programme CN : branche de programme lorsque la condition IF n'est pas remplie		+	-	PM-NC
ELSE	K	Action synchrone : déclenchement d'actions lorsque la condition n'est pas remplie		-	+	FM-SA
ENABLE	P	Activation d'une interruption		+	-	PM-NC
ENAT[6]	G	Raccordement naturel avec le bloc de déplacement suivant	m	+		PM-NC
ENDFOR	K	Dernière ligne d'une boucle de comptage FOR		+		PM-NC
ENDIF	K	Dernière ligne d'un branchement IF		+		PM-NC
ENDLABEL	K	Repère de fin pour répétitions du programme pièce via REPEAT		+		PM-NC, FM-B
ENDLOOP	K	Dernière ligne d'une boucle sans fin LOOP		+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1][2][3][4][5] **Explications voir Légende (Page 1281).**						
ENDPROC	K	Dernière ligne d'un programme commençant par PROC		+		
ENDWHILE	K	Dernière ligne d'une boucle WHILE		+		PM-NC
ESRR	P	Paramétrage d'un retrait ESR contrôlé par le variateur dans le variateur		+		PM-NC
ESRS	P	Paramétrage d'un arrêt ESR contrôlé par le variateur dans le variateur		+		PM-NC
ETAN	G	Raccordement tangentiel avec le bloc de déplacement suivant en début de courbe de type spline	m	+		PM-NC
EVERY	K	Exécution d'une action synchrone lors de la transition de la condition de FALSE à TRUE		-	+	FM-SA
EX	K	Mot-clé pour l'affectation en notation exponentielle		+		PM-NC
EXECSTRING	P	Transfert d'une variable STRING avec la ligne du programme pièce à exécuter		+	-	PM-NC
EXECTAB	P	Exécution d'un élément d'une table de déplacements		+	-	PM-NC
EXECUTE	P	Activation de l'exécution du programme		+	-	PM-NC
EXP	F	Fonction exponentielle ex		+	+	PM-NC
EXTCALL	A	Exécution de sous-programme externe		+	+	PM-NC
EXTCLOSE	P	Fermeture de l'appareil / du fichier externe ouvert pour écriture		+	-	PM-NC
EXTERN	K	Déclaration d'un sous-programme avec transfert de paramètres		+		PM-NC
EXTOPEN	P	Ouverture pour écriture de l'appareil / du fichier externe pour le canal		+	-	PM-NC
F	A	Valeur d'avance (l'arrêt temporisé est programmé aussi avec F en liaison avec G4)		+	+	PM-NC
FA	K	Avance axiale	m	+	+	PM-NC
FAD	A	Avance d'approche pour accostage et retrait en douceur		+		PM-NC
FALSE	K	Constante logique : faux		+	+	PM-NC
FB	A	Avance non modale		+		PM-NC
FCTDEF	P	Définition d'une fonction polynôme		+	-	PM-NC
FCUB	G	Avance variable selon une courbe de type spline cubique	m	+		PM-NC
FD	A	Avance tangentielle pour correction par manivelle	b	+		PM-NC
FDA	K	Avance axiale pour correction par manivelle	b	+		PM-NC
FENDNORM[6]	G	Décélération aux angles OFF	m	+		PM-NC
FFWOF[6]	G	Désactivation de la commande anticipatrice	m	+		PM-NC
FFWON	G	Activation de la commande anticipatrice	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1][2][3][4][5] **Explications voir Légende (Page 1281).**						
FGREF	K	Rayon de référence pour les axes rotatifs ou facteurs de référence de trajectoire pour les axes d'orientation (interpolation vectorielle)	m	+		PM-NC
FGROUP	P	Sélection de l'axe ou des axes avec avance tangentielle		+	-	PM-NC
FI	K	Paramètre d'accès aux données frame : décalage fin		+		PM-NC
FIFOCTRL	G	Commande de la mémoire tampon d'exécution	m	+		PM-NC
FILEDATE	P	Fournit la date du dernier accès en écriture au fichier		+	-	PM-NC
FILEINFO	P	Fournit la somme de FILEDATE, FILESIZE, FILESTAT et FILETIME		+	-	PM-NC
FILESIZE	P	Fournit la taille actuelle du fichier		+	-	PM-NC
FILESTAT	P	Fournit l'état du fichier des droits lecture, écriture, exécution, affichage, suppression (rwxsd)		+	-	PM-NC
FILETIME	P	Fournit l'heure du dernier accès en écriture au fichier		+	-	PM-NC
FINEA	K	Fin du déplacement lors de l'atteinte de "Arrêt précis fin"	m	+		PM-NC
FL	K	Vitesse limite pour axes synchrones	m	+		PM-NC
FLIM	A	Adaptation de la vitesse tangentielle maximale	m	+	-	PM-NC
FLIN	G	Avance variable linéairement	m	+		PM-NC
FMA	K	Plusieurs avances axiales	m	+		PM-NC
FNORM[6]	G	Avance normale selon DIN66025	m	+		PM-NC
FOC	K	Limitation de couple/force à effet non modal	b	-	+	FM-SA
FOCOF	K	Désactivation de la limitation de couple/force à effet modal	m	-	+	FM-SA
FOCON	K	Activation de la limitation de couple/force à effet modal	m	-	+	FM-SA
FOR	K	Boucle de comptage avec nombre fixe d'itérations		+		PM-NC
FP	A	Point fixe : numéro du point fixe à accoster	b	+		PM-NC
FPO	K	Variation de l'avance programmée avec un polynôme		+		PM-NC
FPR	P	Identification axe rotatif		+	-	PM-NC
FPRAOF	P	Désactivation de l'avance par tour		+	-	PM-NC
FPRAON	P	Activation de l'avance par tour		+	-	PM-NC
FRAME	K	Type de données pour la définition des systèmes de coordonnées		+		PM-NC
FRC	A	Avance pour rayon et chanfrein	b	+		PM-NC
FRCM	A	Avance modale pour rayon et chanfrein	m	+		PM-NC
FROM	K	L'action est exécutée lorsque la condition est remplie une fois et tant que l'action synchrone est active		-	+	FM-SA
FTOC	P	Modifier la correction fine d'outil		-	+	FM-SA

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1][2][3][4][5] **Explications voir Légende (Page 1281).**						
FTOCOF[6]	G	Désactivation de la correction fine d'outil active en ligne	m	+		PM-NC
FTOCON	G	Activation de la correction fine d'outil active en ligne	m	+		PM-NC
FXS	K	Déplacement en butée activé	m	+	+	PM-NC
FXST	K	Limite de couple pour l'accostage de butée	m	+	+	PM-NC
FXSW	K	Fenêtre de surveillance pour l'accostage de butée		+	+	PM-NC
FZ	K	Avance par dent	m	+		PM-NC

Instructions G ... L

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
G0	G	Interpolation linéaire en rapide (déplacement à vitesse rapide)	m	+		PM-NC
G1[6]	G	Interpolation linéaire avec avance	m	+		PM-NC
G2	G	Interpolation circulaire sens horaire	m	+		PM-NC
G3	G	Interpolation circulaire sens antihoraire	m	+		PM-NC
G4	G	Arrêt temporisé, prédéterminé dans le temps	b	+		PM-NC
G5	G	Rectification en plongée oblique	b	+		PM-NC
G7	G	Mouvement compensatoire pour la rectification en plongée oblique	b	+		PM-NC
G9	G	Arrêt précis - réduction de vitesse	b	+		PM-NC
G17[6]	G	Sélection du plan de travail X/Y	m	+		PM-NC
G18	G	Sélection du plan de travail Z/X	m	+		PM-NC
G19	G	Sélection du plan de travail Y/Z	m	+		PM-NC
G25	G	Limitation inférieure de la zone de travail	b	+		PM-NC
G26	G	Limitation supérieure de la zone de travail	b	+		PM-NC
G33	G	Filetage à pas constant	m	+		PM-NC
G34	G	Filetage avec pas en progression linéaire	m	+		PM-NC
G35	G	Filetage avec pas en régression linéaire	m	+		PM-NC
G40[6]	G	Désactivation de la correction du rayon d'outil	m	+		PM-NC
G41	G	Correction du rayon d'outil à gauche du contour	m	+		PM-NC
G42	G	Correction du rayon d'outil à droite du contour	m	+		PM-NC
G53	G	Inhibition du décalage d'origine courant (non modal)	b	+		PM-NC
G54	G	1er décalage d'origine réglable	m	+		PM-NC
G55	G	2e décalage d'origine réglable	m	+		PM-NC
G56	G	3e décalage d'origine réglable	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
G57	G	4e décalage d'origine réglable	m	+		PM-NC
G58 (840D sl)	G	Décalage d'origine absolu programmable (décalage grossier)	b	+		PM-NC
G58 (828D)	G	5e décalage d'origine réglable	m	+		PM-NC
G59 (840D sl)	G	Décalage d'origine additif programmable (décalage fin)	b	+		PM-NC
G59 (828D)	G	6e décalage d'origine réglable	m	+		PM-NC
G60[6]	G	Arrêt précis - réduction de vitesse	m	+		PM-NC
G62	G	Décélération aux angles intérieurs avec correction du rayon d'outil activée (G41, G42)	m	+		PM-NC
G63	G	Taraudage avec porte-taraud compensateur	b	+		PM-NC
G64	G	Contournage	m	+		PM-NC
G70	G	Cotation en inch pour indications géométriques (longueurs)	m	+	+	PM-NC
G71[6]	G	Cotation métrique pour indications géométriques (longueurs)	m	+	+	PM-NC
G74	G	Accostage point de référence	b	+		PM-NC
G75	G	Accostage d'un point fixe	b	+		PM-NC
G90[6]	G	Introduction des cotes en valeurs absolues	m/b	+		PM-NC
G91	G	Indication de cotes relatives	m/b	+		PM-NC
G93	G	Avance en sens inverse du temps tr/min	m	+		PM-NC
G94[6]	G	Avance linéaire F en mm/min ou inch/min et degré/min	m	+		PM-NC
G95	G	Avance par tour F en mm/tr, inch/tr	m	+		PM-NC
G96	G	Avance par tour (comme pour G95) et vitesse de coupe constante	m	+		PM-NC
G97	G	Avance par tour et vitesse de rotation de broche constante (vitesse de coupe constante désactivée)	m	+		PM-NC
G110	G	Programmation du pôle par rapport à la dernière position de consigne programmée	b	+		PM-NC
G111	G	Programmation du pôle par rapport à l'origine pièce courante	b	+		PM-NC
G112	G	Programmation du pôle par rapport au dernier pôle valable	b	+		PM-NC
G140[6]	G	Sens d'accostage ARD fixé par G41/G42	m	+		PM-NC
G141	G	Sens d'accostage ARD à gauche du contour	m	+		PM-NC
G142	G	Sens d'accostage ARD à droite du contour	m	+		PM-NC
G143	G	Sens d'accostage ARD en fonction tangente	m	+		PM-NC
G147	G	Accostage en douceur en ligne droite	b	+		PM-NC
G148	G	Retrait en douceur en ligne droite	b	+		PM-NC
G153	G	Inhibition des frames courants, y compris le frame de base	b	+		PM-NC
G247	G	Accostage en douceur en quart de cercle	b	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
G248	G	Retrait en douceur en quart de cercle	b	+		PM-NC
G290[6]	G	Basculement vers mode SINUMERIK en MARCHE	m	+		FM-TM
G291	G	Basculement vers mode ISO2/3 en MARCHE	m	+		FM-TM
G331	G	Taraudage sans porte-taraud compensateur, pas positif, filetage à droite	m	+		PM-NC
G332	G	Taraudage sans porte-taraud compensateur, pas négatif, filetage à gauche	m	+		PM-NC
G335	G	Tournage d'un filetage bombé dans le sens horaire	m	+		PM-NC
G336	G	Tournage d'un filetage bombé dans le sens antihoraire	m	+		PM-NC
G340[6]	G	Accostage dans l'espace (simultanément en profondeur et dans le plan (hélice))	m	+		PM-NC
G341	G	D'abord approche dans l'axe perpendiculaire (z), puis accostage dans le plan	m	+		PM-NC
G347	G	Accostage en douceur en demi-cercle	b	+		PM-NC
G348	G	Retrait en douceur en demi-cercle	b	+		PM-NC
G450[6]	G	Arc de raccordement	m	+		PM-NC
G451	G	Point d'intersection des équidistantes	m	+		PM-NC
G460[6]	G	Activation de la détection des violations du contour pour le bloc d'accostage et de retrait	m	+		PM-NC
G461	G	Insertion d'un cercle dans le bloc CRO	m	+		PM-NC
G462	G	Insertion d'une droite dans le bloc CRO	m	+		PM-NC
G500[6]	G	Désactivation de tous les frames réglables, les frames de base sont actifs	m	+		PM-NC
G505 ... G599	G	5. ... 99. décalage d'origine réglable	m	+		PM-NC
G601[6]	G	Changement de bloc lors arrêt précis fin	m	+		PM-NC
G602	G	Changement de bloc lors arrêt précis grossier	m	+		PM-NC
G603	G	Changement de bloc en fin de bloc IPO	m	+		PM-NC
G621	G	Décélération à tous les angles	m	+		PM-NC
G641	G	Contournage avec arrondissement selon le critère de trajet (= distance de transition programmable)	m	+		PM-NC
G642	G	Contournage avec arrondissement avec prise en compte des tolérances définies	m	+		PM-NC
G643	G	Contournage avec arrondissement avec prise en compte des tolérances définies (internes au bloc)	m	+		PM-NC
G644	G	Contournage avec arrondissement utilisant la dynamique maximale possible	m	+		PM-NC
G645	G	Contournage avec arrondissement d'angles et changements de blocs tangentiels avec prise en compte de tolérances définies	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
G700	G	Cotation en inch pour indications géométriques et technologiques (longueurs, avance)	m	+	+	PM-NC
G710[6]	G	Cotation métrique pour indications géométriques et technologiques (longueurs, avance)	m	+	+	PM-NC
G810[6], ..., G819	G	Groupe G réservé pour l'utilisateur OEM		+		PM-NC
G820[6], ..., G829	G	Groupe G réservé pour l'utilisateur OEM		+		PM-NC
G931	G	Indication de l'avance par durée du déplacement, désactiver la vitesse tangentielle constante	m	+		
G942	G	Gel de l'avance linéaire et de la vitesse de coupe constante ou de la vitesse de rotation de broche	m	+		
G952	G	Gel de l'avance par tour et de la vitesse de coupe constante ou de la vitesse de rotation de broche	m	+		
G961	G	Avance linéaire (comme pour G94) et vitesse de coupe constante	m	+		PM-NC
G962	G	Avance linéaire ou avance par tour et vitesse de coupe constante	m	+		PM-NC
G971	G	Avance linéaire et vitesse de rotation de broche constante (vitesse de coupe constante désactivée)	m	+		PM-NC
G972	G	Avance linéaire ou avance par tour et vitesse de rotation de broche constante (vitesse de coupe constante désactivée)	m	+		PM-NC
G973	G	Avance par tour sans limitation de vitesse de la broche et vitesse de rotation de broche constante (G97 sans LIMS pour mode ISO)	m	+		PM-NC
GEOAX	P	Affecter de nouveaux axes de canal aux axes géométriques 1 à 3		+	-	PM-NC
GET	P	Échange de l'axe libéré entre des canaux		+	+	PM-NC
GETACTT	F	Détermination de l'outil actif parmi un groupe d'outils de même nom		+	-	FM-TM
GETACTTD	F	Détermination du numéro T correspondant à un numéro D absolu		+	-	PM-NC
GETD	P	Échange direct de l'axe entre des canaux		+	-	PM-NC
GETDNO	F	Fournit le numéro D d'un tranchant (CE) d'un outil (T)		+	-	PM-NC
GETEXET	P	Lecture du numéro T changé		+	-	FM-TM
GETFREELOC	P	Recherche d'un emplacement vide dans les magasins pour un outil donné		+	-	FM-TM
GETSELT	P	Délivrer le numéro T présélectionné		+	-	FM-TM
GETT	F	Déterminer le numéro T correspondant à un nom d'outil		+	-	FM-TM
GETTCOR	F	Lecture de longueurs d'outil ou de composantes de longueurs d'outil		+	-	PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
GETTENV	F	Lecture des numéros T, D et DL		+	-	PM-NC
GETVARAP	F	Lecture du droit d'accès à une variable système/utilisateur		+	-	PM-NC
GETVARDFT	F	Lecture de la valeur par défaut d'une variable système/utilisateur		+	-	PM-NC
GETVARLIM	F	Lecture de la valeur limite d'une variable système/utilisateur		+	-	PM-NC
GETVARPHU	F	Lecture de l'unité physique d'une variable système/utilisateur		+	-	PM-NC
GETVARTYP	F	Lecture du type de données d'une variable système/utilisateur		+	-	PM-NC
GFRAME0 ... GFRAME100	G	Activation du frame de rectification <n> de la gestion de données dans le canal	m	+		PM-NC
GOTO	K	Instruction de saut avec destination en aval puis en amont (vers fin puis vers début de programme)		+		PM-NC
GOTOB	K	Saut avec destination en amont (vers début de programme)		+		PM-NC
GOTOC	K	Comme GOTO, mais avec suppression de l'alarme 14080 "Destination de saut pas trouvée"		+		PM-NC
GOTOF	K	Saut avec destination en aval (vers fin programme)		+		PM-NC
GOTOS	K	Retour au début du programme		+		PM-NC
GP	K	Mot clé pour la programmation indirecte d'attributs de positions		+		PM-NC
GROUP_ ADDEND	C (T)	Fin du supplément de mise au point		+		PM-NC
GROUP_BEGIN	C (T)	Début du bloc de programme		+		PM-NC
GROUP_END	C (T)	Fin du bloc de programme		+		PM-NC
GWPSOF	P	Désactiver vitesse périphérique de meule constante (VPM)	b	+	-	PM-NC
GWPSON	P	Activer vitesse périphérique de meule constante (VPM)	b	+	-	PM-NC
H...	A	Sortie de fonction auxiliaire à l'AP		+	+	PM-NC, FM-B
HOLES1	C (T)	Rangée de trous		+		PM-NC
HOLES2	C (T)	Cercle de trous		+		PM-NC
I	A	Paramètres d'interpolation	b	+		PM-NC
I1	A	Coordonnée de point intermédiaire	b	+		PM-NC
IC	K	Saisie de cotes relatives	b	+		PM-NC
ICYCOF	P	Exécution de tous les blocs d'un cycle technologique dans une période d'appel de l'interpolateur après ICYCOF		+	+	FM-SA
ICYCON	P	Exécution de chaque bloc d'un cycle technologique dans une période d'appel de l'interpolateur distincte après ICYCON		+	+	FM-SA

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
ID	K	Identification pour actions synchrones modales	m	-	+	FM-SA
IDS	K	Identification pour actions synchrones statiques modales		-	+	FM-SA
IF	K	Début d'un saut conditionnel dans le programme pièce / cycle technologique		+	+	PM-NC
INDEX	F	Détermination de l'indice d'un caractère dans une chaîne de caractères		+	-	PM-NC
INICF	K	Initialisation des variables pour NEWCONF		+		PM-NC
INIPO	K	Initialisation des variables pour PowerOn		+		PM-NC
INIRE	K	Initialisation des variables pour Reset		+		PM-NC
INIT	P	Sélection d'un programme CN spécifié pour exécution dans un canal déterminé		+	-	PM-NC
INITIAL		Génération d'un fichier INI pour tous les domaines		+		PM-NC
INT	K	Type de données : valeur entière avec signe		+		PM-NC
INTERSEC	F	Calculer le point d'intersection entre deux éléments de contour		+	-	PM-NC
INVCCW	G	Déplacer développante dans le sens antihoraire	m	+		PM-NC
INVCW	G	Déplacer développante dans le sens horaire	m	+		PM-NC
INVFRAME	F	Calculer le frame inverse à partir d'un frame		+	-	FM-B
IP	K	Paramètre d'interpolation variable		+		PM-NC
IPOBRKA	P	Critère de déplacement à partir du point de déclenchement de la rampe de freinage	m	+	+	
IPOENDA	K	Fin du déplacement lors de "Arrêt interpolateur"	m	+		PM-NC
IPTRLOCK	P	Gel du début de la section de programme interdit à la recherche sur le bloc de fonction machine suivant.	m	+	-	PM-NC
IPTRUNLOCK	P	Définition de la fin de la section de programme interdit à la recherche sur le bloc courant du moment d'interruption.	m	+	-	PM-NC
IR	A	Coordonnée du centre d'un cercle (en direction X) lors du tournage d'un filetage bombé		+		PM-NC
ISAXIS	F	Vérifier si l'axe géométrique 1 indiqué comme paramètre existe		+	-	PM-NC
ISD	A	Profondeur de pénétration	m	+		PM-NC
ISFILE	F	Vérifier s'il existe un fichier dans la mémoire d'application CN		+	-	PM-NC
ISNUMBER	F	Vérifier si une chaîne de caractères peut être convertie en un nombre		+	-	PM-NC
ISOCALL	K	Appel indirect d'un programme programmé en langage ISO		+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
ISVAR	F	Vérifier si le paramètre de transfert contient une variable connue de la CN		+	-	PM-NC
J	A	Paramètres d'interpolation	b	+		PM-NC
J1	A	Coordonnée de point intermédiaire	b	+		PM-NC
JERKA	P	Activer, pour les axes programmés, le comportement à l'accélération réglé à l'aide du PM		+	-	
JERKLIM	K	Adaptation de l'à-coup d'axe maximal	m	+		PM-NC
JERKLIMA	K	Réduction ou surhaussement de l'à-coup maximal d'un axe asservi	m	+	+	PM-NC
JR	A	Coordonnée du centre d'un cercle (en direction Y) lors du tournage d'un filetage bombé		+		PM-NC
K	A	Paramètres d'interpolation	b	+		PM-NC
K1	A	Coordonnée de point intermédiaire	b	+		PM-NC
KONT	G	Contournement du contour lors de la correction d'outil	m	+		PM-NC
KONTC	G	Accostage / retrait avec polynôme à courbure continue	m	+		PM-NC
KONTT	G	Accostage / retrait avec polynôme tangentiel	m	+		PM-NC
KR	A	Coordonnée du centre d'un cercle (en direction Z) lors du tournage d'un filetage bombé		+		PM-NC
L	A	Numéro de sous-programme	b	+	+	PM-NC
LEAD	A	Angle d'avance 1re orientation d'outil 2e polynôme d'orientation	m	+		PM-NC
LEADOF	P	Couplage organe pilote d'axe désactivé		+	+	PM-NC
LEADON	P	Couplage organe pilote d'axe activé		+	+	PM-NC
LENTOAX	F	Fournit des informations sur l'affectation des longueurs L1, L2 et L3 de l'outil actif à l'abscisse, à l'ordonnée et à la valeur		+	-	PM-NC
LFOF[6]	G	Désactivation du retrait rapide pour le filetage	m	+		PM-NC
LFON	G	Activation du retrait rapide pour le filetage	m	+		PM-NC
LFPOS	G	Retrait de l'axe introduit avec POLFMASK ou POLFMLIN sur la position d'axe absolue programmée avec POLF	m	+		PM-NC
LFTXT[6]	G	Plan du déplacement de retrait rapide lors du relèvement rapide déterminé par la tangente de la trajectoire et la direction actuelle de l'outil	m	+		PM-NC
LFWP	G	Plan du déplacement de retrait rapide lors du relèvement rapide déterminé par le plan de travail actuel (G17/G18/G19)	m	+		PM-NC
LIFTFAST	K	Relèvement rapide		+		PM-NC
LIMS	K	Limitation de la vitesse de rotation pour G96/G961 et G97	m	+		PM-NC
LLI	K	Valeur limite inférieure de variables		+		PM-NC
LN	F	Logarithme naturel		+	+	PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
LOCK	P	Bloquer une action synchrone avec ID (bloquer un cycle technologique)		-	+	FM-SA
LONGHOLE	C (T)	Trou oblong		+		PM-NC
LOOP	K	Début d'une boucle sans fin		+		PM-NC

Instructions M ... R

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
M0		Arrêt programmé		+	+	PM-NC
M1		Arrêt facultatif		+	+	PM-NC
M2		Fin du programme principal (comme M30)		+	+	PM-NC
M3		Sens de rotation de broche à droite		+	+	PM-NC
M4		Sens de rotation de broche à gauche		+	+	PM-NC
M5		Arrêt de la broche		+	+	PM-NC
M6		Changement d'outil		+	+	PM-NC
M17		Fin du sous-programme		+	+	PM-NC
M19		Positionnement de la broche sur la position inscrite dans le SD43240		+	+	PM-NC
M30		Fin du programme principal (comme M2)		+	+	PM-NC
M40		Changement automatique de rapport de boîte de vitesses		+	+	PM-NC
M41 ... M45		Rapport de boîte de vitesses 1 ... 5		+	+	PM-NC
M70		Basculement dans mode axe		+	+	PM-NC
MASLDEF	P	Définition du couplage d'axes maître/esclave		+	+	PM-NC
MASLDEL	P	Découplage maître/esclave et suppression de la définition du couplage		+	+	PM-NC
MASLOF	P	Arrêter un couplage temporaire		+	+	PM-NC
MASLOFS	P	Désactivation d'un couplage temporaire avec arrêt automatique de l'axe esclave		+	+	PM-NC
MASLON	P	Enclencher un couplage temporaire		+	+	PM-NC
MATCH	F	Rechercher une chaîne dans une chaîne		+	-	PM-NC
MAXVAL	F	Valeur supérieure de deux variables (fonction trigonométrique)		+	+	PM-NC
MCALL	K	Appel modal d'un sous-programme		+		PM-NC
MEAC	K	Mesure axiale continue sans effacement de la distance restant à parcourir	b	+	+	PM-NC
MEAFRAME	F	Calcul d'un frame à partir de points de mesure		+	-	PM-NC
MEAS	A	Mesure avec effacement de la distance restant à parcourir	b	+		PM-NC
MEASA	K	Mesure axiale avec effacement de la distance restant à parcourir	b	+	+	PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
MEASURE	F	Méthode de calcul pour la mesure de pièce et la mesure d'outil		+	-	FM-TE
MEAW	A	Mesure sans effacement de la distance restant à parcourir	b	+		PM-NC
MEAWA	K	Mesure axiale sans effacement de la distance restant à parcourir	b	+	+	PM-NC
MI	K	Accès aux données frame : fonction miroir		+		PM-NC
MINDEX	F	Détermination de l'indice d'un caractère dans une chaîne de caractères		+	-	PM-NC
MINVAL	F	Valeur inférieure de deux variables (fonction trigonométrique)		+	+	PM-NC
MIRROR	G	Fonction miroir programmable	b	+		PM-NC
MMC	P	Appel interactif de boîtes de dialogue à partir du programme pièce pour affichage sur l'IHM		+	-	PM-NC
MOD	K	Division modulo		+		PM-NC
MODAXVAL	F	Détermination de la position modulo d'un axe rotatif modulo		+	-	PM-NC
MOV	K	Démarrage de l'axe de positionnement		-	+	FM-SA
MOVT	A	Indication du point final d'un déplacement dans le sens de l'outil				FM-B
MSG	P	Messages programmables	m	+	-	PM-NC
MVTOOL	P	Instruction de langage pour le déplacement d'un outil		+	-	FM-TM
N	A	Numéro de bloc CN complémentaire		+		PM-NC
NAMETOINT	F	Détermination de l'indice de variables système		+		PM-NC
NC	K	Spécification du domaine de validité de données		+		PM-NC
NEWCONF	P	Application des paramètres machine modifiés (correspond à "Activation des paramètres machine")		+	-	PM-NC
NEWMT	F	Création d'un nouveau multitool		+	-	FM-TM
NEWT	F	Création d'un nouvel outil		+	-	FM-TM
NORM[6]	G	Positionnement à la normale au point de départ / final en correction d'outil	m	+		PM-NC
NOT	K	NON logique (Négation)		+		PM-NC
NPROT	P	Activation/désactivation d'une zone de protection spécifique à la machine		+	-	PM-NC
NPROTDEF	P	Définition d'une zone de protection spécifique à la machine		+	-	PM-NC
NUMBER	F	Convertir une chaîne de caractères en un nombre		+	-	PM-NC
OEMIPO1	G	Interpolation OEM 1	m	+		PM-NC
OEMIPO2	G	Interpolation OEM 2	m	+		PM-NC
OF	K	Mot-clé d'un branchement CASE		+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
OFFN	A	Surépaisseur d'usinage par rapport au contour programmé	m	+		PM-NC
OMA1	A	Adresse OEM 1	m	+		PM-NC
OMA2	A	Adresse OEM 2	m	+		PM-NC
OMA3	A	Adresse OEM 3	m	+		PM-NC
OMA4	A	Adresse OEM 4	m	+		PM-NC
OMA5	A	Adresse OEM 5	m	+		PM-NC
OR	K	Opérateur logique, combinaison OU		+		PM-NC
ORIAXES	G	Interpolation linéaire des axes machine ou des axes d'orientation	m	+		PM-NC
ORIAXPOS	G	Angles d'orientation définis par axes virtuels d'orientation avec positions d'axe rotatif	m	+		PM-NC
ORIC[6]	G	Les changements d'orientation aux angles saillants sont superposés au bloc circulaire à insérer	m	+		PM-NC
ORICONCCW	G	Interpolation sur une enveloppe de cercle dans le sens antihoraire	m	+		PM-NC, FM-TR
ORICONCW	G	Interpolation sur une enveloppe de cercle dans le sens horaire	m	+		PM-NC, FM-TR
ORICONIO	G	Interpolation sur une enveloppe de cercle avec indication d'une orientation intermédiaire	m	+		PM-NC, FM-TR
ORICONTO	G	Interpolation sur une surface circulaire avec transition tangentielle (indication de l'orientation finale)	m	+		PM-NC, FM-TR
ORICURVE	G	Interpolation de l'orientation avec indication du mouvement de deux points de contact de l'outil	m	+		PM-NC, FM-TR
ORID	G	Les changements d'orientation se font avant l'exécution du bloc circulaire inséré	m	+		PM-NC
ORIEULER[6]	G	Orientation définie par angles d'Euler	m	+		PM-NC
ORIMKS	G	Orientation de l'outil dans le système de coordonnées machine	m	+		PM-NC
ORIPATH	G	Orientation de l'outil par rapport à la trajectoire	m	+		PM-NC
ORIPATHS	G	Orientation de l'outil par rapport à la trajectoire, lissage de coude dans le tracé d'orientation	m	+		PM-NC
ORIPLANE	G	Interpolation dans un plan (correspond à ORIVECT) Interpolation circulaire de grand rayon	m	+		PM-NC
ORIRESET	P	Position initiale de l'orientation de l'outil avec jusqu'à 3 axes d'orientation		+	-	PM-NC
ORIROTA[6]	G	Angle de rotation rapporté à un sens de rotation en indication absolue	m	+		PM-NC
ORIROTC	G	Vecteur de rotation tangentiel à la tangente à la trajectoire	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
ORIROTR	G	Angle de rotation relatif par rapport au plan entre orientation de départ et orientation finale	m	+		PM-NC
ORIROTT	G	Angle de rotation relatif par rapport à la modification du vecteur d'orientation	m	+		PM-NC
ORIRPY	G	Orientation définie par angles RPY (XYZ)	m	+		PM-NC
ORIRPY2	G	Orientation définie par angles RPY (ZYX)	m	+		PM-NC
ORIS	A	Modification d'orientation	m	+		PM-NC
ORISOF[6]	G	Lissage du tracé d'orientation désactivé	m	+		PM-NC
ORISOLH	F	Calcul des orientations		+		PM-NC
ORISON	G	Lissage du tracé d'orientation activé	m	+		PM-NC
ORIVECT[6]	G	Interpolation circulaire de grand rayon (identique avec ORIPLANE)	m	+		PM-NC
ORIVIRT1	G	Angles d'orientation définis par axes virtuels d'orientation (définition 1)	m	+		PM-NC
ORIVIRT2	G	Angles d'orientation définis par axes virtuels d'orientation (définition 1)	m	+		PM-NC
ORIWKS[6]	G	Orientation de l'outil dans le système de coordonnées pièce	m	+		PM-NC
OS	K	Activation/désactivation de l'oscillation		+		PM-NC
OSB	K	Oscillation : point de départ	m	+		PM-NC
OSC	G	Lissage constant de l'orientation de l'outil	m	+		PM-NC
OSCILL	K	Axis : 1 à 3 axes de pénétration	m	+		PM-NC
OSCTRL	K	Options d'oscillation	m	+		PM-NC
OSD	G	Arrondissement de l'orientation de l'outil via la spécification de la longueur d'arrondissement avec SD	m	+		PM-NC
OSE	K	Point d'arrêt de l'oscillation	m	+		PM-NC
OSNSC	K	Oscillation : nombre d'extractions par étincelage	m	+		PM-NC
OSOF[6]	G	Désactivation du lissage de l'orientation de l'outil	m	+		PM-NC
OSP1	K	Oscillation : point d'inversion gauche	m	+		PM-NC
OSP2	K	Oscillation : point d'inversion droit	m	+		PM-NC
OSS	G	Lissage de l'orientation de l'outil en fin de bloc	m	+		PM-NC
OSSE	G	Lissage de l'orientation de l'outil en début et en fin de bloc	m	+		PM-NC
OST	G	Arrondissement de l'orientation de l'outil via la spécification de la tolérance angulaire en degré avec SD (écart maximal de l'allure de variation de l'orientation programmée)	m	+		PM-NC
OST1	K	Oscillation : point d'arrêt au point d'inversion gauche	m	+		PM-NC
OST2	K	Oscillation : point d'arrêt au point d'inversion droit	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
OTOL	A	Tolérance d'orientation pour fonctions de compression, lissage de l'orientation et modes d'arrondissement	m	+	-	PM-NC
OTOLG0	A	Tolérance d'orientation pour les déplacements à vitesse rapide	m	+	-	PM-NC
OVR	K	Correction de vitesse de rotation	m	+		PM-NC
OVRA	K	Correction de vitesse de rotation axiale	m	+	+	PM-NC
OVRRAP	K	Correction du rapide	m	+		PM-NC
P	A	Nombre d'exécutions du sous-programme		+		PM-NC
PACCLIM	A	Adaptation de l'accélération maximale sur trajectoire	m	+	-	PM-NC
PAROT	G	Orientation du système de coordonnées pièce sur la pièce	m	+		PM-NC
PAROTOF[6]	G	Désactiver la rotation de frame par rapport à l'outil	m	+		PM-NC
PCALL	K	Appel d'un sous-programme avec chemin d'accès absolu et transfert de paramètres.		+		PM-NC
PDELAYOF	G	Poinçonnage avec temporisation désactivé	m	+		PM-NC
PDELAYON[6]	G	Poinçonnage avec temporisation activé	m	+		PM-NC
PHI	K	Angle de rotation de l'orientation autour de l'axe d'orientation du cône		+		PM-NC
PHU	K	Unité physique d'une variable		+		PM-NC
PL	A	1er spline B distance entre les pôles 2e interpolation polynomiale longueur de l'intervalle des paramètres pour l'interpolation polynomiale	b	+		PM-NC
PM	K	Par minute		+		PM-NC
PO	K	Coefficient polynomial pour l'interpolation polynomiale	b	+		PM-NC
POCKET3	C (T)	Fraisage d'une poche rectangulaire		+		PM-NC
POCKET4	C (T)	Fraisage d'une poche circulaire		+		PM-NC
POLF	K	Position de retrait LIFTFAST	m	+		PM-NC
POLFA	P	Lancement de la position de retrait des axes individuels avec $AA_ESR_TRIGGER	m	+	+	PM-NC
POLFMASK	P	Libération des axes pour le retrait sans corrélation entre les axes	m	+	-	PM-NC
POLFMLIN	P	Libération des axes pour le retrait avec corrélation linéaire entre les axes	m	+	-	PM-NC
POLY	G	Interpolation polynomiale	m	+		PM-NC
POLYPATH	P	Interpolation polynomiale sélectionnable pour les deux groupes d'axes AXIS ou VECT	m	+	-	PM-NC
PON	G	Poinçonnage ACTIVE	m	+		PM-NC
PONS	G	Poinçonnage activé à la période d'appel de l'interpolateur	m	+		PM-NC
POS	K	Positionnement d'un axe		+	+	PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
POSA	K	Positionnement d'un axe sur plusieurs blocs		+	+	PM-NC
POSM	P	Positionnement d'un magasin		+	-	FM-TM
POSMT	P	Positionnement du multitool sur le porte-outil au numéro d'emplacement		+	-	FM-TM
POSP	K	Positionnement en trajets partiels (oscillation)		+		PM-NC
POSRANGE	F	Déterminer si la position de consigne courante interpolée d'un axe se trouve dans une fenêtre autour d'une position de référence prescrite		+	+	FM-SA
POT	F	Puissance 2 (fonction arithmétique)		+	+	PM-NC
PR	K	Par rotation		+		PM-NC
PREPRO	PA	Identifier les sous-programmes avec prétraitement		+		PM-NC
PRESETON	P	Forçage de valeur réelle avec perte de l'état de référencement		+	+	PM-NC
PRESETONS	P	Forçage de valeur réelle sans perte de l'état de référencement		+	+	PM-NC
PRIO	K	Mot-clé pour la définition de priorités lors du traitement d'interruptions		+		PM-NC
PRLOC	K	Initialisation des variables lors d'un Reset uniquement après modification locale		+		PM-NC
PROC	K	Première instruction d'un programme		+		PM-NC
PROTA	P	Demander un recalcul du modèle de collision		+		PM-NC
PROTD	F	Calcul de la distance entre deux zones de protection		+		PM-NC
PROTS	P	Réglage de l'état des zones de protection		+		PM-NC
PSI	K	Angle d'ouverture du cône		+		PM-NC
PTP	G	Déplacement point à point (PTP)	m	+		PM-NC
PTPG0	G	Déplacement point à point uniquement pour G0, sinon déplacement avec interpolation CP	m	+		PM-NC
PTPWOC	G	Déplacement point à point sans déplacement de compensation causés par un changement d'orientation	m	+		PM-NC
PUNCHACC	P	Accélération asservie au grignotage		+	-	PM-NC
PUTFTOC	P	Correction fine d'outil pour dressage simultané		+	-	PM-NC
PUTFTOCF	P	Correction fine d'outil selon fonction définie avec FCTDEF pour dressage simultané		+	-	PM-NC
PW	A	Spline B (poids ponctuel)	b	+		PM-NC
QU	K	Sortie rapide de fonction supplémentaire (auxiliaire)		+		PM-NC
R...	A	Paramètre de calcul également comme descripteur d'adresse paramétrable et avec extension numérique		+		PM-NC
RAC	K	Programmation au rayon spécifique à l'axe non modale absolue	b	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
RDISABLE	P	Blocage de l'introduction via l'interface.		-	+	FM-SA
READ	P	Permet de lire une ou plusieurs lignes dans le fichier indiqué, les informations lues étant rangées dans un tableau		+	-	PM-NC
REAL	K	Type de données : variable à virgule flottante avec signe (nombres réels)		+		PM-NC
REDEF	K	Redéfinition de variables système, variables utilisateur et instruction de langage CN		+		PM-NC
RELEASE	P	Libérer axes machine pour l'échange		+	+	PM-NC
REP	K	Mot-clé pour initialisation de tous les éléments d'un tableau avec la même valeur		+		PM-NC
REPEAT	K	Itération d'une boucle		+		PM-NC
REPEATB	K	Itération d'une ligne de programme		+		PM-NC
REPOSA	G	Réaccostage linéaire du contour avec tous les axes	b	+		PM-NC
REPOSH	G	Réaccostage du contour avec demi-cercle	b	+		PM-NC
REPOSHA	G	Réaccostage du contour avec tous les axes ; axes géométriques en demi-cercle	b	+		PM-NC
REPOSL	G	Réaccostage linéaire du contour	b	+		PM-NC
REPOSQ	G	Réaccostage du contour en quart de cercle	b	+		PM-NC
REPOSQA	G	Réaccostage linéaire du contour avec tous les axes ; axes géométriques en quart de cercle	b	+		PM-NC
RESETMON	P	Instruction de langage pour activation de la consigne		+	-	FM-TM
RET	P	Fin de sous-programme		+	+	PM-NC
RETB	P	Fin de sous-programme		+	+	PM-NC
RIC	K	Programmation au rayon spécifique à l'axe non modale relative	b	+		PM-NC
RINDEX	F	Détermination de l'indice d'un caractère dans une chaîne de caractères		+	-	PM-NC
RMB	G	Réaccostage du point de début de bloc	m	+		PM-NC
RMBBL	G	Réaccostage du point de début de bloc	b	+		PM-NC
RME	G	Réaccostage du point final du bloc	m	+		PM-NC
RMEBL	G	Réaccostage du point final du bloc	b	+		PM-NC
RMI[6]	G	Réaccostage du point d'interruption	m	+		PM-NC
RMIBL [6]	G	Réaccostage du point d'interruption	b	+		PM-NC
RMN	G	Réaccostage du point de contour le plus proche	m	+		PM-NC
RMNBL	G	Réaccostage du point de contour le plus proche	b	+		PM-NC
RND	A	Arrondir un angle	b	+		PM-NC
RNDM	A	Arrondissement modal	m	+		PM-NC
ROT	G	Rotation programmable	b	+		PM-NC
ROTS	G	Programmation de rotations de frames avec des angles solides	b	+		PM-NC
ROUND	F	Arrondissement des décimales		+	+	PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
ROUNDUP	F	Arrondissement d'une valeur d'entrée		+	+	PM-NC
RP	A	Rayon polaire	m/b	+		PM-NC
RPL	A	Rotation dans le plan	b	+		PM-NC
RT	K	Paramètre d'accès aux données frame : rotation		+		PM-NC
RTLIOF	G	G0 sans interpolation linéaire (interpolation axiale individuelle)	m	+		PM-NC
RTLION[6]	G	G0 avec interpolation linéaire	m	+		PM-NC

Instructions S ... Z

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
S	A	Vitesse de rotation de broche (pour G4, G96/G961, autre signification)	m/b	+	+	PM-NC
SAVE	PA	Attribut pour sauvegarde d'informations en cas d'appel de sous-programme		+		PM-NC
SBLOF	P	Inhibition du bloc par bloc		+	-	PM-NC
SBLON	P	Annulation de l'inhibition du bloc par bloc		+	-	PM-NC
SC	K	Paramètre d'accès aux données frame : facteur d'échelle		+		PM-NC
SCALE	G	Facteur d'échelle programmable	b	+		PM-NC
SCC	K	Affectation sélective d'un axe transversal pour G96/G961/G962. Les descripteurs d'axes peuvent être des axes géométriques, des axes de canal ou des axes machine.		+		PM-NC
SCPARA	K	Programmation du jeu de paramètres servo		+	+	PM-NC
SD	A	Degré de spline	b	+		PM-NC
SET	K	Mot-clé pour initialisation de tous les éléments d'un tableau avec les valeurs mentionnées		+		PM-NC
SETAL	P	Activation d'une alarme		+	+	PM-NC
SETDNO	F	Affectation du numéro D du tranchant (CE) d'un outil (T)		+	-	PM-NC
SETINT	K	Définir la routine d'interruption qui doit être activée lorsqu'une entrée CN devient active		+		PM-NC
SETM	P	Définition de repères dans le canal spécifique		+	+	PM-NC
SETMS	P	Retour à la broche maître définie dans le paramètre machine		+	-	PM-NC
SETMS(n)	P	La broche n doit devenir broche maître		+		PM-NC
SETMTH	P	Définir le numéro du porte-outil maître		+	-	FM-TM
SETPIECE	P	Tenir compte du nombre de pièces pour tous les outils assignés à la broche		+	-	FM-TM
SETTA	P	Activation d'un outil du groupe d'usure		+	-	FM-TM

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
SETTCOR	F	Modification de composantes d'outils avec prise en compte de l'ensemble des conditions marginales		+	-	PM-NC
SETTIA	P	Désactivation d'un outil du groupe d'usure		+	-	FM-TM
SF	A	Décalage du point de départ pour filetage	m	+		PM-NC
SIN	F	Sinus (fonction trigonométrique)		+	+	PM-NC
SIRELAY	F	Activer les fonctions de sécurité paramétrées avec SIRELIN, SIRELOUT et SIRELTIME		-	+	FM-SI
SIRELIN	P	Initialiser les grandeurs d'entrée du bloc fonctionnel		+	-	FM-SI
SIRELOUT	P	Initialiser les grandeurs de sortie du bloc fonctionnel		+	-	FM-SI
SIRELTIME	P	Initialiser les temporisations du bloc fonctionnel		+	-	FM-SI
SLOT1	C (T)	Rainure rectiligne		+		PM-NC
SLOT2	C (T)	Rainure circulaire		+		PM-NC
SOFT	G	Accélération sur trajectoire avec limitation des à-coups	m	+		PM-NC
SOFTA	P	Activation de l'accélération d'axe avec limitation des à-coups pour les axes programmés		+	-	PM-NC
SON	G	Grignotage ACTIVÉ	m	+		PM-NC
SONS	G	Grignotage activé à la période d'appel de l'interpolateur	m	+		PM-NC
SPATH [6]	G	La référence trajectoire pour axes FGROUP est la longueur d'arc	m	+		PM-NC
SPCOF	P	Commutation la broche maître ou broche avec numéro n de l'asservissement de position sur la régulation de la vitesse de rotation	m	+	-	PM-NC
SPCON	P	Commutation la broche maître ou broche avec numéro n de la régulation de la vitesse de rotation sur l'asservissement de position	m	+	-	PM-NC
SPI	F	Conversion du numéro de broche en descripteur d'axe		+	-	PM-NC
SPIF1 [6]	G	E/S CN rapides pour poinçonnage/grignotage, octet 1	m	+		FM-TE
SPIF2	G	E/S CN rapides pour poinçonnage/grignotage, octet 2	m	+		FM-TE
SPLINEPATH	P	Détermination du groupe Spline		+	-	PM-NC
SPN	A	Nombre de distances partielles par bloc	b	+		PM-NC
SPOF [6]	G	Coup désactivé, poinçonnage, grignotage désactivés	m	+		PM-NC
SPOS	K	Position de broche	m	+	+	PM-NC
SPOSA	K	Position de broche au-delà des limites de bloc	m	+		PM-NC
SPP	A	Longueur d'une distance partielle	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
SPRINT	F	Retourne une chaîne de caractères d'entrée formatée		+		PM-NC
SQRT	F	Racine carrée (fonction arithmétique) (square root)		+	+	PM-NC
SR	A	Distance de retrait de l'oscillation pour une action synchrone	b	+		PM-NC
SRA	K	Distance de retrait axiale de l'oscillation suite à une action externe pour une action synchrone	m	+		PM-NC
ST	A	Durée des passes à lécher oscillatoires pour une action synchrone	b	+		PM-NC
STA	K	Durée des passes à lécher oscillatoires axiales pour une action synchrone	m	+		PM-NC
START	P	Lancement des programmes sélectionnés simultanément dans plusieurs canaux, à partir du programme en cours		+	-	PM-NC
STARTFIFO [6]	G	Exécution ; remplissage en parallèle du tampon d'exécution	m	+		PM-NC
STAT		Position des articulations	b	+		PM-NC
STOLF	A	Facteur de tolérance G0	m	+		PM-NC
STOPFIFO	G	Arrêt de l'exécution remplissage du tampon d'exécution jusqu'à la reconnaissance de STARTFIFO, la saturation du tampon d'exécution ou la fin du programme	m	+		PM-NC
STOPRE	P	Arrêt du prétraitement des blocs jusqu'à ce que tous les blocs préparés aient été exécutés		+	-	PM-NC
STOPREOF	P	Annulation de l'arrêt du prétraitement des blocs		-	+	FM-SA
STRING	K	Type de données : chaîne de caractères		+		PM-NC
STRINGIS	F	Contrôle le langage CN existant et si les noms de cycles CN, les variables utilisateurs, les macros et les noms d'étiquettes appartenant spécifiquement à cette instruction existent, sont valables, définis ou encore actifs.		+	-	PM-NC
STRLEN	F	Déterminer la longueur d'une chaîne de caractères		+	-	PM-NC
SUBSTR	F	Détermination de l'indice d'un caractère dans une chaîne de caractères		+	-	PM-NC
SUPA	G	Inhibition du décalage d'origine actuel, y compris les décalages programmés, les frames système, les décalages par manivelle (DRF), le décalage externe d'origine et le déplacement forcé	b	+		PM-NC
SUPD	G	Inhibition des corrections d'outil actives	b	+	-	PM-NC
SVC	K	Vitesse de coupe d'outil	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
SYNFCT	P	Évaluation d'un polynôme en fonction d'une condition dans une action synchrone au déplacement		-	+	FM-SA
SYNR	K	Lecture synchrone de la variable, c'est-à-dire au moment de l'exécution		+		PM-NC
SYNRW	K	Lecture et écriture synchrones de la variable, c'est-à-dire au moment de l'exécution		+		PM-NC
SYNW	K	Écriture synchrone de la variable, c'est-à-dire au moment de l'exécution		+		PM-NC
T	A	Appel d'outil (changement uniquement s'il est fixé dans les paramètres machine ; sinon instruction M6 nécessaire)		+		PM-NC
TAN	F	Tangente (fonction trigonométrique)		+	+	PM-NC
TANG	P	Positionnement tangentiel : définition du couplage		+	-	PM-NC
TANGDEL	P	Positionnement tangentiel : suppression du couplage		+	-	PM-NC
TANGOF	P	Positionnement tangentiel : désactivation du couplage		+	-	PM-NC
TANGON	P	Positionnement tangentiel : activation du couplage		+	-	PM-NC
TCA (828D: _TCA)	P	Sélection /changement d'outil indépendant de l'état de l'outil		+	-	FM-TM
TCARR	A	Appeler un organe porte-outil (numéro "m")		+		PM-NC
TCI	P	Mettre en place l'outil se trouvant à l'emplacement tampon dans le magasin		+	-	FM-TM
TCOABS [6]	G	Détermination des composantes de longueur d'outil à partir de l'orientation actuelle de l'outil	m	+		PM-NC
TCOFR	G	Détermination des composantes de longueur d'outil à partir de l'orientation du frame actif.	m	+		PM-NC
TCOFRX	G	Lors de la sélection d'outil, détermination de l'orientation de l'outil d'un frame activé, outil pointé en direction X	m	+		PM-NC
TCOFRY	G	Lors de la sélection d'outil, détermination de l'orientation de l'outil d'un frame activé, outil pointé en direction Y	m	+		PM-NC
TCOFRZ	G	Lors de la sélection d'outil, détermination de l'orientation de l'outil d'un frame activé, outil pointé en direction Z	m	+		PM-NC
THETA	A	Angle de rotation	b	+		PM-NC
TILT	A	Angle latéral	m	+		PM-NC
TLIFT	P	Positionnement tangentiel : activation de la création de blocs intermédiaires		+	-	PM-NC
TML	P	Sélection d'un outil à l'aide du numéro d'emplacement de magasin		+	-	FM-TM

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
TMOF	P	Désactivation de la surveillance d'outil		+	-	PM-NC
TMON	P	Activation de la surveillance d'outil		+	-	PM-NC
TO	K	Désigne la valeur finale d'une boucle de comptage FOR		+		PM-NC
TOFF	A	Offset de longueur d'outil dans la direction de la composante de longueur d'outil agissant parallèlement à l'axe géométrique indiqué dans l'indice	m	+		PM-NC
TOFFL	A	Offset de longueur d'outil dans la direction de la composante de longueur d'outil L1, L2 ou L3	m	+		PM-NC
TOFFLR	A	Offset de rayon d'outil et offset de longueur d'outil simultanés	m	+		PM-NC
TOFFOF	P	Initialisation de la correction de la longueur d'outil en ligne		+	-	PM-NC
TOFFON	P	Activation de la correction en ligne de la longueur d'outil		+	-	PM-NC
TOFFR	A	Offset de rayon d'outil	m	+		PM-NC
TOFRAME	G	Orienter l'axe Z du SCP parallèlement à l'orientation de l'outil par rotation du frame	m	+		PM-NC
TOFRAMEX	G	Orienter l'axe X du SCP parallèlement à l'orientation de l'outil par rotation du frame	m	+		PM-NC
TOFRAMEY	G	Orienter l'axe Y du SCP parallèlement à l'orientation de l'outil par rotation du frame	m	+		PM-NC
TOFRAMEZ	G	Identique à TOFRAME	m	+		PM-NC
TOLOWER	F	Transformer les caractères d'une chaîne de caractères en minuscules		+	-	PM-NC
TOOLENV	F	Enregistrer tous les états actuels significatifs pour l'exploitation des données d'outil mémorisées dans la mémoire		+	-	PM-NC
TOOLGNT	F	Détermination du nombre d'outils d'un groupe d'outils		+	-	FM-TM
TOOLGT	F	Détermination du numéro T d'un outil issu d'un groupe d'outils		+	-	FM-TM
TOROT	G	Orienter l'axe Z du SCP parallèlement à l'orientation de l'outil par rotation du frame	m	+		PM-NC
TOROTOF [6]	G	Rotations de frames en direction de l'outil désactivées	m	+		PM-NC
TOROTX	G	Orienter l'axe X du SCP parallèlement à l'orientation de l'outil par rotation du frame	m	+		PM-NC
TOROTY	G	Orienter l'axe Y du SCP parallèlement à l'orientation de l'outil par rotation du frame	m	+		PM-NC
TOROTZ	G	Identique à TOROT	m	+		PM-NC
TOUPPER	F	Transformer les caractères d'une chaîne de caractères en majuscules		+	-	PM-NC
TOWBCS	G	Valeurs d'usure dans le système de coordonnées de base (SCB)	m	+		PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
TOWKCS	G	Valeurs d'usure dans le système de coordonnées de la tête d'outil avec transformation cinétique (diffère du SCM avec rotation d'outil)	m	+		PM-NC
TOWMCS	G	Valeurs d'usure dans le système de coordonnées machine (SCM)	m	+		PM-NC
TOWSTD [6]	G	Valeur de position de base pour corrections sur la longueur d'outil	m	+		PM-NC
TOWTCS	G	Valeurs d'usure dans le système de coordonnées outil (point de référence du porte-outil T sur le mandrin du porte-outil)	m	+		PM-NC
TOWWCS	G	Valeurs d'usure dans le système de coordonnées pièce (SCP)	m	+		PM-NC
TR	K	Composante de décalage d'une variable de frame		+		PM-NC
TRAANG	P	Transformation axe incliné		+	-	PM-NC
TRACON	P	Transformation en cascade		+	-	PM-NC
TRACYL	P	Cylindre : transformation sur surfaces latérales		+	-	PM-NC
TRAFOOF	P	Désactiver les transformations actives dans le canal		+	-	PM-NC
TRAFOON	P	Activation de la transformation définie avec des chaînes cinématiques		+	-	PM-NC
TRAILOF	P	Déplacement conjugué synchrone à l'axe désactivé		+	+	PM-NC
TRAILON	P	Activation du déplacement conjugué synchrone à l'axe		+	+	PM-NC
TRANS	G	Décalage d'origine absolu programmable	b	+		PM-NC
TRANSMIT	P	Transformation polaire (usinage sur face frontale)		+	-	PM-NC
TRAORI	P	Transformation 4, 5 axes, transformation générique		+	-	PM-NC
TRUE	K	Constante logique : vrai		+		PM-NC
TRUNC	F	Troncature des décimales		+	+	PM-NC
TU		Angle d'axe	b	+		PM-NC
TURN	A	Nombre de tours pour une hélice	b	+		PM-NC
ULI	K	Valeur limite supérieure de variables		+		PM-NC
UNLOCK	P	Libérer une action synchrone avec ID (poursuivre un cycle technologique)		-	+	FM-SA
UNTIL	K	Condition de sortie d'une boucle REPEAT		+		PM-NC
UPATH	G	La référence trajectoire pour axes FGROUP est paramètre de courbe	m	+		PM-NC
VAR	K	Mot-clé : type du transfert de paramètres		+		PM-NC
VELOLIM	K	Adaptation de la vitesse d'axe ou de la vitesse de broche maximale	m	+	-	PM-NC
VELOLIMA	K	Réduction ou dépassement de la vitesse maximale d'un axe asservi	m	+	+	PM-NC

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
WAITC	P	Attendre que le critère de changement de bloc de couplage soit rempli pour les axes/broches		+	-	PM-NC
WAITE	P	Attente de la fin du programme dans un autre canal.		+	-	PM-NC
WAITENC	P	Attente des positions d'axe synchronisées ou restaurées		+	-	PM-NC
WAITM	P	Attente de la marque dans canal indiqué ; terminer le bloc précédent par un arrêt précis.		+	-	PM-NC
WAITMC	P	Attente de la marque dans canal indiqué ; arrêt précis uniquement si les autres canaux n'ont pas encore atteint la marque.		+	-	PM-NC
WAITP	P	Attente de la fin du déplacement de l'axe de positionnement		+	-	PM-NC
WAITS	P	Attente jusqu'à ce que la position de la broche soit atteinte		+	-	PM-NC
WALCS0 [6]	G	Limitation de la zone de travail SCP désélectionnée	m	+	-	PM-NC
WALCS1	G	Groupe de limitations de la zone de travail SCP 1 actif	m	+	-	PM-NC
WALCS2	G	Groupe de limitations de la zone de travail SCP 2 actif	m	+	-	PM-NC
WALCS3	G	Groupe de limitations de la zone de travail SCP 3 actif	m	+	-	PM-NC
WALCS4	G	Groupe de limitations de la zone de travail SCP 4 actif	m	+	-	PM-NC
WALCS5	G	Groupe de limitations de la zone de travail SCP 5 actif	m	+	-	PM-NC
WALCS6	G	Groupe de limitations de la zone de travail SCP 6 actif	m	+	-	PM-NC
WALCS7	G	Groupe de limitations de la zone de travail SCP 7 actif	m	+	-	PM-NC
WALCS8	G	Groupe de limitations de la zone de travail SCP 8 actif	m	+	-	PM-NC
WALCS9	G	Groupe de limitations de la zone de travail SCP 9 actif	m	+	-	PM-NC
WALCS10	G	Groupe de limitations de la zone de travail SCP 10 actif	m	+	-	PM-NC
WALIMOF	G	Désactivation de la limitation de la zone de travail SCB	m	+	-	PM-NC
WALIMON [6]	G	Activation de la limitation de la zone de travail SCB	m	+	-	PM-NC
WHEN	K	L'action est exécutée de manière unique lorsque la condition est remplie une fois.		-	+	FM-SA
WHENEVER	K	L'action est exécutée de manière cyclique dans chaque cycle d'interpolation lorsque la condition est remplie.		-	+	FM-SA

Instruction	Type [1]	Signification	E [2]	PP [3]	AS [4]	Description, voir [5]
[1] [2] [3] [4] [5] **Explications voir Légende (Page 1281).**						
WHILE	K	Début d'une boucle WHILE		+		PM-NC
WRITE	P	Écrire un texte dans le système de fichiers. Ajoute un bloc à la fin du fichier indiqué.		+	-	PM-NC
WRTPR	P	Écrire la chaîne de caractères dans la variable OPI		+	-	PM-NC
X	A	Nom d'axe	m/b	+	+	PM-NC
XOR	O	OU exclusif logique		+		PM-NC
Y	A	Nom d'axe	m/b	+	+	PM-NC
Z	A	Nom d'axe	m/b	+	+	PM-NC

Légende

[1] Type d'instruction :

A Adresse

Descripteur au quel une valeur a été attribuée (par ex. OVR=10). Certaines adresses activent ou désactivent également des fonctions sans attribution de valeur (par ex. CPLON et CPLOF).

C (A) Cycle AST

Programme CN défini par défaut pour la réoptimisation automatique avec AST(= Optimisation Servo automatique). L'adaptation à la situation d'optimisation concrète s'effectue à l'aide des paramètres transmis au cycle lors de l'appel.

C (M) Cycle de mesure

Programme CN prédéfini, contenant un processus de mesure, par ex. la détermination du diamètre intérieur d'une pièce cylindrique, pour être applicable de manière générale. L'adaptation à la situation de mesure concrète s'effectue à l'aide des paramètres transmis au cycle lors de l'appel.

C (T) Cycle technologique

Programme CN prédéfini dans lequel un certain cycle (procédé d'usinage), par ex. le taraudage ou le fraisage d'une poche, est programmé pour être applicable de manière générale. L'adaptation à la situation d'usinage concrète s'effectue à l'aide des paramètres transmis au cycle lors de l'appel.

F Fonction prédéfinie (fournit une valeur de retour)

L'appel de la fonction prédéfinie peut figurer sous forme d'opérande dans l'expression.

G Instruction G

Les instructions G sont réparties en groupes. Dans un bloc, on ne peut écrire qu'une seule instruction G d'un groupe. Une instruction G peut avoir un effet modal (elle s'applique jusqu'à ce qu'elle soit remplacée par une autre instruction du même groupe), ou un effet non modal, auquel cas elle s'applique uniquement dans le bloc dans lequel elle figure.

K Mot-clé

Descripteur qui détermine la syntaxe d'un bloc. Aucune valeur n'est attribuée à un mot clé et un mot clé ne permet pas d'activer ou de désactiver des fonctions CN.

Exemples : conditions (IF, ELSE, ENDIF, WHEN, ...), sauts (GOTOB, GOTO, RET ...)

O Opérateur

Opérateur d'une opération mathématique, logique ou relationnelle

P Procédure prédéfinie (ne fournit pas de valeur de retour)

PA Attribut de programme

Les attributs de programme figurent à la fin de la ligne de définition d'un sous-programme :
`PROC <nom de programme>(...) <attribut de programme>`

Ils déterminent le comportement lors de l'exécution du sous-programme.

[2] Prise d'effet de l'instruction :

m Modale

b Bloc par bloc (non modale)

[3] Possibilité de programmation dans le programme pièce :

\+ Programmable

\- Non programmable

M Programmable uniquement par le constructeur de la machine

[4] Possibilité de programmation dans des actions synchrones :

+	Programmable
-	Non programmable
T	Programmable uniquement dans des cycles technologiques

[5] Référence au document contenant la description complète de l'instruction :

FM-A	Description fonctionnelle Axes et broches
FM-B	Description fonctionnelle Fonctions de base
FM-SA	Description fonctionnelle Actions synchrones
FM-SI	Description fonctionnelle Safety Integrated
FM-TE	Description fonctionnelle Technologies
FM-TM	Description fonctionnelle Gestion d'outils
FM-TR	Description fonctionnelle Transformations
PM-MC	Manuel de programmation Cycles de mesure
PM-NC	Manuel de programmation CN

[6] Réglage par défaut en début du programme (dans la version de base de la commande, si elle n'a pas été programmée autrement).

4.2 Adresses

4.2.1 Lettres adresses

Lettre	Signification	Extension numérique
A	Descripteur d'adresse réglable	x
B	Descripteur d'adresse réglable	x
C	Descripteur d'adresse réglable	x
D	Activation/désactivation de la correction de longueur d'outil, de l'arête tranchante de l'outil	
E	Descripteur d'adresse réglable	x
F	Avance Temporisation en secondes	x
G	Instruction G	
H	Fonction H	x
I	Descripteur d'adresse réglable	x
J	Descripteur d'adresse réglable	x
K	Descripteur d'adresse réglable	x
L	Sous-programmes, appel de...	
M	Fonction M	x
N	Numéro de bloc secondaire	
O	inutilisé	
P	Nombre d'exécutions du programme	
Q	Descripteur d'adresse réglable	x
R	Descripteur de variable (paramètres R) Descripteur de variable réglable (sans extension numérique)	x
S	Valeur de broche Temporisation en tours de broche	x x
T	numéro d'outil	x
U	Descripteur d'adresse réglable	x
V	Descripteur d'adresse réglable	x
W	Descripteur d'adresse réglable	x
X	Descripteur d'adresse réglable	x
Y	Descripteur d'adresse réglable	x
Z	Descripteur d'adresse réglable	x
%	Caractère de début et de séparation lors du transfert de fichiers	
:	Numéro de bloc principal	
/	Bloc optionnel	

4.2.2 Adresses fixes

Adresses fixes sans extension axiale

Descripteur d'adresse	Type d'adresse	Modale/ bloc par bloc (non modale)	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Type de donnée de la valeur affectée
D	Numéro de correction	m								x	INT sans signe
F	Avance, arrêt temporisé	m, s	x							x	REAL sans signe
FLIM	Vitesse tangentielle maximale	m									REAL sans signe
G	Instruction G	Voir liste des fonctions G									INT sans signe
H	Fonctions auxiliaires	b								x	M : INT sans signe H : REAL
L	Numéro de sous-programme	b									INT sans signe
M	Fonctions auxiliaires	b								x	M : INT sans signe H : REAL
N	Numéro de bloc	b									INT sans signe
OVR	correction	m									REAL sans signe
OVRRAP	Correction pour vitesse rapide	m									REAL sans signe
P	Nombre d'exécutions du sous-programme	b									INT sans signe

Descripteur d'adresse	Type d'adresse	Modale/ bloc par bloc (non modale)	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Type de donnée de la valeur affectée
PACCLIM	Accélération maximale sur trajectoire	m									REAL sans signe
S	Broche, arrêt temporisé	m, s								x	REAL sans signe
SCC	Affectation d'un axe transversal à G96 /G961/G962	m									REAL
SPOS	Position de broche	m				x	x	x			REAL
SPOSA	Position de broche au-delà des limites de bloc	m				x	x	x			REAL
T	Numéro d'outil	m								x	INT sans signe

Adresses fixes avec extension axiale

Descripteur d'adresse	Type d'adresse	Modale/ bloc par bloc (non modale)	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Type de donnée de la valeur affectée
ACC	Accélération axiale	m									REAL sans signe
ACCLIMA	Limitation axiale de l'accélération de l'axe asservi	m									REAL sans signe
AX	Descripteur d'axe variable	[1)]	x	x	x	x	x	x			REAL
FA	Avance axiale	m	x							x	REAL sans signe

Descripteur d'adresse	Type d'adresse	Modale/ bloc par bloc (non modale)	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Type de donnée de la valeur affectée
FDA	Avance axiale pour correction par manivelle	b	x								REAL sans signe
FGREF	Rayon de référence	m	x	x							REAL sans signe
FL	Limite avance axiale	m	x								REAL sans signe
FMA	Avance synchrone axiale	m									REAL sans signe
FOC	Déplacement avec couple limité à effet non modal	b									REAL
FOCOF	Désactivation du déplacement avec couple limité à effet modal	m									REAL
FOCON	Activation du déplacement avec couple limité à effet modal	m									REAL
FXS	Activation de l'accostage de butée	m									INT sans signe
FXST	Limite de couple pour l'accostage de butée	m									REAL
FXSW	Fenêtre de surveillance pour l'accostage de butée	m									REAL
IP	Paramètre d'interpolation variable	b	x	x	x	x	x				REAL

Descripteur d'adresse	Type d'adresse	Modale/ bloc par bloc (non modale)	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Type de donnée de la valeur affectée
JERKLIM	Limitation axiale des à-coups	m									REAL sans signe
JERKLIMA	Limitation axiale des à-coups de l'axe asservi	m									REAL sans signe
MEAC	Mesure cyclique	b									INT Mode et 1 - 4 événements déclencheurs
MEASA	Mesure axiale avec effacement de la distance restant à parcourir	b									INT Mode et 1 - 4 événements déclencheurs
MEAWA	Mesure axiale sans effacement de la distance restant à parcourir	b									INT Mode et 1 - 4 événements déclencheurs
MOV	Démarrage de l'axe de positionnement	m	x	x	x	x	x	x	x		REAL
OS	Activation/ désactivation de l'oscillation	m									INT sans signe
OSB	Point de départ de l'oscillation	m	x	x	x	x	x	x			REAL
OSCILL	Affectation des axes d'oscillation, activation de l'oscillation	m									Axes : 1 à 3 axes de pénétration

Descripteur d'adresse	Type d'adresse	Modale/ bloc par bloc (non modale)	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Type de donnée de la valeur affectée
OSCTRL	Options d'oscillation	m									INT sans signe : options d'activation, INT sans signe : options de désactivation
OSE	Point d'arrêt de l'oscillation	m	x	x	x	x	x	x			REAL
OSNSC	Nombre de passes à lécher, oscillation	m									INT sans signe
OSP1	Point d'inversion gauche (oscillation)	m	x	x	x	x	x	x			REAL
OSP2	Point d'inversion droit (oscillation)	m	x	x	x	x	x	x			REAL
OST1	Temps d'arrêt au point d'inversion gauche (oscillation)	m									REAL
OST2	Temps d'arrêt au point d'inversion droit (oscillation)	m									REAL
OVRA	Correction axiale	m	x								REAL sans signe
PO	Coefficient de polynôme	b	x	x		x	x	x			REAL sans signe
POLF	Position LIFTFAST	m	x	x							REAL sans signe
POS	Axe de positionnement	m	x	x	x	x	x	x	x		REAL
POSA	Axe de positionnement au-delà des limites de bloc	m	x	x	x	x	x	x	x		REAL

Descripteur d'adresse	Type d'adresse	Modale/ bloc par bloc (non modale)	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Type de donnée de la valeur affectée
POSP	Positionnement par trajets partiels (oscillation)	m	x	x	x	x	x	x			REAL : Position finale Real : Longueur de pièce INT : option
STA	Durée des passes à lécher (axiale)	m									REAL sans signe
SRA	Distance de retrait avec entrée externe (axiale)	m									REAL sans signe
VELOLIM	Limitation axiale de la vitesse	m									REAL sans signe
VELOLIMA	Limitation axiale de la vitesse de l'axe asservi	m									REAL sans signe

[1] Points finaux absolus : fonction modale, points finaux relatifs : fonction non modale, sinon modale / non modale en fonction de l'instruction G définissant la syntaxe

4.2.3 Adresses réglables

Descripteur d'adresse (réglage par défaut)	Type d'adresse	Modale/ bloc par bloc (non modale)	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Nombre max.	Type de donnée de la valeur affectée
Valeurs d'axe et points finaux											
X, Y, Z, A, B, C	Axe	[1]	x	x	x	x				8	REAL
AP	Angle polaire	m/b[1]	x	x	x					1	REAL
RP	Rayon polaire	m/b[1]	x	x	x					1	REAL sans signe

Descripteur d'adresse (réglage par défaut)	Type d'adresse	Modale/ bloc par bloc (non modale)	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Nombre max.	Type de donnée de la valeur affectée
Orientation de l'outil											
A2, B2, C2	Angles d'Euler ou RPY	b								3	REAL
A3, B3, C3	Composantes du vecteur de direction	b								3	REAL
A4, B4, C4	Composantes du vecteur normal à la surface en début de bloc	b								3	REAL
A5, B5, C5	Composantes du vecteur normal à la surface en fin de bloc	b								3	REAL
A6, B6, C6	Composantes du vecteur de direction pour l'axe rotatif du cône	b								3	REAL
A7, B7, C7	Composantes du vecteur pour l'orientation intermédiaire sur l'enveloppe du cône	b								3	REAL
LEAD	Angle d'avance	m								1	REAL
THETA	Angle de rotation Rotation autour de l'orientation de l'outil	m		x	x					1	REAL
TILT	Angle latéral	m								1	REAL
ORIS	Modification de l'orientation (rapportée à la trajectoire)	m								1	REAL
Paramètres d'interpolation											
I, J, K	Paramètres d'interpolation Coordonnées du point intermédiaire	b		x[2)]	x[2)]					3	REAL
I1, J1, K1		b	x	x	x					3	REAL

Descripteur d'adresse (réglage par défaut)	Type d'adresse	Modale/ bloc par bloc (non modale)	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Nombre max.	Type de donnée de la valeur affectée
RPL	Rotation dans le plan	b								1	REAL
CR	Rayon du cercle	b								1	REAL sans signe
AR	Angle au centre	b								1	REAL sans signe
TURN	Nombre de tours pour une hélice	b								1	INT sans signe
PL	Longueur d'intervalle de paramètres	b								1	REAL sans signe
PW	Poids de point	b								1	REAL sans signe
SD	Degré de spline	m								1	INT sans signe
TU	Angle d'axe	b								1	INT sans signe
STAT	Position des articulations	m								1	INT sans signe
SF	Décalage du point de départ pour filetage	m								1	REAL
DISCL	Distance de sécurité ARD	b								1	REAL sans signe
DISR	Distance de repositionnement / distance ARD	b								1	REAL sans signe
DISPR	Différence sur le contour pour repositionnement	b								1	REAL sans signe
ALF	Angle de relèvement rapide	m								1	INT sans signe
DILF	Longueur de relèvement rapide	m								1	REAL
FP	Point fixe : Numéro du point fixe à accoster	b								1	INT sans signe
RNDM	Arrondissement modal	m								1	REAL sans signe
RND	Arrondissement non modal	b								1	REAL sans signe
CHF	Chanfrein non modal	b								1	REAL sans signe

Descripteur d'adresse (réglage par défaut)	Type d'adresse	Modale/ bloc par bloc (non modale)	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Nombre max.	Type de donnée de la valeur affectée
CHR	Chanfrein dans le sens de déplacement initial	b								1	REAL sans signe
ANG	Angle d'élément de contour	b								1	REAL
ISD	Profondeur de pénétration	m								1	REAL
DISC	Surhaussement de l'arc de raccordement Correction du rayon d'outil	m								1	REAL sans signe
OFFN	Offset sur contour normal	m								1	REAL
DITS	Trajectoire d'approche en filetage	m								1	REAL
DITE	Trajectoire de dégagement en filetage	m								1	REAL
Critères de transition entre blocs											
ADIS	Distance de transition	m								1	REAL sans signe
ADISPOS	Distance de transition pour vitesse rapide	m								1	REAL sans signe
Mesure											
MEAS	Mesure avec palpeur à déclenchement	b								1	INT sans signe
MEAW	Mesure avec palpeur à déclenchement sans suppression de la distance restant à parcourir	b								1	INT sans signe
Comportement des axes et des broches											
LIMS	Limitation de la vitesse de rotation de la broche	m								1	REAL sans signe

Descripteur d'adresse (réglage par défaut)	Type d'adresse	Modale/ bloc par bloc (non modale)	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Nombre max.	Type de donnée de la valeur affectée
COARSEA	Mode de changement de bloc : arrêt précis grossier axial	m									
FINEA	Mode de changement de bloc : arrêt précis fin axial	m									
IPOENDA	Mode de changement de bloc : arrêt interpolateur axial	m									
DIACYCOFA	Axe transversal : désactivation de la programmation axiale du diamètre dans les cycles	m									
DIAM90A	Axe transversal : programmation axiale du diamètre pour G90	m									
DIAMCHAN	Axe transversal : prise en compte de tous les plans transversaux dans l'état de canal de la programmation de diamètre	m									
DIAMCHANA	Axe transversal : prise en compte de l'état de canal de la programmation de diamètre	m									
DIAMOFA	Axe transversal : désactivation de la programmation de diamètre axiale	m									
DIAMONA	Axe transversal : activation de la programmation de diamètre axiale	m									

Descripteur d'adresse (réglage par défaut)	Type d'adresse	Modale/ bloc par bloc (non modale)	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Nombre max.	Type de donnée de la valeur affectée
GP	Position : programmation indirecte d'attributs de positions	m									
Avances											
FAD	Vitesse du mouvement de pénétration lente	b						x		1	REAL sans signe
FD	Avance tangentielle pour correction par manivelle	b								1	REAL sans signe
FRC	Avance pour rayon et chanfrein	b								1	REAL sans signe
FRCM	Avance modale pour rayon et chanfrein	m								1	REAL sans signe
FB	Avance non modale	b								1	REAL sans signe
Grignotage/poinçonnage											
SPN	Nombre de distances partielles par bloc	b								1	INT
SPP	Longueur d'une distance partielle	m								1	REAL
Rectification											
ST	Durée des passes à lécher	b								1	REAL sans signe
SR	Distance de retrait	b								1	REAL sans signe
Sélection de l'outil											
TCARR	Organe porte-outil	m								1	INT
Gestion des outils											
DL	Correction totale d'outil	m								1	INT
Adresses OEM											

Descripteur d'adresse (réglage par défaut)	Type d'adresse	Modale/ bloc par bloc (non modale)	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Nombre max.	Type de donnée de la valeur affectée
OMA1	Adresse OEM 1	m		x	x	x				1	REAL
OMA2	Adresse OEM 2	m		x	x	x				1	REAL
OMA3	Adresse OEM 3	m		x	x	x				1	REAL
OMA4	Adresse OEM 4	m		x	x	x				1	REAL
OMA5	Adresse OEM 5	m		x	x	x				1	REAL
Divers											
CUTMOD	Modification des données de correction pour outils orientables (en liaison avec des organes porte-outil orientables)	m									INT
CUTMODK	Modification des données de correction pour outils orientables (en liaison avec des transformations d'orientation définies à l'aide de chaînes cinématiques)	m									STRING
TOFF	Offset de longueur d'outil parallèlement à l'axe géométrique indiqué	m									REAL
TOFFL	Offset de longueur d'outil dans le sens de la composante de longueur d'outil L1, L2 ou L3	m									REAL
TOFFR	Offset de rayon d'outil	m									REAL
TOFFLR	Offset de rayon d'outil et offset de longueur d'outil simultanés	m									REAL

1) Points finaux absolus : fonction modale, points finaux relatifs : fonction non modale, sinon modale / non modale en fonction de l'instruction G définissant la syntaxe

2) Lorsqu'ils sont utilisés en tant que centres de cercle les paramètres IPO agissent de façon relative. Ils peuvent être programmés de manière absolue avec AC. Lorsque la signification est différente (par ex. pas du filet), la modification d'adresse est ignorée.

4.3 Instructions G

4.3.1 Instructions G

Les instructions G sont réparties en groupes G. Dans les programmes pièce ou les actions synchrones, on ne peut écrire qu'une instruction G d'un groupe G dans un bloc. Une instruction G peut être modale ou non modale.

Modale : jusqu'à la programmation d'une autre instruction G du même groupe G.

4.3.2 Groupe G 1 : instructions de déplacement modales

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G0	1	Déplacement à vitesse rapide	+	m		
G1	2	Interpolation linéaire	+	m	x	
G2	3	Interpolation circulaire sens horaire	+	m		
G3	4	Interpolation circulaire sens antihoraire	+	m		
CIP	5	Interpolation circulaire avec point intermédiaire	+	m		
ASPLINE	6	Akima Spline	+	m		
BSPLINE	7	Spline B	+	m		
CSPLINE	8	Spline cubique	+	m		
POLY	9	Interpolation polynomiale	+	m		
G33	10	Filetage à pas constant	+	m		
G331	11	Taraudage	+	m		
G332	12	Retrait (taraudage)	+	m		
OEMIPO1	13	Réservé	+	m		
OEMIPO2	14	Réservé	+	m		
CT	15	Cercle avec transition tangentielle	+	m		
G34	16	Filetage avec pas en progression linéaire	+	m		
G35	17	Filetage avec pas en régression linéaire	+	m		
INVCW	18	Interpolation de développante dans le sens horaire	+	m		
INVCCW	19	Interpolation de développante dans le sens antihoraire	+	m		
G335	20	Tournage d'un filetage bombé dans le sens horaire	+	m		
G336	21	Tournage d'un filetage bombé dans le sens antihoraire	+	m		

4.3.3 Groupe G 2 : instructions de déplacement non modales, arrêt temporisé

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G4	1	Arrêt temporisé, prédéterminé dans le temps	-	b		
G63	2	Taraudage sans synchronisation	-	b		
G74	3	Accostage du point de référence avec synchronisation	-	b		
G75	4	Accostage d'un point fixe	-	b		
REPOSL	5	Réaccostage linéaire du contour	-	b		
REPOSQ	6	Réaccostage du contour en quart de cercle	-	b		
REPOSH	7	Réaccostage du contour en demi-cercle	-	b		
REPOSA	8	Réaccostage linéaire du contour avec tous les axes	-	b		
REPOSQA	9	Réaccostage du contour avec tous les axes ; axes géométriques en quart de cercle	-	b		
REPOSHA	10	Réaccostage du contour avec tous les axes ; axes géométriques en demi-cercle	-	b		
G147	11	Accostage du contour avec une droite	-	b		
G247	12	Accostage du contour avec un quart de cercle	-	b		
G347	13	Accostage du contour avec un demi-cercle	-	b		
G148	14	Retrait du contour avec une droite	-	b		
G248	15	Retrait du contour avec un quart de cercle	-	b		
G348	16	Retrait du contour avec un demi-cercle	-	b		
G5	17	Rectification en plongée oblique	-	b		
G7	18	Mouvement compensatoire pour la rectification en plongée oblique	-	b		

4.3.4 Groupe G 3 : frame programmable, limitation de la zone de travail et programmation de pôle

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
TRANS	1	TRANSLATION : décalage programmable	-	b		
ROT	2	ROTATION : rotation programmable	-	b		
SCALE	3	SCALE : facteur d'échelle programmable	-	b		
MIRROR	4	MIRROR : fonction miroir programmable	-	b		
ATRANS	5	Additive TRANSLATION : décalage additif programmable	-	b		
AROT	6	Additive ROTATION : rotation programmable	-	b		
ASCALE	7	Additive SCALE : facteur d'échelle programmable	-	b		
AMIRROR	8	Additive MIRROR : fonction miroir programmable	-	b		
-	9	inutilisé	-	-		
G25	10	Limite minimale de la zone de travail / vitesse de rotation de broche minimale	-	b		

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G26	11	Limite maximale de la zone de travail / vitesse de rotation de broche maximale	-	b		
G110	12	Programmation du pôle par rapport à la dernière position de consigne programmée	-	b		
G111	13	Programmation du pôle par rapport à l'origine pièce courante	-	b		
G112	14	Programmation du pôle par rapport au dernier pôle valable	-	b		
G58	15	Décalage d'origine absolu programmable	-	b		
G59	16	Décalage d'origine additif programmable	-	b		
ROTS	17	Rotation avec un angle solide	-	b		
AROTS	18	Rotation additive avec un angle solide	-	b		

4.3.5 Groupe G 4 : FIFO

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
STARTFIFO	1	Départ FIFO Exécution avec remplissage en parallèle du tampon d'exécution	+	m	x	
STOPFIFO	2	Arrêt FIFO Arrêt de l'exécution ; remplissage du tampon d'exécution jusqu'à la reconnaissance de STARTFIFO, la saturation du tampon d'exécution ou la fin du programme	+	m		
FIFOCTRL	3	Activation de la commande automatique du tampon d'exécution	+	m		

4.3.6 Groupe G 6 : Sélection du plan

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G17	1	Sélection du plan du 1er - 2ème axe géométrique	+	m	x	
G18	2	Sélection du plan du 3ème - 1er axe géométrique	+	m		
G19	3	Sélection du plan du 2ème - 3ème axe géométrique	+	m		

4.3.7 Groupe G 7 : Correction du rayon d'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G40	1	Pas de correction du rayon d'outil	+	m	x	
G41	2	Correction du rayon d'outil à gauche du contour	-	m		
G42	3	Correction du rayon d'outil à droite du contour	-	m		

4.3.8 Groupe G 8 : décalage d'origine réglable

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G500	1	Désactivation du décalage d'origine réglable (G54 ... G57, G505 ... G599)	+	m	x	
G54	2	1er décalage d'origine réglable	+	m		
G55	3	2ème décalage d'origine réglable	+	m		
G56	4	3ème décalage d'origine réglable	+	m		
G57	5	4ème décalage d'origine réglable	+	m		
G505	6	5ème décalage d'origine réglable	+	m		
...	...	...	+	m		
G599	100	99ème décalage d'origine réglable	+	m		
Les instructions G de ce groupe G vous permettent d'activer respectivement un frame utilisateur $P_UIFR[ ] réglable. G54 correspond au frame $P_UIFR[1], G505 au frame $P_UIFR[5]. Le nombre de frames utilisateur réglables et, par conséquent, le nombre d'instructions G de ce groupe G peut être réglé via PM28080 $MC_MM_NUM_USER_FRAMES.						

4.3.9 Groupe G 9 : Inhibition des corrections de frame et d'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G53	1	Inhibition des frames actuels : frame programmable y compris frame système pour TOROT et TOFRAME et frame actif réglable (G54 ... G57, G505 ... G599)	-	b		
SUPA	2	Idem G153, y compris l'inhibition des frames système pour le préréglage des valeurs réelles, l'effleurement, le décalage d'origine externe, PAROT incluant les décalages par manivelle (DRF), [décalage d'origine externe], déplacement forcé	-	b		
G153	3	Idem G53, y compris l'inhibition de tous les frames de base spécifiques au canal et/ou à définition globale pour NCU	-	b		
SUPD	4	Inhibition des corrections d'outil actives	-	b		

4.3.10 Groupe G 10 : arrêt précis - contournage

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G60	1	Arrêt précis	+	m	x	
G64	2	Contournage	+	m		
G641	3	Contournage avec arrondissement selon le critère de trajet (= distance de transition programmable)	+	m		
G642	4	Contournage avec arrondissement avec prise en compte des tolérances définies	+	m		
G643	5	Contournage avec arrondissement avec prise en compte des tolérances définies (internes au bloc)	+	m		
G644	6	Contournage avec arrondissement utilisant la dynamique maximale possible	+	m		
G645	7	Contournage avec arrondissement d'angles et changements de blocs tangentiels avec prise en compte de tolérances définies	+	m		

4.3.11 Groupe G 11 : Arrêt précis à effet non modal

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G9	1	Arrêt précis	-	b		

4.3.12 Groupe G 12 : critères de changement de bloc pour arrêt précis (G60/G9)

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G601	1	Changement de bloc lors arrêt précis fin	+	m	x	
G602	2	Changement de bloc lors arrêt précis grossier	+	m		
G603	3	Changement de bloc en fin de bloc IPO	+	m		

4.3.13 Groupe G 13 : cotes de la pièce en système anglo-saxon / métrique

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G70	1	Cotes en inch (longueurs)	+	m		
G71	2	Système de coordonnées métrique mm (longueurs)	+	m	x	

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G700	3	Système de coordonnées anglo-saxon inch, inch/min (longueurs + vitesse + variable système)	+	m		
G710	4	Système de coordonnées métrique mm, mm/min (longueurs + vitesse + variable système)	+	m		

4.3.14 Groupe G 14 : cotes de la pièce en absolu/relatif

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G90	1	Cotation absolue	+	m	x	
G91	2	Indication de cotes relatives	+	m		

4.3.15 Groupe G 15 : type d'avance

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G93	1	Avance en sens inverse du temps tr/min	+	m		
G94	2	Avance linéaire en mm/min, inch/min	+	m	x	
G95	3	Avance par tour en mm/tr, inch/tr	+	m		
G96	4	Avance par tour (comme G95) et vitesse de coupe constante	+	m		
G97	5	Avance par tour et vitesse de rotation de broche constante (vitesse de coupe constante désactivée)	+	m		
G931	6	Indication de l'avance par durée du déplacement, désactiver la vitesse tangentielle constante	+	m		
G961	7	Avance linéaire (comme G94) et vitesse de coupe constante	+	m		
G971	8	Avance linéaire et vitesse de rotation de broche constante (vitesse de coupe constante désactivée)	+	m		
G942	9	Avance linéaire et vitesse de coupe constante ou vitesse de rotation de broche constante	+	m		
G952	10	Avance par tour et vitesse de coupe constante ou vitesse de rotation de broche constante	+	m		
G962	11	Avance linéaire ou avance par tour et vitesse de coupe constante	+	m		
G972	12	Avance linéaire ou avance par tour et vitesse de rotation de broche constante (vitesse de coupe constante désactivée)	+	m		
G973	13	Avance par tour sans limitation de vitesse de la broche et vitesse de rotation de broche constante (G97 sans LIMS pour mode ISO)	+	m		

4.3.16 Groupe G 16 : correction de l'avance au niveau des courbures concave et convexe

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
CFC	1	Avance constante au niveau du contour actif pour courbure concave et convexe	+	m	x	
CFTCP	2	Avance constante au point de référence de l'arête tranchante de l'outil (trajectoire du centre)	+	m		
CFIN	3	Avance constante pour courbure concave, accélération pour courbure convexe	+	m		

4.3.17 Groupe G 17 : correction d'outil pour comportement d'accostage / de retrait

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
NORM	1	Positionnement à la normale au point de départ/point final	+	m	x	
KONT	2	Contournement du point de départ/point final	+	m		
KONTT	3	Accostage/retrait tangentiel	+	m		
KONTC	4	Accostage/retrait à courbure continue	+	m		

4.3.18 Groupe G 18 : correction d'outil pour comportement aux angles

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G450	1	Arc de raccordement (l'outil contourne les angles de la pièce sur une trajectoire circulaire)	+	m	x	
G451	2	Point d'intersection des équidistantes (l'outil s'arrête aux angles de la pièce)	+	m		

4.3.19 Groupe G 19 : raccordement en début de courbe de type spline

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
BNAT	1	Raccordement naturel avec le premier bloc spline	+	m	x	
BTAN	2	Raccordement tangentiel avec le premier bloc spline	+	m		
BAUTO	3	Détermination de la première section spline avec les 3 points suivants	+	m		

4.3.20 Groupe G 20 : raccordement en fin de courbe de type spline

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
ENAT	1	Raccordement naturel avec le bloc de déplacement suivant	+	m	x	
ETAN	2	Raccordement tangentiel avec le bloc de déplacement suivant	+	m		
EAUTO	3	Détermination de la dernière section spline avec les 3 derniers points	+	m		

4.3.21 Groupe G 21 : profil d'accélération

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
BRISK	1	Accélération résultante sous forme d'échelon	+	m	x	
SOFT	2	Accélération résultante limitée	+	m		
DRIVE	3	Accélération tangentielle fonction de la vitesse	+	m		

4.3.22 Groupe G 22 : type de correction d'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
CUT2D	1	CRO 2½-D	+	m	x	
CUT2DF	2	CRO 2½D, relative au frame actuel (plan incliné)	+	m		
CUT3DC	3	CRO 3D pour le fraisage périphérique	+	m		
CUT3DF	4	CRO 3D pour le fraisage en bout avec modification de l'orientation.	+	m		
CUT3DFS	5	CRO 3D pour le fraisage en bout avec orientation constante. L'orientation de l'outil est définie par G17 - G19 et n'est pas influencée par les frames.	+	m		
CUT3DFF	6	CRO 3D pour le fraisage en bout avec orientation constante. L'orientation de l'outil est définie par G17 - G19 et, le cas échéant, par la composante rotationnelle d'un frame.	+	m		
CUT3DCC	7	CRO 3D pour le fraisage périphérique en tenant compte d'une surface de délimitation avec correction de rayon 3D : contour de la surface d'usinage	+	m		
CUT3DCCD	8	CRO 3D pour le fraisage périphérique en tenant compte d'une surface de délimitation avec outil différentiel sur la trajectoire du centre de l'outil : déplacement perpendiculaire jusqu'à la surface de délimitation	+	m		
CUT2DD	9	CRO 2½D rapportée à un outil différentiel	+	m		

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
CUT2DFD	10	CRO 2½D rapportée à un outil différentiel, relative au frame actuel (plan incliné)	+	m		
CUT3DCD	11	CRO 3D rapportée à un outil différentiel pour le fraisage périphérique	+	m		
CUT3DFD	12	CRO 3D rapportée à un outil différentiel pour le fraisage en bout avec modification d'orientation	+	m		

4.3.23 Groupe G 23 : surveillance anticollision sur contours intérieurs

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
CDOF	1	Désactivation de la surveillance anticollision	+	m	x	
CDON	2	Activation de la surveillance anticollision	+	m		
CDOF2	3	Désactivation de la détection des violations du contour en fraisage périphérique 3D	+	m		

4.3.24 Groupe G 24 : commande anticipatrice

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
FFWOF	1	Désactivation de la commande anticipatrice	+	m	x	
FFWON	2	Activation de la commande anticipatrice	+	m		

4.3.25 Groupe G 25 : référence pour l'orientation de l'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
ORIWKS	1	Orientation de l'outil dans le système de coordonnées pièce (SCP)	+	m	x	
ORIMKS	2	Orientation de l'outil dans le système de coordonnées machine (SCM)	+	m		

4.3.26 Groupe G 26 : mode de réaccostage pour REPOS (modal)

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
RMB	1	Réaccostage du point de début de bloc	-	m		
RMI	2	Réaccostage du point d'interruption	-	m	x	
RME	3	Réaccostage du point final du bloc	-	m		
RMN	4	Réaccostage du point de contour le plus proche	-	m		

4.3.27 Groupe G 27 : correction d'outil pour changement d'orientation au niveau des angles saillants

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
ORIC	1	les changements d'orientation aux angles saillants sont superposés au bloc circulaire à insérer	+	m	x	
ORID	2	les changements d'orientation se font avant l'exécution du bloc circulaire inséré	+	m		

4.3.28 Groupe G 28 : limitation de la zone de travail

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
WALIMON	1	Activation de la limitation de la zone de travail	+	m	x	
WALIMOF	2	Désactivation de la limitation de la zone de travail	+	m		

4.3.29 Groupe G 29 : programmation du rayon/diamètre

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
DIAMOF	1	Désactivation de la programmation modale active du diamètre spécifique au canal Cette désactivation active la programmation du rayon spécifique au canal.	+	m	x	
DIAMON	2	Activation de la programmation modale active indépendante du diamètre spécifique au canal L'effet est indépendant du mode programmé pour les cotes (G90/G91).	+	m		

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
DIAM90	3	Activation de la programmation modale active dépendante du diamètre spécifique au canal L'effet est dépendant du mode programmé pour les cotes (G90/G91).	+	m		
DIAMCYCOF	4	Désactivation de la programmation modale active du diamètre spécifique au canal pendant l'exécution du cycle	+	m		

4.3.30 Groupe G 30 : compression du bloc CN

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
COMPOF	1	Compression du bloc CN désactivée	+	m	x	
COMPON	2	Fonction de compresseur COMPON activée	+	m		
COMPCURV	3	Fonction de compresseur COMPCURV activée	+	m		
COMPCAD	4	Fonction de compresseur COMPCAD activée	+	m		
COMPSURF	5	Fonction de compresseur COMPSURF activée	+	m		

4.3.31 Groupe G 31 : Instructions G OEM

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G810	1	Instruction G OEM	-	m		
G811	2	Instruction G OEM	-	m		
G812	3	Instruction G OEM	-	m		
G813	4	Instruction G OEM	-	m		
G814	5	Instruction G OEM	-	m		
G815	6	Instruction G OEM	-	m		
G816	7	Instruction G OEM	-	m		
G817	8	Instruction G OEM	-	m		
G818	9	Instruction G OEM	-	m		
G819	10	Instruction G OEM	-	m		
Deux groupes G sont réservés à l'utilisateur OEM. Ce faisant, il autorise la programmation des fonctions spécifiques élaborées par ses soins par des services de programmation extérieurs.						

4.3.32 Groupe G 32 : Instructions G OEM

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G820	1	Instruction G OEM	-	m		
G821	2	Instruction G OEM	-	m		
G822	3	Instruction G OEM	-	m		
G823	4	Instruction G OEM	-	m		
G824	5	Instruction G OEM	-	m		
G825	6	Instruction G OEM	-	m		
G826	7	Instruction G OEM	-	m		
G827	8	Instruction G OEM	-	m		
G828	9	Instruction G OEM	-	m		
G829	10	Instruction G OEM	-	m		
Deux groupes G sont réservés à l'utilisateur OEM. Ce faisant, il autorise la programmation des fonctions spécifiques élaborées par ses soins par des services de programmation extérieurs.						

4.3.33 Groupe G 33 : correction fine d'outil réglable

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
FTOCOF	1	Désactivation de la correction fine d'outil active en ligne	+	m	x	
FTOCON	2	Activation de la correction fine d'outil active en ligne	-	m		

4.3.34 Groupe G 34 : lissage pour orientation de l'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
OSOF	1	Désactivation du lissage constant de l'orientation de l'outil	+	m	x	
OSC	2	Lissage constant de l'orientation de l'outil	+	m		
OSS	3	Lissage de l'orientation de l'outil en fin de bloc	+	m		
OSSE	4	Lissage de l'orientation de l'outil en début et fin de bloc	+	m		
OSD	5	Arrondissement interne au bloc avec spécification de la longueur de la trajectoire	+	m		
OST	6	Arrondissement interne au bloc avec spécification de la tolérance angulaire	+	m		

4.3.35 Groupe G 35 : poinçonnage et grignotage

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
SPOF	1	Course désactivée, poinçonnage et grignotage désactivés	+	m	x	
SON	2	Grignotage ACTIVE	+	m		
PON	3	Poinçonnage ACTIVE	+	m		
SONS	4	Grignotage activé à la période d'appel de l'interpolateur	-	m		
PONS	5	Poinçonnage activé à la période d'appel de l'interpolateur	-	m		

4.3.36 Groupe G 36 : poinçonnage avec temporisation

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
PDELAYON	1	Poinçonnage avec temporisation activé	+	m	x	
PDELAYOF	2	Poinçonnage avec temporisation désactivé	+	m		

4.3.37 Groupe G 37 : profil d'avance

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
FNORM	1	avance normale selon DIN66025	+	m	x	
FLIN	2	avance variable linéairement	+	m		
FCUB	3	avance variable selon une courbe de type spline cubique	+	m		

4.3.38 Groupe G 38 : affectation des entrées/sorties rapides pour poinçonnage/grignotage

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
SPIF1	1	E/S CN rapides pour poinçonnage/grignotage, octet 1	+	m	x	
SPIF2	2	E/S CN rapides pour poinçonnage/grignotage, octet 2	+	m		

4.3.39 Groupe G 39 : précision de contour programmable

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
CPRECOF	1	Désactivation de la précision de contour programmable	+	m	x	
CPRECON	2	Précision de contour programmable activée	+	m		

4.3.40 Groupe G 40 : correction constante du rayon d'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
CUTCONOF	1	Correction constante du rayon d'outil désactivée	+	m	x	
CUTCONON	2	Correction constante du rayon d'outil activée	+	m		

4.3.41 Groupe G 41 : possibilité d'interruption du filetage

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
LFOF	1	Possibilité d'interruption du filetage désactivée	+	m	x	
LFON	2	Possibilité d'interruption du filetage activée	+	m		

4.3.42 Groupe G 42 : organe porte-outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
TCOABS	1	Détermination des composantes de longueur d'outil à partir de l'orientation actuelle de l'outil	+	m	x	
TCOFR	2	Détermination des composantes de longueur d'outil à partir de l'orientation du frame actif.	+	m		
TCOFRZ	3	Lors de la sélection d'outil, détermination de l'orientation de l'outil d'un frame activé, outil pointé en direction Z	+	m		
TCOFRY	4	Lors de la sélection d'outil, détermination de l'orientation de l'outil d'un frame activé, outil pointé en direction Y	+	m		
TCOFRX	5	Lors de la sélection d'outil, détermination de l'orientation de l'outil d'un frame activé, outil pointé en direction X		m		

4.3.43 Groupe G 43 : sens d'accostage ARD

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G140	1	Sens d'accostage ARD fixé par G41/G42	+	m	x	
G141	2	Sens d'accostage ARD à gauche du contour	+	m		
G142	3	Sens d'accostage ARD à droite du contour	+	m		
G143	4	Sens d'accostage ARD en fonction tangente	+	m		

4.3.44 Groupe G 44 : segmentation des déplacements ARD

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G340	1	Accostage dans l'espace, c'est-à-dire déplacement de profondeur et accostage dans le plan dans un bloc	+	m	x	
G341	2	D'abord approche dans l'axe perpendiculaire (Z), puis accostage dans le plan	+	m		

4.3.45 Groupe G 45 : référence de trajectoire des axes FGROUP

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
SPATH	1	La référence trajectoire pour axes FGROUP est la longueur d'arc	+	m	x	
UPATH	2	La référence trajectoire pour axes FGROUP est le paramètre de courbe	+	m		

4.3.46 Groupe G 46 : choix du plan pour le relèvement rapide

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
LFTXT	1	Plan déterminé à partir de la tangente à la trajectoire et de l'orientation actuelle de l'outil	+	m	x	
LFWP	2	Plan déterminé par le plan de travail actuel (G17/G18/G19)	+	m		
LFPOS	3	Relèvement axial sur une position	+	m		

4.3.47 Groupe G 47 : commutation de mode pour code CN externe

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G290	1	Activer mode de langage SINUMERIK	+	m	x	
G291	2	Activer mode de langage ISO	+	m		

4.3.48 Groupe G 48 : comportement d'accostage/retrait pour la correction du rayon d'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
G460	1	Surveillance anticollision activée pour bloc d'accostage et de retrait	+	m	x	
G461	2	Prolonger le bloc marginal par un arc de cercle, si aucun point d'intersection dans le bloc CRO	+	m		
G462	3	Prolonger le bloc marginal par une droite, si aucun point d'intersection dans le bloc CRO	+	m		

4.3.49 Groupe G 49 : déplacement point à point

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
CP	1	Déplacement avec interpolation	+	m	x	
PTP	2	Déplacement point à point (déplacement d'axe synchrone)	+	m		
PTPG0	3	Déplacement point à point uniquement pour G0, sinon déplacement avec interpolation CP	+	m		
PTPWOC	4	Déplacement point à point sans les mouvements compensatoires provoqués par les changements d'orientation	+	m		

4.3.50 Groupe G 50 : programmation de l'orientation

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
ORIEULER	1	Orientation définie par angles d'Euler	+	m	x	
ORIRPY	2	Orientation définie par angles RPY (ordre de rotation XYZ)	+	m		

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
ORIVIRT1	3	Angles d'orientation définis par axes virtuels d'orientation (définition 1)	+	m		
ORIVIRT2	4	Angles d'orientation définis par axes virtuels d'orientation (définition 2)	+	m		
ORIAXPOS	5	Angles d'orientation définis par axes virtuels d'orientation avec positions d'axe rotatif	+	m		
ORIRPY2	6	Orientation définie par angles RPY (ordre de rotation ZYX)	+	m		

4.3.51 Groupe G 51 : programmation de l'orientation du mode d'interpolation

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
ORIVECT	1	Interpolation circulaire de grand rayon (identique avec ORIPLANE)	+	m	x	
ORIAXES	2	Interpolation linéaire des axes machine ou des axes d'orientation	+	m		
ORIPATH	3	Chemin d'orientation de l'outil rapporté à la trajectoire	+	m		
ORIPLANE	4	Interpolation dans le plan (identique avec ORIVECT)	+	m		
ORICONCW	5	Interpolation sur une enveloppe de cône dans le sens horaire	+	m		
ORICONCCW	6	Interpolation sur une enveloppe de cône dans le sens antihoraire	+	m		
ORICONIO	7	Interpolation sur une enveloppe de cône avec indication d'une orientation intermédiaire	+	m		
ORICONTO	8	Interpolation sur une enveloppe de cône avec transition tangentielle	+	m		
ORICURVE	9	Interpolation avec une courbe 3D additionnelle pour l'orientation	+	m		
ORIPATHS	10	Orientation de l'outil rapportée à la trajectoire, une cassure dans le tracé de l'orientation est lissée	+	m		

4.3.52 Groupe G 52 : rotation de frame par rapport à la pièce

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
PAROTOF	1	Rotation de frame par rapport à la pièce désactivée	+	m	x	
PAROT	2	Rotation de frame par rapport à la pièce activée Oriente le système de coordonnées pièce sur la pièce.	+	m		

4.3.53 Groupe G 53 : rotation de frame par rapport à l'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
TOROTOF	1	Rotation de frame par rapport à l'outil désactivée	+	m	x	
TOROT	2	Orienter l'axe Z du SCP parallèlement à l'orientation de l'outil par rotation du frame	+	m		
TOROTZ	3	Identique à TOROT	+	m		
TOROTY	4	Orienter l'axe Y du SCP parallèlement à l'orientation de l'outil par rotation du frame	+	m		
TOROTX	5	Orienter l'axe X du SCP parallèlement à l'orientation de l'outil par rotation du frame	+	m		
TOFRAME	6	Orienter l'axe Z du SCP parallèlement à l'orientation de l'outil par rotation du frame	+	m		
TOFRAMEZ	7	Identique à TOFRAME	+	m		
TOFRAMEY	8	Orienter l'axe Y du SCP parallèlement à l'orientation de l'outil par rotation du frame	+	m		
TOFRAMEX	9	Orienter l'axe X du SCP parallèlement à l'orientation de l'outil par rotation du frame	+	m		

4.3.54 Groupe G 54 : rotation vectorielle pour la programmation polynomiale

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
ORIROTA	1	Rotation vectorielle absolue	+	m	x	
ORIROTR	2	Rotation vectorielle relative	+	m		
ORIROTT	3	Rotation vectorielle tangentielle	+	m		
ORIROTC	4	Vecteur de rotation tangentiel à la tangente à la trajectoire	+	m		

4.3.55 Groupe G 55 : déplacement en rapide avec/sans interpolation linéaire

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
RTLION	1	Déplacement à vitesse rapide avec interpolation linéaire activé	+	m	x	
RTLIOF	2	Déplacement à vitesse rapide avec interpolation linéaire désactivé Déplacement à vitesse rapide réalisé avec interpolation axiale individuelle.	+	m		

4.3.56 Groupe G 56 : prise en compte du calcul de l'usure de l'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
TOWSTD	1	Valeur de position de base pour corrections sur la longueur d'outil	+	m	x	
TOWMCS	2	Valeurs d'usure dans le système de coordonnées machine (SCM)	+	m		
TOWWCS	3	Valeurs d'usure dans le système de coordonnées pièce (SCP)	+	m		
TOWBCS	4	Valeurs d'usure dans le système de coordonnées de base (SCB)	+	m		
TOWTCS	5	Valeurs d'usure dans le système de coordonnées outil (point de référence de l'organe porte-outil T sur le mandrin du porte-outil)	+	m		
TOWKCS	6	Valeurs d'usure dans le système de coordonnées de la tête d'outil avec transformation cinétique (diffère du SCM par la rotation d'outil)	+	m		

4.3.57 Groupe G 57 : décélération aux angles

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
FENDNORM	1	Décélération aux angles OFF	+	m	x	
G62	2	Décélération aux angles intérieurs lorsque la correction du rayon d'outil est activée (G41/G42)	+	m		
G621	3	Décélération à tous les angles	+	m		

4.3.58 Groupe G 59 : mode dynamique pour interpolation de trajectoire

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
DYNNORM	1	Dynamique normale	+	m	x	
DYNPOS	2	Mode de positionnement, taraudage	+	m		
DYNROUGH	3	Ebauche	+	m		
DYNSEMIFIN	4	Préfinition	+	m		
DYNFINISH	5	Finition	+	m		
DYNPREC	6	Superfinition	+	m		

4.3.59 Groupe G 60 : limitation de la zone de travail

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
WALCS0	1	Limitation de la zone de travail SCP désactivée	+	m	x	
WALCS1	2	Groupe de limitations de la zone de travail SCP 1 actif	+	m		
WALCS2	3	Groupe de limitations de la zone de travail SCP 2 actif	+	m		
WALCS3	4	Groupe de limitations de la zone de travail SCP 3 actif	+	m		
WALCS4	5	Groupe de limitations de la zone de travail SCP 4 actif	+	m		
WALCS5	6	Groupe de limitations de la zone de travail SCP 5 actif	+	m		
WALCS6	7	Groupe de limitations de la zone de travail SCP 6 actif	+	m		
WALCS7	8	Groupe de limitations de la zone de travail SCP 7 actif	+	m		
WALCS8	9	Groupe de limitations de la zone de travail SCP 8 actif	+	m		
WALCS9	10	Groupe de limitations de la zone de travail SCP 9 actif	+	m		
WALCS10	11	Groupe de limitations de la zone de travail SCP 10 actif	+	m		

4.3.60 Groupe G 61 : lissage pour orientation de l'outil

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
ORISOF	1	Désactivation du lissage constant de l'orientation de l'outil	+	m	x	
ORISON	2	Activation du lissage de l'orientation de l'outil	+	m		

4.3.61 Groupe G 62 : mode de réaccostage pour REPOS (non modal)

Instruction G	N°[1]	Signification	PM20150[2]	E[3]	STD[4]	
					SAG	CM
RMBBL	1	Réaccostage du point de début de bloc	-	b		
RMIBL	2	Réaccostage du point d'interruption	-	b	x	
RMEBL	3	Réaccostage du point final du bloc	-	b		
RMNBL	4	Réaccostage du point de contour le plus proche	-	b		

4.3.62 Groupe G 64 : frames de rectification

Instruction G	N° [1]	Signification Frame de rectification actif dans le canal $P_GFRAME =	PM20150[2]	E[3]	STD[4]	
					SAG	CM
GFRAME[0]	1	Frame de rectification de la gestion des données $P_GFR[0] (frame neutre)	+	m	x	
GFRAME[1]	2	Frame de rectification de la gestion des données $P_GFR[1]	+	m		
GFRAME[2]	3	Frame de rectification de la gestion des données $P_GFR[2]	+	m		
...	...		+	m		
GFRAME[100]	101	Frame de rectification de la gestion des données $P_GFR[100]	+	m		

Légende

1) Numéro interne (p. ex. pour l'interface avec l'AP)

2) Possibilité de configuration de l'instruction G en tant que position d'effacement du groupe G lors du démarrage, de la réinitialisation (Reset) ou de la fin de programme pièce (avec PM20150 $MC_GCODE_RESET_VALUES) :

+ configurable

- non configurable

3) Prise d'effet de l'instruction G :

m modale (pour l'ensemble des blocs)

b bloc par bloc (non modale)

4) Position d'effacement, voir les paramètres machine suivants :

- PM20149 $MC_GCODE_RESET_S_VALUES (Position d'effacement des groupes G (fixe))
- PM20150 $MC_GCODE_RESET_VALUES (Position d'effacement des groupes G)
- PM20151 $MC_GCODE_RESET_S_MODE (Comportement des groupes G lors du Reset (fixe))
- PM20152 $MC_GCODE_RESET_MODE (Comportement des groupes G lors du Reset)
- PM20154 $MC_EXTERN_GCODE_RESET_VALUES (Position d'effacement des groupes G en mode ISO)
- PM20156 $MC_EXTERN_GCODE_RESET_MODE (Comportement des groupes G externes lors du Reset)

SAG Réglage par défaut **S**iemens **AG**

CM Réglage par défaut **C**onstructeur de **m**achines (voir les indications du constructeur de machines)

4.4 Procédures prédéfinies

L'appel d'une procédure prédéfinie lance l'exécution d'une fonction CN prédéfinie. Contrairement à une fonction prédéfinie, une procédure prédéfinie ne fournit **aucune** valeur en retour.

Système de coordonnées					
Descripteur	**Paramètre**				**Signification**
	1.	**2.**	**3. - 15.**	**4. - 16.**	
PRESETON	AXIS*) : descripteur d'axe axe machine	REAL : décalage Preset contexte G700/G710	comme 1...	comme 2...	Forçage des valeurs réelles pour les axes programmés avec perte de l'état de référencement
PRESETONS	AXIS*) : descripteur d'axe axe machine	REAL : décalage Preset contexte G700/G710	comme 1...	comme 2...	Forçage des valeurs réelles pour les axes programmés sans perte de l'état de référencement
DRFOF					Effacement du décalage DRF pour tous les axes affectés au canal

*) A la place des descripteurs d'axes machine, vous pouvez aussi indiquer, d'une manière générale, des descripteurs d'axes géométriques ou d'axes supplémentaires, dans la mesure où une transformation cinématique univoque est possible.

Groupes d'axes					
Descripteur	**Paramètre**				**Signification**
GEOAX	**1.**	**2.**	**3. / 5.**	**4. / 6.**	Sélection d'un système de coordonnées parallèle
	INT : numéro d'axe géométrique 1 - 3	AXIS : descripteur d'axe de canal	comme 1	comme 2	
FGROUP	**1. – 8.**				Référence variable à une valeur F : détermination des axes auxquels se rapporte l'avance tangentielle Nombre maximal d'axes : 8 L'utilisation de FGROUP ( ) sans indication de paramètres permet d'activer le réglage par défaut pour la référence de valeur F.
	AXIS : descripteur d'axe de canal				
SPLINEPATH	**1.**	**2. - 9.**			Définition du groupe spline Nombre maximal d'axes : 8
	INT : groupe spline (doit être égal à 1)	AXIS : descripteur géométrique ou supplémentaire			

Groupes d'axes				
Descripteur	**Paramètre**			**Signification**
POLYPATH	**1.**	**2.**		Activation de l'interpolation polynomiale pour des groupes d'axes sélectifs
	STRING	STRING		

Déplacements conjugués							
Descripteur	**Paramètre**						**Signification**
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
TANG	AXIS : nom d'axe axe asservi	AXIS : axe pilote 1	AXIS : axe pilote 2	REAL : facteur de couplage	CHAR : Option : "B" : asservissement dans le SCB "W" : asservissement dans le SCP	Optimisation CHAR : "S" : Standard "P" : autom. avec distance d'arrondissement, tolérance d'angle	Positionnement tangentiel : définition du couplage La tangente pour l'asservissement est déterminée à partir des deux axes pilotes indiqués. Le facteur de couplage indique le rapport entre les variations angulaires de la tangente et de l'axe asservi. Il est en général égal à 1.
TANGON	AXIS : nom d'axe axe asservi	REAL : angle d'offset	REAL : distance d'arrondissement	REAL : tolérance d'angle			Positionnement tangentiel : activation du couplage
TANGOF	AXIS : nom d'axe axe asservi						Positionnement tangentiel : désactivation du couplage
TLIFT	AXIS : axe asservi						Positionnement tangentiel : activation de la création de blocs intermédiaires
TRAILON	AXIS : axe asservi	AXIS : axe pilote	REAL : facteur de couplage				Activation des déplacements conjugués synchrones à l'axe
TRAILOF	AXIS : axe asservi	AXIS : axe pilote					Désactivation les déplacements conjugués synchrones à l'axe
TANGDEL	AXIS : axe asservi						Positionnement tangentiel : suppression du couplage

Tables de courbes						
Descripteur	**Paramètre**					**Signification**
	1.	**2.**	**3.**	**4.**	**5.**	
CTABDEF	AXIS : axe asservi	AXIS : axe pilote	INT : numéro de table	INT : comportement aux limites de la plage de déf.	STRING : indication du lieu de stockage	Activation de la définition de table Les blocs de déplacement suivants définissent la table de courbes.
CTABEND	AXIS : axe asservi	AXIS : axe pilote	INT : numéro de table	INT : comportement aux limites de la plage de déf.		Désactivation de la définition de table
CTABDEL	INT : numéro de table n	INT : numéro de table m	STRING : indication du lieu de stockage			Suppression d'une table de courbes
CTABLOCK	INT : numéro de table n					Bloque la table de courbes ayant le numéro n, c.-à-d. que cette table ne peut pas être supprimée/écrasée
CTABUNLOCK	INT : numéro de table n					Débloque la table ayant le numéro n protégée par CTABLOCK
LEADON	AXIS : axe asservi	AXIS : axe pilote	INT : numéro de table			Activation du couplage par valeur pilote
LEADOF	AXIS : axe asservi	AXIS : axe pilote				Désactivation du couplage par valeur pilote

Profil d'accélération axial		
Descripteur	**Paramètre**	**Signification**
	1. – 8.	
BRISKA	AXIS	Activation de l'accélération en échelon pour les axes programmés
SOFTA	AXIS	Activation de l'accélération d'axe avec limitation des à-coups pour les axes programmés
DRIVEA	AXIS	Activation de la caractéristique d'accélération avec coude pour les axes programmés
JERKA	AXIS	Le comportement à l'accélération, réglé à l'aide du paramètre machine $MA_AX_JERK_ENABLE, s'applique aux axes programmés.

Avance par tour			
Descripteur	**Paramètre**		**Signification**
FPRAON	**1.**	**2.**	Activation de l'avance par tour axial
	AXIS : axe pour lequel l'avance par tour est activée	AXIS : axe/broche dont est dérivée l'avance par tour. Si aucun axe n'est programmé, l'avance par tour est dérivée de la broche maître.	
FPRAOF	**1er - n-ième**		Désactivation de l'avance par tour axial L'avance par tour peut être désactivée pour plusieurs axes à la fois. Vous pouvez programmer autant d'axes qu'il est admis au maximum par bloc.
	AXIS : axes pour lesquels l'avance par tour est désactivée		
FPR	**1.**		Sélection d'un axe rotatif / d'une broche dont sera dérivée l'avance par tour de la trajectoire pour G95. Le paramétrage défini avec FPR a un effet modal.
	AXIS : axe/broche dont est dérivée l'avance par tour. Si aucun axe n'est programmé, l'avance par tour est dérivée de la broche maître.		

Transformations				
Descripteur	**Paramètre**			**Signification**
	1.	**2.**	**3.**	
TRACYL	REAL : diamètre de travail	INT : numéro de la transformation		Cylindre : transformation sur surfaces latérales Plusieurs transformations peuvent être réglées dans un même canal. Le numéro de transformation indique quelle est la transformation à activer. Quand le 2e paramètre est omis, ce sont les transformations réglées avec le PM qui sont activées.
TRANSMIT	INT : numéro de la transformation			Transmit : transformation polaire Plusieurs transformations peuvent être réglées dans un même canal. Le numéro de transformation indique quelle est la transformation à activer. Quand le paramètre fait défaut, ce sont les transformations réglées avec le PM qui sont activées.

Transformations				
Descripteur	**Paramètre**			**Signification**
	1.	**2.**	**3.**	
TRAANG	REAL : angle	INT : numéro de la transformation		Transformation axe incliné Plusieurs transformations peuvent être réglées dans un même canal. Le numéro de transformation indique quelle est la transformation à activer. Quand le 2e paramètre est omis, ce sont les transformations réglées avec le PM qui sont activées. Si l'angle n'est pas programmé (TRAANG (,2) ou TRAANG) le dernier angle est modal.
TRAORI	INT : numéro de la transformation			Transformation 4-5 axes Plusieurs transformations peuvent être réglées dans un même canal. Le numéro de transformation indique quelle est la transformation à activer.
TRACON	INT : numéro de la transformation	REAL : autres paramètres selon PM		Transformation en cascade La signification des paramètres dépend du type de cascade.
TRAFOOF				Désactiver la transformation
TRAFOON	STRING : Nom du jeu de paramètres de transformation	REAL : Diamètre de référence ou d'usinage (uniquement TRACYL)	BOOL : avec/sans correction des flancs de rainure (uniquement TRACYL)	Activation de la transformation définie avec des chaînes cinématiques

Broche			
Descripteur	**Paramètre**		**Signification**
	1	**2e - n-ième**	
SPCON	INT : numéro de broche	INT : numéro de broche	Basculement broche en mode asservissement de position
SPCOF	INT : numéro de broche	INT : numéro de broche	Basculement broche en mode régulation de vitesse de rotation
SETMS	INT : numéro de broche		Déclaration de la broche en tant que broche maître pour le canal actuel Lorsque SETMS( ) est utilisé sans indication de paramètres, c'est le préréglage défini par les paramètres machine qui prend effet.

Rectification		
Descripteur	**Paramètre**	**Signification**
	1.	
GWPSON	INT : numéro de broche	Activation de la vitesse circonférentielle de meule constante Si le numéro de broche n'est pas programmé, la vitesse circonférentielle de meule constante est activée pour la broche entraînant l'outil actif.
GWPSOF	INT : numéro de broche	Désactivation de la vitesse circonférentielle de meule constante Si le numéro de broche n'est pas programmé, la vitesse circonférentielle de meule constante est désactivée pour la broche entraînant l'outil actif.
TMON	INT : numéro T	Activation de la surveillance d'outil spécifique à la rectification Si aucun numéro T n'est programmé, la surveillance est activée pour l'outil actif.
TMOF	INT : numéro T	Désactivation de la surveillance d'outil Si aucun numéro T n'est programmé, la surveillance est désactivée pour l'outil actif.

Chariotage					
Descripteur	**Paramètre**				**Signification**
	1.	**2.**	**3.**	**4.**	
CONTPRON	REAL [,11] : table de contour	CHAR : type d'usinage	INT : nombre de détalonnages	INT : état du calcul	Activation de la préparation des références Les programmes de contour ou blocs CN appelés par la suite sont décomposés en déplacements élémentaires, qui sont rangés dans la table de contour. Le nombre de détalonnages est retourné.
CONTDCON	REAL [, 6] : table de contour	INT : sens d'usinage			Décodage du contour Les blocs d'un contour sont codés de façon à économiser de la capacité mémoire (un bloc par ligne de table) puis enregistrés dans la table indiquée.
EXECUTE	INT : état d'erreur				Activation de l'exécution du programme Retour à l'exécution normale du programme à partir du mode de préparation des références ou après la définition d'une zone de protection.

Exécution d'une table		
Descripteur	**Paramètre**	**Signification**
	1.	
EXECTAB	REAL [11] : élément de la table de déplacements	Exécution d'un élément d'une table de déplacements

Zones de protection						
Descripteur	**Paramètre**					**Signification**
	1.	**2.**	**3.**	**4.**	**5.**	
CPROTDEF	INT : numéro de la zone de protection	BOOL : TRUE : zone de protection orientée outil	INT : 0 : les 4e et 5e paramètres ne sont pas exploités 1 : le 4e paramètre est exploité 2 : le 5e paramètre est exploité 3 : les 4e et 5e paramètres sont exploités	REAL : limitation dans le sens plus	REAL : limitation dans le sens moins	Définition d'une zone de protection spécifique à un canal
NPROTDEF	INT : numéro de la zone de protection	BOOL : TRUE : zone de protection orientée outil	INT : 0 : les 4e et 5e paramètres ne sont pas exploités 1 : le 4e paramètre est exploité 2 : le 5e paramètre est exploité 3 : les 4e et 5e paramètres sont exploités	REAL : limitation dans le sens plus	REAL : limitation dans le sens moins	Définition d'une zone de protection spécifique à la machine

Zones de protection						
Descripteur	**Paramètre**					**Signification**
	1.	**2.**	**3.**	**4.**	**5.**	
CPROT	INT : numéro de la zone de protection	INT : option 0 : zone de protection désactivée 1 : préactivation de la zone de protection 2 : zone de protection activée 3 : préactivation de la zone de protection avec arrêt conditionnel, uniquement pour zones de protection actives	REAL : décalage de la zone de protection dans le premier axe géométrique	REAL : décalage de la zone de protection dans le deuxième axe géométrique	REAL : décalage de la zone de protection dans le troisième axe géométrique	Activation/désactivation de la zone de protection spécifique à un canal
NPROT	INT : numéro de la zone de protection	INT : option 0 : zone de protection désactivée 1 : préactivation de la zone de protection 2 : zone de protection activée 3 : préactivation de la zone de protection avec arrêt conditionnel, uniquement pour zones de protection actives	REAL : décalage de la zone de protection dans le premier axe géométrique	REAL : décalage de la zone de protection dans le deuxième axe géométrique	REAL : décalage de la zone de protection dans le troisième axe géométrique	Activation/désactivation de la zone de protection spécifique à une machine

Prétraitement des blocs / bloc par bloc		
Descripteur	**Paramètre**	**Signification**
STOPRE		Arrêt du prétraitement des blocs jusqu'à ce que tous les blocs préparés aient été exécutés
SBLOF		Inhibition du traitement bloc par bloc
SBLON		Annulation de l'inhibition du traitement bloc par bloc

Interruptions		
Descripteur	**Paramètre**	**Signification**
	1.	
DISABLE	INT : numéro de l'entrée d'interruption	La routine d'interruption affectée à l'entrée matérielle indiquée est désactivée. Même une instruction de relèvement rapide n'est pas exécutée. L'affectation entre borne d'entrée et routine d'interruption effectuée à l'aide de SETINT est conservée et peut être réactivée à l'aide de l'instruction ENABLE.
ENABLE	INT : numéro de l'entrée d'interruption	Réactivation de l'affectation de routines d'interruption qui a été désactivée via DISABLE.
CLRINT	INT : numéro de l'entrée d'interruption	Suppression de l'affectation des routines d'interruption et des attributs à une entrée d'interruption. La routine d'interruption est ainsi annulée. Lorsque survient l'interruption, aucune réaction n'a lieu.

Actions synchrones		
Descripteur	**Paramètre**	**Signification**
	1er – n-ième	
CANCEL	INT : numéro de l'action synchrone	Abandon de l'action synchrone modale ayant l'ID spécifiée. Il est également possible de spécifier plusieurs ID séparés par des virgules.
CANCELSUB		Annuler le niveau de sous-programme actuel

Définition de fonction					
Descripteur	**Paramètre**				**Signification**
	1.	**2.**	**3.**	**4.-7.**	
FCTDEF	INT : numéro de fonction	REAL : valeur limite inférieure	REAL : valeur limite supérieure	REAL : coefficients a0-a3	Définition d'une fonction polynôme Celle-ci est utilisée pour les fonctions SYNFCT ou PUTFTOCF.

Communication			
Descripteur	**Paramètre**		**Signification**
	1.	**2.**	
MMC	STRING : ordre	CHAR : mode d'acquittement*) "N" : sans acquittement "S" : acquittement synchrone "A" : acquittement asynchrone	Ordre à l'interpréteur d'ordres IHM pour la configuration de fenêtres par l'intermédiaire du programme CN

*) Les ordres sont acquittés sur demande par les éléments assurant leur exécution (canal, CN...).

Coordination de programmes					
Descripteur	**Paramètre**				**Signification**
INIT	**1.**	**2.**	**3.**		Sélection d'un programme CN pour exécution dans un canal
	INT : numéro de canal **ou** nom du canal du PM20000*)	STRING : indication du chemin	CHAR : mode d'acquittement**)		
	1er - n-ième				
START	INT : numéro de canal **ou** nom du canal du PM20000*)				Lancement des programmes sélectionnés simultanément dans plusieurs canaux, à partir du programme en cours Cet ordre n'a aucun effet pour le canal propre.
WAITE	INT : numéro de canal **ou** nom du canal du PM20000*)				Attente de la fin du programme dans un ou plusieurs autres canaux
	1.	**2e - n-ième**			
WAITM	INT : numéro de repère	INT : numéro de canal **ou** nom du canal du PM20000*)			Attente de l'atteinte d'une marque dans les canaux spécifiés Le bloc précédent est terminé avec un arrêt précis.
WAITMC	INT : numéro de repère	INT : numéro de canal **ou** nom du canal du PM20000*)			Attente de l'atteinte d'une marque dans les canaux spécifiés L'arrêt précis n'est déclenché que si les autres canaux n'ont pas encore atteint le repère.
	1er - n-ième				
SETM	INT : numéro de repère				Positionnement d'un ou plusieurs repères pour la coordination de canaux L'exécution dans le canal propre n'en est pas affectée.
CLEARM	INT : numéro de repère				Suppression d'un ou plusieurs repères pour la coordination de canaux L'exécution dans le canal propre n'en est pas affectée.
	1er - n-ième				
WAITP	AXIS : descripteur d'axe				Attente jusqu'à ce que les axes de positionnement spécifiés, qui ont préalablement été programmés avec POSA, atteignent leur point final programmé

Coordination de programmes					
Descripteur	**Paramètre**				**Signification**
WAITS	INT : numéro de broche				Attente jusqu'à ce que les broches spécifiées, qui ont préalablement été programmées avec SPOSA, atteignent leur point final programmé
RET	**1.**	**2.**	**3.**	**4.**	Fin de sous-programme sans sortie de fonction à l'AP. En spécifiant le 1er paramètre (destination de saut), le retour s'effectue d'abord au bloc suivant le bloc d'appel. Ensuite, en fonction de la programmation (RET ou RETB), la recherche de la destination s'effectue selon la stratégie suivante : • RET : Recherche en direction de la fin du programme. Si la recherche est infructueuse, elle s'effectue alors en direction du début du programme. • RETB : Recherche en direction du début du programme. Si la recherche est infructueuse, elle s'effectue alors en direction de la fin du programme.
	INT (ou STRING) : destination de saut (n° de bloc / repère) pour le retour	INT : 0 : retour à la destination de saut du 1er par. > 0 : retour au bloc suivant	INT : nombre de niveaux de sous-programme à sauter	BOOL : retour au premier bloc du programme principal	
RETB	INT (ou STRING) : destination de saut (n° de bloc / repère) pour le retour	INT : 0 : retour à la destination de saut du 1er par. > 0 : retour au bloc suivant	INT : nombre de niveaux de sous-programme à sauter	BOOL : retour au premier bloc du programme principal	
	1er - n-ième				
GET	AXIS : descripteur d'axe***)				Prise en charge d'axe(s) machine Les axes indiqués doivent être libérés avec RELEASE dans l'autre canal.
GETD	AXIS : descripteur d'axe***)				Prise en charge directe d'axe(s) machine Les axes indiqués ne doivent **pas** être libérés avec RELEASE.
RELEASE	AXIS : descripteur d'axe***)				Libération d'axe(s) machine
	1.	**2.**	**3.**	**4.**	
PUTFTOC	REAL : valeur de correction	INT : numéro de paramètre	INT : numéro de canal **ou** nom du canal du PM20000*)	INT : numéro de broche	Modification de la correction fine d'outil

Coordination de programmes					
Descripteur	**Paramètre**				**Signification**
PUTFTOCF	INT : n° de la fonction	VAR REAL : valeur de référence	INT : numéro de paramètre	INT : numéro de canal **ou** nom du canal du PM20000*)	Modification de la correction fine d'outil selon une fonction définie avec FCTDEF (polynôme de 3e degré au maximum) Le numéro utilisé ici doit être spécifié pour FCTDEF.
AXTOCHAN	**1.**	**2.**	**3e - n-ième**	**4e - m-ième**	Transfert d'axes à d'autres canaux
	AXIS : descripteur d'axe	INT : numéro de canal **ou** nom du canal du PM20000*)	comme 1...	comme 2...	

*) Les noms de canaux définis par le PM20000 $MC_CHAN_NAME peuvent également être programmés à la place des numéros de canaux.

**) Les ordres sont acquittés sur demande par les éléments assurant leur exécution (canal, CN...).

***) Une broche peut également être programmée à la place de l'axe au moyen la fonction SPI : p. ex. GET(SPI(1))

Accès aux données		
Descripteur	**Paramètre**	**Signification**
CHANDATA	**1.** INT : numéro de canal	Réglage du numéro de canal pour les accès aux données de canal (admissible uniquement dans le bloc d'initialisation). Les accès suivants se rapportent au canal réglé avec CHANDATA.
NEWCONF		Application des paramètres machine modifiés

Messages			
Descripteur	**Paramètre**		**Signification**
	1.	**2.**	
MSG	STRING : message	INT : exécution	Génération en sortie de chaînes de caractères au choix en tant que message sur l'interface utilisateur
WRTPR	STRING : chaîne de caractères	INT : exécution	Écrire la chaîne de caractères dans la variable OPI

Accès fichier						
Descripteur	**Paramètre**					**Signification**
READ	**1.**	**2.**	**3.**	**4.**	**5.**	Lecture de blocs à partir du système de fichiers
	VAR INT : erreur	CHAR[160] : nom de fichier	INT : ligne de début de la zone de fichier à lire	INT : nombre de lignes à lire	VAR CHAR[255] : tableau de variable dans lequel l'information lue est consignée	
WRITE	**1.**	**2.**	**3.**	**4.**		Écriture d'un bloc dans le système de fichiers (ou sur un appareil/ fichier externe)
	VAR INT : erreur	CHAR[160] : nom de fichier	STRING : appareil/ fichier pour sortie externe	CHAR[200] : bloc		
DELETE	**1.**	**2.**				Suppression d'un fichier
	VAR INT : erreur	CHAR[160] : nom de fichier				

Alarmes			
Descripteur	**Paramètre**		**Signification**
	1.	**2.**	
SETAL	INT : numéro d'alarme (alarme de cycle)	STRING : chaîne de caractères	Activation d'une alarme Une chaîne de caractères comportant jusqu'à 4 paramètres peut aussi être ajoutée au numéro d'alarme. Les paramètres prédéfinis suivants sont disponibles : %1 = numéro de canal %2 = numéro de bloc, étiquette %3 = indice de texte pour alarmes de cycles %4 = paramètre d'alarme supplémentaire

Gestion des outils				
Descripteur	**Paramètre**			**Signification**
	1.	**2.**		
DELDL	INT : n° T	INT : n° D		Effacer toutes les corrections totales d'un tranchant (ou d'un outil, si D n'est pas indiqué)

Gestion des outils							
Descripteur	**Paramètre**						**Signification**
DELT	STRING[32] : descripteur d'outil	INT : n° d'outil frère					Suppression d'un outil Le numéro d'outil frère peut être omis.
DELTC	INT : n° de bloc de données n	INT : n° de bloc de données m					Suppression des blocs de données d'organe porte-outil n à m
DZERO							Invalidation des numéros D de tous les outils de l'unité TO affectée au canal
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
GETFREELOC	VAR INT : n° de magasin (valeur de retour)	VAR INT : n° d'emplacement (valeur de retour)	INT : n° T	INT : n° du magasin de référence	CHAR : spécification en fonction du 4e paramètre	INT : mode de réservation	Recherche d'un emplacement vide pour un outil
	1.	**2.**					
GETSELT	VAR INT : n° T (valeur de retour)	INT : n° de broche					Fournit le numéro T de l'outil présélectionné pour la broche
GETEXET	VAR INT : n° T (valeur de retour)	INT : n° de broche					Fournit le numéro T de l'outil actif du point de vue du programme CN
GETTENV	STRING : nom de l'environnement d'outil	INT ARRAY[3] : valeurs de retour					Lecture des numéros T, D et DL consignés dans un environnement d'outil
	1.	**2.**	**3.**	**4.**			
POSM	INT : n° de l'emplacement sur lequel le positionnement doit être réalisé	INT : n° du magasin à déplacer	INT : n° d'emplacement du magasin interne	INT : n° du magasin interne			Positionnement d'un magasin
RESETMON	VAR INT : état = résultat de l'opération (valeur de retour)	INT : n° T interne	INT : n° D de l'outil	INT : paramètre facultatif codé sur bits			Régler la valeur réelle de l'outil sur la valeur de consigne

Gestion des outils						
Descripteur	**Paramètre**					**Signification**
SETDNO	**1.**	**2.**	**3.**			Définition du numéro de correction (D) du tranchant de l'outil (T)
	INT : n° T	INT : n° du tranchant	INT : n° D			
SETMTH	**1.**					Déclarer un numéro de porte-outil
	INT : n° de porte-outil					
SETPIECE	**1.**	**2.**				Décrémentation du compteur de pièces de la broche L'utilisateur peut ainsi actualiser les données de surveillance du nombre de pièces pour les outils qui interviennent dans le processus d'usinage.
	INT : valeur de la décrémentation	INT : n° de broche				
	1.	**2.**	**3.**	**4.**		
SETTA	VAR INT : état = résultat de l'opération (valeur de retour)	INT : n° de magasin	INT : n° de groupe d'usure	INT : sous-groupe d'outils		Activation d'un outil du groupe d'usure
SETTIA	VAR INT : état = résultat de l'opération (valeur de retour)	INT : n° de magasin	INT : n° de groupe d'usure	INT : sous-groupe d'outils		Désactivation d'un outil du groupe d'usure
TCA	**1.**	**2.**	**3.**			Sélection/changement d'outil indépendamment de l'état de l'outil
	STRING[32] : descripteur d'outil	INT : n° d'outil frère	INT : n° de porte-outil			
TCI	**1.**	**2.**				Mise en place de l'outil dans le magasin depuis l'emplacement tampon
	INT : n° de l'emplacement tampon	INT : n° de porte-outil				

Gestion des outils							
Descripteur	**Paramètre**						**Signification**
MVTOOL	**1.**	**2.**	**3.**	**4.**	**5.**		Instruction de langage pour le déplacement d'un outil
	INT : état	INT : n° de magasin	INT : n° d'emplacement	INT : n° de magasin après le déplacement	INT : n° d'emplacement de destination après le déplacement		

Orientation de l'outil				
Descripteur	**Paramètre**			**Signification**
	1.	**2.**	**3.**	
ORIRESET	REAL : position de base du 1er axe géo.	REAL : position de base du 2e axe géo.	REAL : position de base du 3e axe géo.	Position de base de l'orientation de l'outil

Broche synchrone							
Descripteur	**Paramètre**						**Signification**
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
COUPDEF	AXIS : broche asservie	AXIS : broche pilote	REAL : numérateur du rapport de transmission	REAL : dénominateur du rapport de transmission	STRING[8] : mode de changement de bloc	STRING[2] : type de couplage	Définition du groupe de broches synchrones
COUPDEL	AXIS : broche asservie	AXIS : broche pilote					Suppression du groupe de broches synchrones
COUPRES	AXIS : broche asservie	AXIS : broche pilote					Remise à zéro des paramètres de couplage sur les valeurs PM et SD configurées

Broche synchrone							
Descripteur	**Paramètre**						**Signification**
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
COUPON	AXIS : broche asservie	AXIS : broche pilote	REAL : position d'activation de la broche asservie				Activation du couplage de broches synchrones Si une position d'activation (décalage d'angle entre BA et BP, qui se rapporte – de manière absolue ou relative – à la position zéro degré de la BP dans le sens de rotation positif) est spécifiée pour la broche asservie, le couplage n'est activé que lorsque la position spécifiée a été dépassée.
COUPONC	AXIS : broche asservie	AXIS : broche pilote					Activation du couplage de broches synchrones Le descripteur COUPONC permet d'appliquer la vitesse actuelle de la broche asservie (M3/M4 S...) au démarrage du couplage.
COUPOF	AXIS : broche asservie	AXIS : broche pilote	REAL : position de désactivation de la broche asservie (absolue)	REAL : position de désactivation de la broche pilote (absolue)			Désactivation du couplage de broches synchrones Si des positions sont spécifiées, le couplage n'est supprimé qu'après que toutes les positions indiquées ont été dépassées. La broche asservie continue à tourner avec la dernière vitesse de rotation atteinte avant le découplage.

Broche synchrone							
Descripteur	**Paramètre**						**Signification**
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
COUPOFS	AXIS : broche asservie	AXIS : broche pilote	REAL : position de désactivation de la broche asservie (absolue)				Désactivation du couplage de broches synchrones avec arrêt de la broche asservie Si une position est spécifiée, le couplage n'est annulé qu'après le dépassement de la position spécifiée.
WAITC	AXIS : broche asservie	STRING [8] : mode de changement de bloc	AXIS : broche asservie	STRING[8] : mode de changement de bloc			Attente jusqu'à ce que le critère de changement de bloc de couplage pour les broches (2 max.) soit rempli Si le mode de changement de bloc n'est pas spécifié, le mode spécifié lors de la définition avec COUPDEF s'applique.

Réducteur électronique							
Descripteur	**Paramètre**						**Signification**
EGDEL	**1.**						Suppression de la définition de couplage pour l'axe asservi
	AXIS : axe asservi						
EGDEF	**1.**	**2. / 4. / 6. / 8. / 10.**	**3. / 5. / 7. / 9. / 11.**				Définition d'un réducteur électronique
	AXIS : axe asservi	AXIS : axe pilote	INT : type de couplage				
EGON	**1.**	**2.**	**3. / 6. / 9. / 12. / 15.**	**4. / 7. / 10. / 13. / 16.**	**5. / 8. / 11. / 14. / 17.**		Activation du réducteur électronique sans synchronisation
	AXIS : axe asservi	STRING : mode de changement de bloc	AXIS : axe pilote	REAL : numérateur du facteur de couplage	REAL : dénominateur du facteur de couplage		

Réducteur électronique									
Descripteur	**Paramètre**								**Signification**
EGONSYN	**1.**	**2.**	**3.**	**4. / 8. / 12. / 16. / 20.**	**5. / 9. / 13. / 17. / 21.**	**6. / 10. / 14. / 18. / 22.**	**7. / 11. / 15. / 19. / 23.**		Activation du réducteur électronique avec synchronisation
	AXIS : axe asservi	STRING : mode de changement de bloc	REAL : position synchrone de l'axe asservi	AXIS : axe pilote	REAL : position synchrone de l'axe pilote	REAL : numérateur du facteur de couplage	REAL : dénominateur du facteur de couplage		
EGONSYNE	**1.**	**2.**	**3.**	**4.**	**5. / 9. / 13. / 17. / 21.**	**6. / 10. / 14. / 18. / 22.**	**7. / 11. / 15. / 19. / 23.**	**8. / 12. / 16. / 20. / 24.**	Activation du réducteur électronique avec synchronisation et spécification du mode d'accostage
	AXIS : axe asservi	STRING : mode de changement de bloc	REAL : position synchrone de l'axe asservi	STRING : mode d'accostage	AXIS : axe pilote	REAL : position synchrone de l'axe pilote	REAL : numérateur du facteur de couplage	REAL : dénominateur du facteur de couplage	
EGOFS	**1.**	**2e - n-ième**							Désactivation sélective du réducteur électronique
	AXIS : axe asservi	AXIS : axe pilote							
EGOFC	**1.**								Désactivation du réducteur électronique (variante pour broches uniquement)
	AXIS : broche asservie								

Grignotage					
Descripteur	**Paramètre**				**Signification**
	1.	**2.**	**3.**	**4.**	
PUNCHAAC	REAL : distance minimale entre les trous	REAL : accélération initiale	REAL : distance maximale entre les trous	REAL : accélération finale	Activation de l'accélération en fonction de la distance

Fonctions d'information dans le système de fichiers passif				
Descripteur	**Paramètre**			**Signification**
	1.	**2.**	**3.**	
FILEDATE	VAR INT : message d'erreur	CHAR[160] : nom de fichier	VAR CHAR[8] : date au format "jj.mm.aa"	Fournit la date du dernier accès en écriture à un fichier
FILETIME	VAR INT : message d'erreur	CHAR[160] : nom de fichier	VAR CHAR[8] : heure au format "hh.mm.ss"	Fournit l'heure du dernier accès en écriture à un fichier
FILESIZE	VAR INT : message d'erreur	CHAR[160] : nom de fichier	VAR INT : taille de fichier	Fournit la taille actuelle d'un fichier
FILESTAT	VAR INT : message d'erreur	CHAR[160] : nom de fichier	VAR CHAR[5] : donnée au format "rwxsd"	Fournit l'état d'un fichier en relation avec les droits suivants : • Lecture (r : read) • Écriture (w : write) • Exécution (x : execute) • Affichage (s : show) • Suppression (d : delete)
FILEINFO	VAR INT : message d'erreur	CHAR[160] : nom de fichier	VAR CHAR[32] : donnée au format "rwxsd nnnnnnnn jj.mm.aa hh:mm:ss"	Fournit, pour un fichier, l'ensemble des informations lisibles via FILEDATE, FILETIME, FILESIZE et FILESTAT.

Conteneur d'axes		
Descripteur	**Paramètre**	**Signification**
	1er - n-ième	
AXCTSWE	AXIS : conteneur d'axes	Rotation d'un conteneur d'axes
AXCTSWED	AXIS : conteneur d'axes	Rotation d'un conteneur d'axes (variante d'instruction pour la mise en service !)
AXCTSWEC :	AXIS : conteneur d'axes	Suppression du déblocage de la rotation du conteneur d'axes

Couplage maître-esclave		
Descripteur	**Paramètre**	**Signification**
	1er - n-ième	
MASLON	AXIS : descripteur d'axe	Activation du couplage maître-esclave
MASLOF	AXIS : descripteur d'axe	Désactivation du couplage maître-esclave
MASLOFS	AXIS : descripteur d'axe	Désactivation du couplage maître-esclave et freinage automatique des broches esclaves
MASLDEF	AXIS : descripteur d'axe	Définition du couplage maître-esclave Le dernier axe est l'axe maître.
MASLDEL	AXIS : descripteur d'axe	Ouverture du couplage maître/esclave et suppression de la définition du groupe

Correction en ligne de la longueur d'outil			
Descripteur	**Paramètre**		**Signification**
	1.	**2.**	
TOFFON	AXIS : sens de correction	REAL : valeur d'offset dans le sens de la correction	Activation de la correction en ligne de la longueur d'outil dans le sens de correction spécifié
TOFFOF	AXIS : sens de correction		Annulation de la correction en ligne de la longueur d'outil dans le sens de correction spécifié

SERUPRO		
Descripteur	**Paramètre**	**Signification**
IPTRLOCK		Début de la section de programme interdite à la recherche
IPTRUNLOCK		Fin de la section de programme interdite à la recherche

Retrait		
Descripteur	**Paramètre**	**Signification**
	1er - n-ième	
POLFMASK	AXIS : nom de l'axe géométrique ou machine	Déblocage des axes pour le retrait rapide (sans corrélation entre les axes)
POLFMLIN	AXIS : nom de l'axe géométrique ou machine	Déblocage des axes pour le retrait rapide linéaire

Retrait				
Descripteur	**Paramètre**			**Signification**
POLFA	**1.**	**2.**	**3.**	Position de retrait pour axes individuels
	AXIS : descripteur d'axe de canal	INT : type	REAL : valeur	

Prévention des collisions			
Descripteur	**Paramètre**		**Signification**
	1.		
PROTA	STRING : "R"		Demander un recalcul du modèle de collision
PROTS	**1.**	**2e - n-ième**	Régler l'état des zones de protection
	CHAR : état	STRING : nom de zone de protection	

Adaptation intelligente de la charge					
Descripteur	**Paramètre**				**Signification**
CADAPTON	**1.**	**2.**	**3.**	**4.**	Activer l'adaptation de la charge
	INT : état	AXIS : nom de l'axe machine	INT : grandeur d'entrée	REAL : valeur d'entrée (facultatif)	
CADAPTOF	**1.**	**2.**	**3.**		Désactiver l'adaptation de la charge
	INT : état	AXIS : nom de l'axe machine	INT : grandeur d'entrée		

Adaptation de l'à-coup		
Descripteur	**Paramètre**	**Signification**
CALCFIR		Adaptation du filtre d'à-coup FIR au mode dynamique CALCFIR est utilisé dans le cycle constructeur CUST_832.

4.5 Procédures prédéfinies dans les actions synchrones

Les procédures prédéfinies suivantes sont disponibles exclusivement dans les actions synchrones.

Procédures synchrones		
Descripteur	**Paramètre**	**Signification**
STOPREOF		Annulation de l'arrêt du prétraitement des blocs Une action synchrone avec une instruction STOPREOF provoque un arrêt du prétraitement des blocs après le prochain bloc prétraité (= bloc à exécuter). L'arrêt du prétraitement est annulé à la fin de ce bloc ou quand la condition pour STOPREOF est remplie. L'ensemble des instructions d'actions synchrones avec STOPREOF est alors considéré comme exécuté.
RDISABLE		blocage de l'introduction via l'interface.
DELDTG	**1.** AXIS : axe pour effacement axial de la distance restant à parcourir (facultatif). Si l'axe est omis, l'effacement de la distance restant à parcourir est déclenché pour la trajectoire.	Effacement de la distance restant à parcourir Une action synchrone avec une instruction DELDTG provoque un arrêt du prétraitement des blocs après le prochain bloc prétraité (= bloc à exécuter). L'arrêt du prétraitement est annulé à la fin de ce bloc ou quand la première condition pour DELDTG est remplie. Dans $AA_DELT[<Axe>] se trouve la distance axiale jusqu'au point de destination pour l'effacement axial de la distance restant à parcourir et dans $AC_DELT le trajet résiduel.

Coordination de programmes Cycles technologiques		
Descripteur	**Paramètre**	**Signification**
	1.	
LOCK	INT : ID de l'action synchrone à bloquer	Blocage d'une action synchrone avec ID ou arrêt d'un cycle technologique Un ou plusieurs ID peuvent être programmés.
UNLOCK	INT : ID de l'action synchrone à libérer	Libération d'une action synchrone avec ID ou poursuite d'un cycle technologique Un ou plusieurs ID peuvent être programmés.
ICYCON		Exécution de chaque bloc d'un cycle technologique dans une période d'appel de l'interpolateur distincte après ICYCON
ICYCOF		Exécution de tous les blocs d'un cycle technologique dans une période d'appel de l'interpolateur après ICYCOF

Fonctions polynomiales						
Descripteur	**Paramètre**					**Signification**
SYNFCT	**1.**	**2.**	**3.**			Lorsque la condition est remplie dans l'action synchrone au déplacement, le polynôme défini par la première expression est appliqué à la variable d'entrée. La valeur obtenue est ensuite écrêtée vers le bas et vers le haut et affectée aux variables résultat.
	INT : numéro de la fonction polynomiale définie avec FCTDEF	VAR REAL : variable résultat*)	VAR REAL : variable d'entrée**)			
FTOC	**1.**	**2.**	**3.**	**4.**	**5.**	Modification de la correction fine d'outil selon une fonction définie avec FCTDEF (polynôme de 3ème degré au maximum) Le numéro utilisé ici doit être spécifié dans l'instruction FCTDEF.
	INT : numéro de la fonction polynomiale définie avec FCTDEF	VAR REAL : variable d'entrée**)	INT: longueur 1, 2, 3	INT : numéro de canal	INT : numéro de broche	

*) Seules des variables système spéciales sont autorisées en tant que variables résultat (voir la Description fonctionnelle Actions synchrones).

**) Seules des variables système spéciales sont autorisées en tant que variables d'entrée (voir la Description fonctionnelle Actions synchrones).

4.6 Fonctions prédéfinies

L'appel d'une procédure prédéfinie lance l'exécution d'une fonction CN prédéfinie qui, contrairement à la procédure prédéfinie, fournit une valeur de retour. L'appel de la fonction prédéfinie peut figurer sous forme d'opérande dans l'expression.

Système de coordonnées						
Descripteur	**Valeur de retour**	**Paramètre**				**Signification**
		1.	**2.**	**3. - 15.**	**4. - 16.**	
CTRANS	FRAME	AXIS : descripteur d'axe	REAL : décalage	AXIS : descripteur d'axe	REAL : décalage	Translation : décalage d'origine GROSSIER pour plusieurs axes
CFINE	FRAME	AXIS : descripteur d'axe	REAL : décalage	AXIS : descripteur d'axe	REAL : décalage	Translation : décalage d'origine FIN pour plusieurs axes
CSCALE	FRAME	AXIS : descripteur d'axe	REAL : facteur d'échelle	AXIS : descripteur d'axe	REAL : facteur d'échelle	Scale : facteur d'échelle pour plusieurs axes
		1.	**2.**	**3. et 5.**	**4. et 6.**	
CROT	FRAME	AXIS : descripteur d'axe	REAL : rotation	AXIS : descripteur d'axe	REAL : rotation	Rotation : rotation du système de coordonnées actuel Nombre maximal de paramètres : 6 (un descripteur d'axe et une valeur par axe géométrique).
CROTS	FRAME	AXIS : descripteur d'axe	REAL : rotation avec un angle solide	AXIS : descripteur d'axe	REAL : rotation avec un angle solide	Rotation : rotation du système de coordonnées actuel avec un angle solide Nombre maximal de paramètres : 6 (un descripteur d'axe et une valeur par axe géométrique).
CMIRROR		**1.**	**2. - 8.**			Mirror : fonction miroir par rapport à un axe de coordonnées
	FRAME	AXIS	AXIS			

Système de coordonnées						
Descripteur	**Valeur de retour**	**Paramètre**				**Signification**
		1.	**2.**			
CRPL	FRAME	INT : axe de rotation	REAL : angle de rotation			Rotation de frame dans un plan quelconque
ADDFRAME	INT : 0 : OK 1 : la destination spécifiée (String) est incorrecte 2 : le frame de destination n'est pas configuré 3 : la rotation n'est pas autorisée dans le frame	FRAME : frame additif mesuré ou calculé	STRING : frame de destination spécifié			Calcul du frame de destination spécifié par la chaîne Le frame de destination est calculé de manière que le nouveau frame global résulte d'une concaténation de l'ancien frame global et du frame transmis.
INVFRAME	FRAME	**1.** FRAME				Calcul du frame inverse à partir d'un frame La concaténation d'un frame et de son inverse génère toujours un frame nul
MEAFRAME	FRAME	**1.** REAL[3,3] : coordonnées des points mesurés dans l'espace	**2.** REAL[3,3] : coordonnées des points de consigne	**3.** VAR REAL : variable avec laquelle sont retournées des informations sur la qualité du calcul de FRAME		Calcul du frame à partir de 3 points de mesure dans l'espace

Fonctions géométriques					
Descripteur	**Valeur de retour**	**Paramètre**			**Signification**
		1.	**2.**	**3.**	
CALCDAT	BOOL : état d'erreur	VAR REAL [n, 2] : tableau (abscisse, ordonnée) des points 1 à n	INT : nombre de points	VAR REAL [3]: résultat : abscisse, ordonnée et rayon du centre du cercle calculé	Calcule les coordonnées du centre et le rayon du cercle à partir de 3 ou 4 points. Les points doivent être différents.
INTERSEC	BOOL : état d'erreur	VAR REAL [11] : premier élément de contour	VAR REAL [11] : deuxième élément de contour	VAR REAL [2] : vecteur résultat pour les coordonnées du point d'intersection : abscisse et ordonnée	Calcule les coordonnées du point d'intersection entre deux éléments de contour. L'état d'erreur indique si un point d'intersection a été trouvé.

Fonctions de table de courbes								
Descripteur	**Valeur de retour**	**Paramètre**						**Signification**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTAB	REAL : position de l'axe asservi	REAL : position de l'axe pilote	INT : numéro de table	VAR REAL[ ] : résultat pas	AXIS : axe asservi pour homothétie	AXIS : axe pilote pour homothétie		Détermine la position de l'axe asservi pour la position de l'axe pilote spécifiée à partir de la table de courbes. Si les paramètres 4/5 ne sont pas programmés, l'échelle par défaut est utilisée pour le calcul.
CTABINV	REAL : position de l'axe pilote	REAL : position de l'axe asservi	REAL : position pilote	INT : numéro de table	VAR REAL[ ] : résultat pas	AXIS : axe asservi pour homothétie	AXIS : axe pilote pour homothétie	Détermine la position de l'axe pilote pour la position de l'axe asservi spécifiée à partir de la table de courbes. Si les paramètres 5/6 ne sont pas programmés, l'échelle par défaut est utilisée pour le calcul.
CTABID	INT : numéro de table de courbes	INT : numéro d'entrée dans la mémoire	STRING : lieu de stockage : "SRAM", "DRAM"					Détermine le numéro de table de courbes enregistré dans la mémoire sous le numéro spécifié.

Fonctions de table de courbes								
Descripteur	**Valeur de retour**	**Paramètre**						**Signification**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABISLOCK	INT : état du blocage	INT : numéro de table						Détermine l'état du blocage de la table de courbes : > 0 : la table est bloquée 1 : CTABLOCK 2 : couplage actif 3 : CTABLOCK et couplage actif 0 : la table n'est pas bloquée -1 : La table n'existe pas.
CTABEXISTS	INT : existence	INT : numéro de table						Détermine l'existence de la table de courbes dans la mémoire statique ou dynamique de la CN 0 : FALSE 1 : TRUE
CTABMEMTYP	INT : lieu d'enregistrement	INT : numéro de table						Détermine le lieu d'enregistrement de la table de courbes : 1 : DRAM 0 : SRAM -1 : La table n'existe pas.
CTABPERIOD	INT : périodicité	INT : numéro de table						Détermine la périodicité de la table de courbes : 0 : non périodique 1 : périodique dans l'axe pilote 2 : périodique dans l'axe pilote et l'axe asservi -1 : La table n'existe pas.
CTABNO	INT : nombre de tables de courbes							Détermine le nombre de tables de courbes définies (dans les mémoires statique et dynamique de la CN)

Fonctions de table de courbes								
Descripteur	**Valeur de retour**	**Paramètre**						**Signification**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABNOMEM	INT : nombre de tables de courbes	STRING : lieu de stockage : "SRAM", "DRAM"						Détermine le nombre de tables de courbes définies dans la mémoire spécifiée
CTABFNO	INT : nombre de tables	STRING : lieu de stockage : "SRAM", "DRAM"						Détermine le nombre de tables de courbes encore possibles dans la mémoire spécifiée
CTABSEG	INT : nombre de segments de courbe	STRING : lieu de stockage : "SRAM", "DRAM"	STRING : type de segment : "L" : linéaire "P" : polynôme					Détermine le nombre de segments de courbe utilisés du type spécifié dans la mémoire spécifiée >=0 : Nombre -1 : type de mémoire non valide Si le paramètre 2 n'est pas programmé, la somme des segments linéaires et polynomiaux est générée en sortie.
CTABFSEG	INT : nombre de segments de courbe	STRING : lieu de stockage : "SRAM", "DRAM"	STRING : type de segment : "L" : linéaire "P" : polynôme					Détermine le nombre de segments de courbe du type spécifié encore possibles dans la mémoire spécifiée >=0 : Nombre -1 : type de mémoire non valide
CTABSEGID	INT : nombre de segments de courbe	INT : numéro de table	STRING : type de segment : "L" : linéaire "P" : polynôme					Détermine le nombre de segments de courbe du type spécifié utilisés par la table de courbes >=0 : Nombre -1 : La table n'existe pas.
CTABMSEG	INT : nombre de segments de courbe	STRING : lieu de stockage : "SRAM", "DRAM"	STRING : type de segment : "L" : linéaire "P" : polynôme					Détermine le nombre maximal de segments de courbe du type spécifié possibles dans la mémoire spécifiée >=0 : Nombre -1 : La table n'existe pas.

Fonctions de table de courbes								
Descripteur	**Valeur de retour**	**Paramètre**						**Signification**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABPOL	INT : nombre de polynômes de courbe	STRING : lieu de stockage : "SRAM", "DRAM"						Détermine le nombre de polynômes de courbe utilisés dans la mémoire spécifiée >=0 : Nombre -1 : La table n'existe pas.
CTABPOLID	INT : nombre de polynômes de courbe	INT : numéro de table						Détermine le nombre de polynômes de courbe utilisés par la table de courbes >=0 : Nombre -1 : La table n'existe pas.
CTABFPOL	INT : nombre de polynômes de courbe	STRING : lieu de stockage : "SRAM", "DRAM"						Détermine le nombre maximal de polynômes de courbe possibles dans la mémoire spécifiée : >=0 : Nombre -1 : La table n'existe pas.
CTABMPOL	INT : nombre de polynômes de courbe	STRING : lieu de stockage : "SRAM", "DRAM"						Détermine le nombre maximal de polynômes de courbe possibles dans la mémoire spécifiée : >=0 : Nombre -1 : La table n'existe pas.
CTABSSV	REAL : position de l'axe asservi	REAL : position de l'axe pilote	INT : numéro de table	VAR REAL[ ] : résultat pas	AXIS : axe asservi pour homothétie	AXIS : axe pilote pour homothétie		Détermine la position de l'axe asservi au début du segment de courbe correspondant à la valeur de l'axe pilote spécifié.
CTABSEV	REAL : position de l'axe asservi	REAL : position de l'axe pilote	INT : numéro de table	VAR REAL[ ] : résultat pas	AXIS : axe asservi pour homothétie	AXIS : axe pilote pour homothétie		Détermine la position de l'axe asservi à la fin du segment de courbe correspondant à la valeur de l'axe pilote spécifié.
CTABTSV	REAL : position de l'axe asservi	INT : numéro de table	VAR REAL[ ] : résultat pas début de la table	AXIS : axe asservi				Détermine la position de l'axe asservi au début de la table de courbes.

Fonctions de table de courbes								
Descripteur	**Valeur de retour**	**Paramètre**						**Signification**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABTEV	REAL : position de l'axe asservi	INT : numéro de table	VAR REAL[ ] : résultat pas fin de la table	AXIS : axe asservi				Détermine la position de l'axe asservi à la fin de la table de courbes.
CTABTSP	REAL : position de l'axe pilote	INT : numéro de table	VAR REAL[ ] : résultat pas début de la table	AXIS : axe pilote				Détermine la position de l'axe pilote au début de la table de courbes.
CTABTEP	REAL : position de l'axe pilote	INT : numéro de table	VAR REAL[ ] : résultat pas fin de la table	AXIS : axe pilote				Détermine la position de l'axe pilote à la fin de la table de courbes.
CTABTMIN	REAL : valeur minimale	INT : numéro de table	REAL : intervalle de valeur pilote limite inférieure	REAL : intervalle de valeur pilote limite supérieure	AXIS : axe asservi	AXIS : axe pilote		Détermine la valeur minimale de l'axe asservi dans l'ensemble de la plage de définition de la table de courbes ou dans un intervalle défini.
CTABTMAX	REAL : valeur maximale	INT : numéro de table	REAL : intervalle de valeur pilote limite inférieure	REAL : intervalle de valeur pilote limite supérieure	AXIS : axe asservi	AXIS : axe pilote		Détermine la valeur maximale de l'axe asservi dans l'ensemble de la plage de définition de la table de courbes ou dans un intervalle défini.
Remarque : Les fonctions de table de courbes peuvent également être programmées dans des actions synchrones.								

Fonctions d'axe						
Descripteur	**Valeur de retour**	**Paramètre**				
		1.	**2.**	**3.**	**4.**	**Signification**
AXNAME	AXIS : descripteur d'axe	STRING [ ] : chaîne de caractères d'entrée				Conversion de la chaîne d'entrée en descripteur d'axe
AXSTRING	STRING[ ] : nom d'axe	AXIS : descripteur d'axe				Conversion du descripteur d'axe en chaîne de caractères

Fonctions d'axe						
Descripteur	**Valeur de retour**	**Paramètre**				
		1.	**2.**	**3.**	**4.**	**Signification**
ISAXIS	BOOL : axe existant (TRUE) ou non (FALSE)	INT : numéro de l'axe géométrique (1 à 3)				Vérifie si l'axe géométrique 1 à 3 spécifié en paramètre figure bien dans les paramètres machine PM20050 $MC_AXCONF_GEOAX_ASSIGN_TAB.
SPI	AXIS : descripteur d'axe	INT : numéro de broche				Conversion du numéro de broche en descripteur d'axe
AXTOSPI	INT : numéro de broche	AXIS : descripteur d'axe				Conversion du descripteur d'axe en numéro de broche
MODAXVAL	REAL : valeur modulo	AXIS : descripteur d'axe	REAL : position de l'axe			Calcul du reste modulo à partir de la position d'axe saisie Si l'axe spécifié n'est pas un axe modulo, la position de l'axe est retournée sans modification.
POSRANGE	BOOL : position de consigne dans la fenêtre de positions (TRUE) ou non (FALSE)	AXIS : descripteur d'axe	REAL : position de référence dans le système de coordonnées	REAL : largeur de la fenêtre de position	INT : système de coordonnées	Détermine si la position de consigne d'un axe se trouve dans une fenêtre autour d'une position de référence spécifiée

Gestion des outils					
Descripteur	**Valeur de retour**	**Paramètre**			**Signification**
		1.	**2.**	**3.**	
CHKDM	INT : état : résultat de la vérification	INT : numéro de magasin	INT : numéro D		Vérifie l'univocité du numéro D dans un magasin
CHKDNO	INT : état : résultat de la vérification	INT : numéro T du 1er outil	INT : numéro T du 2e outil	INT : numéro D	Vérifie l'univocité du numéro D
GETACTT	INT : état	INT : numéro T	STRING[32] : nom d'outil		Détermine l'outil actif parmi un groupe d'outils de même nom
GETACTTD	INT : état : résultat de la vérification	VAR INT : numéro T trouvé (valeur de retour)	INT : numéro D		Détermine le numéro T correspondant au numéro D absolu

Gestion des outils					
Descripteur	**Valeur de retour**	**Paramètre**			**Signification**
		1.	**2.**	**3.**	
GETDNO	INT : numéro D	INT : numéro T	INT : numéro de tranchant		Détermine le numéro D du tranchant de l'outil T
GETT	INT : numéro T	STRING[32] : nom d'outil	INT : numéro d'outil frère		Détermine le numéro T associé à un nom d'outil
NEWT	INT : numéro T	STRING[32] : nom d'outil	INT : numéro d'outil frère		Crée un nouvel outil (préparer les données d'outil) Le numéro d'outil frère peut être omis.
TOOLENV	INT : état	STRING : nom			Enregistre l'environnement d'outil portant le nom spécifié dans la mémoire statique de la CN
DELTOOLENV	INT : état	STRING : nom			Supprime l'environnement d'outil portant le nom spécifié dans la mémoire statique de la CN Supprime tous les environnements d'outil si aucun nom n'est spécifié.
GETTENV	INT : état	STRING : nom	VAR INT : Numéro T [0] Numéro D [1] Numéro DL [2]		Détermine le numéro T, le numéro D et le numéro DL dans un environnement d'outil sous le nom spécifié

Arithmétique					
Descripteur	**Valeur de retour**	**Paramètre**			**Signification**
		1.	**2.**	**3.**	
SIN	REAL	REAL			Sinus
ASIN	REAL	REAL			Arc sinus
COS	REAL	REAL			Cosinus
ACOS	REAL	REAL			Arc cosinus
TAN	REAL	REAL			Tangente
ATAN2	REAL	REAL	REAL		Arc tangente 2
SQRT	REAL	REAL			Racine carrée
POT	REAL	REAL			Puissance 2
TRUNC	REAL	REAL			Partie entière
ROUND	REAL	REAL			Arrondir au chiffre inférieur
ROUNDUP	REAL	REAL			Arrondir au chiffre supérieur
ABS	REAL	REAL			Valeur absolue
LN	REAL	REAL			Logarithme naturel
EXP	REAL	REAL			Fonction exponentielle e^x
MINVAL	REAL	REAL	REAL		Détermine la plus petite valeur des deux paramètres

Arithmétique					
Descripteur	**Valeur de retour**	**Paramètre**			**Signification**
		1.	**2.**	**3.**	
MAXVAL	REAL	REAL	REAL		Détermine la plus grande valeur des deux paramètres
BOUND	REAL : état de vérification	REAL : limite inférieure	REAL : limite supérieure	REAL : valeur de comparaison	Détermine si la valeur de comparaison se trouve dans les limites.
Remarque : Les fonctions arithmétiques peuvent également être programmées dans des actions synchrones. Le calcul ou l'exploitation de ces fonctions arithmétiques s'effectue alors pendant l'exécution des blocs. Le paramètre d'action synchrone $AC_PARAM[<n>] peut également être utilisé pour les calculs et en tant que mémoire tampon.					

Fonctions de chaîne de caractères					
Descripteur	**Valeur de retour**	**Paramètre**			**Signification**
		1.	**2.**	**3.**	
ISNUMBER	BOOL	STRING : chaîne d'entrée			Vérifie si la chaîne d'entrée peut être convertie en un nombre
NUMBER	REAL	STRING : chaîne d'entrée			Convertit la chaîne d'entrée en un nombre
TOUPPER	STRING	STRING : chaîne d'entrée			Convertit la chaîne d'entrée en majuscules
TOLOWER	STRING	STRING : chaîne d'entrée			Convertit la chaîne d'entrée en minuscules
STRLEN	INT	STRING : chaîne d'entrée			Détermine la longueur de la chaîne d'entrée jusqu'à la fin (/0)
INDEX	INT	STRING : chaîne d'entrée	CHAR : caractère de recherche		Détermine la position du caractère dans la chaîne d'entrée de gauche à droite. Le 1er caractère de la chaîne à partir de la gauche porte l'indice 0.
RINDEX	INT	STRING : chaîne d'entrée	CHAR : caractère de recherche		Détermine la position du caractère dans la chaîne d'entrée de droite à gauche. Le 1er caractère de la chaîne à partir de la droite porte l'indice 0.
MINDEX	INT	STRING : chaîne d'entrée	STRING : caractère de recherche		Détermine la position d'un caractère spécifié dans le 2e paramètre de la chaîne d'entrée de gauche à droite. Le 1er caractère de la chaîne d'entrée à partir de la gauche porte l'indice 0.

Fonctions de chaîne de caractères					
Descripteur	**Valeur de retour**	**Paramètre**			**Signification**
		1.	**2.**	**3.**	
SUBSTR	STRING	STRING : chaîne d'entrée	INT	INT	Détermine la chaîne partielle de la chaîne d'entrée définie par le début (2e paramètre) et le nombre de caractères (3e paramètre).
SPRINT	STRING	STRING : chaîne d'entrée			Détermine la chaîne d'entrée formatée

Fonctions pour cycles de mesure								
Descripteur	**Valeur de retour**	**Paramètre**						**Signification**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CALCPOSI	INT : état	REAL[3] : position de départ dans le SCP	REAL[3] : forçage de déplacement relatif à la position de départ	REAL[5] : distances minimales par rapport aux limites de surveillance	REAL[3] : champ de retour pour l'éventuelle distance rel.	BOOL : conversion du système de mesure oui/non	INT : type de surveillance de limites	Vérifie si, en partant d'un point de départ donné, les axes géométriques peuvent être déplacés sur une distance définie sans dépassement des limites des axes Si la distance prédéfinie ne peut être parcourue sans dépassement, la fonction retourne la valeur maximale admissible.
GETTCOR	INT : état	REAL [11] :	STRING : composante de longueur d'outil : système de coordonnées	STRING : nom de l'environnement d'outil	INT : n° T interne de l'outil	INT : numéro de tranchant (n° D) de l'outil	INT : numéro de la correction fonction du lieu (n° DL de l'outil)	Détermine les longueurs d'outil et les composantes de longueur d'outil à partir de l'environnement d'outil ou de l'environnement actuel
LENTOAX	INT : état	INT[3] : affectation des axes géométriques	REAL[3] : matrice de représentation des longueurs d'outils dans le système de coordonnées	STRING : système de coordonnées pour l'affectation				Détermine les informations relatives à l'affectation des longueurs d'outil L1, L2, L3 de l'outil actif par rapport à l'abscisse, l'ordonnée et la cote L'affection aux axes géométriques est également fonction des frames ainsi que du plan actif (G17 - G19).

SETTCOR	INT : état	**1.**	**2.**	**3.**	**4.**	**5.**	**6.**	**7.**	**8.**	**9.**	
		REAL [3] : vecteur de correction dans l'espace	STR. : descripteur de composante	INT : composant(s) à corr. 0 - 11	INT : type d'opération d'écriture 0 - 3	INT : indice des axes géo.	STRING : nom de l'environnement d'outil	INT : n° T int. de l'outil	INT : n° D de l'outil	INT : n° DL de l'outil	Corrige les composantes d'outil en tenant compte de toutes les conditions marginales influant sur l'évaluation des composantes individuelles

Autres fonctions								
Descripteur	**Valeur de retour**	**Paramètre**						**Signification**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
STRINGIS	INT : info sur la chaîne de caractères	STRING : nom de l'élément à vérifier						Vérifie si la chaîne de caractères spécifiée est disponible en tant qu'élément de langage de programmation CN dans l'étendue actuelle du langage
ISVAR	BOOL : variable connue oui/non	STRING : nom de la variable						Vérifie si le paramètre de transfert contient une variable connue de la CN (paramètre machine, donnée de réglage, variable système ou variable générale telle qu'une donnée utilisateur GUD)
GETVARTYP	INT : type de données	STRING : nom de la variable						Détermine le type de données d'une variable système ou utilisateur
GETVARPHU	INT : valeur numérique de l'unité physique	STRING : nom de la variable						Détermine l'unité physique d'une variable système ou utilisateur
GETVARAP	INT : niveau de protection pour l'accès	STRING : nom de la variable	STRING : type d'accès					Détermine le droit d'accès à une variable système ou utilisateur

Autres fonctions								
Descripteur	**Valeur de retour**	**Paramètre**						**Signification**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
GETVARLIM	INT : état	STRING : nom de la variable	CHAR : indique la valeur limite à lire	VAR REAL : valeur limite retournée				Détermine la valeur limite inférieure ou supérieure d'une variable système ou utilisateur
GETVARDFT	INT : état	STRING : nom de la variable	VAR REAL/ STRING/ FRAME : valeur par défaut retournée	INT : indice de la première dimension (facultatif)	INT : indice de la deuxième dimension (facultatif)	INT : indice de la troisième dimension (facultatif)		Détermine la valeur par défaut d'une variable système ou utilisateur
COLLPAIR	INT : résultat du contrôle	STRING : nom de la 1re zone de protection	STRING : nom de la 2e zone de protection	BOOL : inhibition des alarmes (facultatif)				Vérifie l'appartenance à une paire de collision
PROTD	REAL : distance entre les deux zones de protection	STRING : nom de la 1re zone de protection	STRING : nom de la 2e zone de protection	VAR REAL : valeur de retour : vecteur de distance tridimensionnel	BOOL : système de mesure pour la distance et le vecteur de distance (facultatif)			Détermine la distance entre les deux zones de protection spécifiées
DELOBJ	INT : numéro d'erreur	STRING : type du composant à supprimer	INT : indice de début du composant à supprimer (facultatif)	INT : indice de fin du composant à supprimer (facultatif)	BOOL : inhibition des alarmes (facultatif)			Supprime les éléments de chaînes cinématiques, de zones de protection, d'éléments de zones de protection, de paires de collision et de données de transformation
NAMETOINT	INT : indice de variables système	STRING : nom du tableau de variables système	STRING : chaîne de caractères / nom	BOOL : inhibition des alarmes (facultatif)				Détermine à partir de la chaîne de caractères l'indice de variables système correspondant

Autres fonctions								
Descripteur	**Valeur de retour**	**Paramètre**						**Signification**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
ORISOLH	INT : numéro d'erreur	INT : commande le comportement de la fonction	REAL : premier angle	REAL : deuxième angle				Assiste l'utilisateur au réglage des positions d'axe rotatif d'une machine de telle sorte qu'un outil de tournage soit orienté dans une position définie, relative à la pièce indépendante de la cinématique. Condition : Une transformation 6 axes paramétrée avec des chaînes cinématiques doit être active.
CORRTRAFO	INT : numéro d'erreur	REAL : vecteur de correction	INT : élément à modifier	INT : mode de correction	BOOL : inhibition des alarmes (facultatif)			Modification des vecteurs d'offset ou vecteurs directionnels des axes d'orientation dans le modèle cinématique de la machine.
CORRTC	INT : numéro d'erreur	REAL : vecteur de correction	INT : élément à modifier	INT : mode de correction	BOOL : inhibition des alarmes (facultatif)			Modification des vecteurs d'offset ou des vecteurs de direction des organes porte-outil orientables après mesurage de la machine.

4.7 Langue actuelle dans l'IHM

Le tableau ci-après contient toutes les langues disponibles dans l'interface utilisateur.

La langue actuellement réglée peut être interrogée dans le programme pièces et dans les actions synchrones, au moyen de la variable système suivante :

$AN_LANGUAGE_ON_HMI = <valeur>

<valeur>	Langue	Code langue
1	Allemand (Allemagne)	DEU
2	Français	FRA
3	Anglais (Royaume-Uni)	ENG
4	Espagnol	ESP
6	Italien	ITA
7	Néerlandais	NLD
8	Chinois (simplifié)	CHS
9	Suédois	SVE
18	Hongrois	HUN
19	Finnois	FIN
28	Tchèque	CSY
50	Portugais (Brésil)	PTB
53	Polonais	PLK
55	Danois	DAN
57	Russe	RUS
68	Slovaque	SKY
72	Roumain	ROM
80	Chinois (traditionnel)	CHT
85	Coréen	KOR
87	Japonais	JPN
89	Turc	TRK

Remarque

L'actualisation de $AN_LANGUAGE_ON_HMI s'effectue :

- après le démarrage du système,
- après un Reset de la CN et/ou de l'AP
- après la commutation sur une autre CN dans le cadre de M2N,
- après le basculement de la langue sur l'IHM.

Annexe

A

A.1 Liste des abréviations

A	
A	Sortie
ASCII	American Standard Code for Information Interchange: code standard américain pour l'échange d'information
ASIC	Application Specific Integrated Circuit: circuit utilisateur
ASUP	Asynchrones Unterprogramm : sous-programme asynchrone
AUTO	Mode de fonctionnement "Automatic"
AUXFU	Auxiliary Function: fonction auxiliaire
AWL	Liste d'instructions

B	
BAG	Groupe à mode de fonctionnement commun
BCD	Binary Coded Decimals: décimales codées en binaire
BICO	Binector Connector
BIN	Binary Files: fichiers binaires
BKS	Système de coordonnées de base
BM	Message de fonctionnement
BO	Binector Output
BTSS	Operator Panel Interface : interface du pupitre opérateur

C	
CLC	Régulation de la distance
CNC	Computerized Numerical Control: Computerunterstützte numerische Steuerung
COM	Communication
CP	Communication Processor
CPU	Central Processing Unit: unité centrale de traitement
CST	Configured Stop: Arrêt configuré

D	
DB	Bloc de données (AP)
DBB	Octet de bloc de données (AP)
DBD	Double mot de bloc de données (AP)
DBW	Mot de bloc de données (AP)
DBX	Bit de bloc de données (AP)
DDS	Drive Data Set: jeu de paramètres d'entraînement

D	
DIR	Directory: répertoire
DO	Drive Object
DRF	Differential Resolver Function: fonction de résolveur différentiel (manivelle électronique)
DRY	Dry Run: avance de marche d'essai
DW	Mot de données
DWORD	Double mot (soit 32 bits)

E	
E	Entrée
EES	Execution from External Storage
E/A	Entrée/sortie
ESR	Arrêt étendu et retrait
ETC	Touche ETC ">" ; permet d'étendre la barre de touches logicielles dans le même menu

F	
FB	Bloc fonctionnel (AP)
FC	Function Call: Bloc fonctionnel (AP)
FDD	Feed Disable: Blocage de l'avance
FdStop	Feed Stop: Arrêt avance
FIFO	First In First Out: premier entré, premier sorti ; mémoire fonctionnant sans adressage et dont les données sont lues dans l'ordre de leur stockage
FM	Message d'erreur
FUP	Logigramme (méthode de programmation pour AP)
FW	Firmware

G	
GEO	Géométrique, par exemple axe géométrique
GP	Programme de base (AP)
GUD	Global User Data: données utilisateur globales

H	
HEX	Abréviation pour nombre hexadécimal
HiFu	Fonction auxiliaire
HMI	Human Machine Interface: interface utilisateur SINUMERIK
HSA	Entraînement de la broche principale
HT	Handheld Terminal
HW	Hardware : matériel

I	
IBN	Mise en service
INC	Increment: déplacement en manuel incrémental
INI	Initializing Data: données d'initialisation
IPO	Interpolator

J	
JOG	Jogging : mode réglage

K	
KOP	Schéma à contacts (méthode de programmation pour AP)

L	
LED	Light Emitting Diode: diode électroluminescente
LMS	Lagemesssystem : système de mesure de position
LR	Lageregler : régulateur de position

M	
Main	Main program: programme principal (OB1, AP)
MCP	Machine Control Panel: tableau de commande machine
MD	Paramètre(s) machine
MDA	Manual Data Automatic: saisie manuelle des données
MDS	Motor Data Set: jeu de paramètres moteur
MELDW	Mot de signalisation
MKS	Système de coordonnées machine
MM	Motor Module
MPF	Main Program File: programme principal (CN)
MPI	Multi Point Interface
MSTT	Tableau de commande machine

N	
NC	Numerical Control: commande numérique avec traitement de blocs, plage de déplacement, etc.
NCU	Numerical Control Unit: unité matérielle de la CN
NCK	Numerical Control Kernel
NCSD	NC Start Disable
NST	Signal d'interface
NV	Décalage d'origine
NX	Numerical Extension: module d'extension d'axe

O	
OB	Bloc d'organisation dans l'AP
OEM	Original Equipment Manufacturer
OP	Operation Panel: pupitre opérateur

P	
PCU	PC Unit: boîtier PC (unité de traitement)
PG	Console de programmation
PLC	Programmable Logic Control: programmable
PN	PROFINET
PO	Power On
POS	Position/Positionnement
PPO	Parameter Prozessdaten Objekt : télégramme de données cyclique lors de la transmission avec PROFIBUS-DP et le profil "Entraînements à vitesse variable"
PPU	Panel Processing Unit (matériel central d'une commande CNC à pupitre opérateur, p. ex. SINUMERIK 828D)
PROFIBUS	Process Field Bus: bus de données série
PRT	Test du programme
PTP	Point to Point: point à point
PZD	Prozessdaten : partie données process d'un PPO

R	
REF	Fonction accostage de point de référence
REPOS	Fonction repositionnement
RESU	Retrace support
RID	Read In Disable
RP	Paramètre R : paramètre de calcul, variable utilisateur prédéfinie

S	
SA	Action synchrone
SBL	Single Block: bloc par bloc
SBT	Safe Brake Test
SCC	Safety Control Channel
SCL	Structured Control Language
SD	Settingdatum ou données de réglage
SDI	Safe Direction
SERUPRO	Search-Run by Program Test: Recherche en mode test du programme
SIC	Safety Info Channel
SKP	Skip: fonction de masquage d'un bloc de programme pièce
SLP	Safe Limited Position
SLS	Safely Limited Speed

S	
SMI	Sensor Module Integrated
SOS	Safe Operating Stop
SPF	Sub Program File: fichier de sous-programme (CN)
SS1	Safe Stop 1
SS2	Safe Stop 2
STO	Safe Torque Off
STW	Steuerwort : mot de commande
SUG	Vitesse circonférentielle de meule
SW	Logiciel

T	
TCU	Thin Client Unit
TM	Terminal Module (SINAMICS)
TO	Tool Offset: correction d'outil
TOA	Tool Offset Active: identificateur (type de fichier) pour corrections d'outil
TOFF	Correction en ligne de la longueur d'outil
TRANSMIT	Transform Milling Into Turning: transformation des coordonnées pour des opérations de fraisage sur un tour

U	
UP	Sous-programme
USB	Universal Serial Bus

V	
VDI	Interface de communication interne entre CN et AP

W	
WKS	Système de coordonnées pièce
WPD	Work Piece Directory: répertoire pièce
WZ	Outil
WZV	Gestion des outils

Z	
ZSW	Zustandswort : mot d'état (de l'entraînement)

A.2 Vue d'ensemble de la documentation

Une documentation détaillée relative aux fonctions de SINUMERIK 840D sl à partir de la version 4.8 SP4 est disponible sous Vue d'ensemble de la documentation 840D sl (https://support.industry.siemens.com/cs/ww/en/view/109766213).

Il est possible d'afficher les documents ou de les télécharger aux formats PDF et HTML5.

La documentation est divisée en plusieurs catégories comme suit :

- Utilisateur : Commande
- Utilisateur : Programmation
- Constructeur/S.A.V. : Fonctions
- Constructeur/S.A.V. : Matériel
- Constructeur/S.A.V. : Configuration/mise en service
- Constructeur/S.A.V. : Safety Integrated
- Constructeur/S.A.V. : SINUMERIK Integrate / MindApp
- Information et formation
- Constructeur/S.A.V. : SINAMICS

Index

$

$A_PROBE, 628, 635
$A_PROBE_LIMITED, 636
$AA_ACC, 131
$AA_ATOL, 897
$AA_COUP_ACT
  Couplage de deux axes par valeur pilote, 948
  Lors de déplacements conjugués, 925
$AA_ESR_ENABLE, 1010
$AA_FGREF, 115
$AA_FGROUP, 115
$AA_G0MODE, 186
$AA_LEAD_SP, 948
$AA_LEAD_SV, 948
$AA_MM, 629
$AA_MM1...4, 636
$AA_MW, 629
$AA_MW1...4, 636
$AC_ACT_PROG_NET_TIME, 1018
$AC_ACTUAL_PARTS, 1020
$AC_AXCTSWA, 913
$AC_AXCTSWE, 914
$AC_CTOL, 897
$AC_CTOL_G0_ABS, 190
$AC_CUT_INV, 828
$AC_CUTMOD, 827
$AC_CUTMOD_ANG, 827
$AC_CUTMODK, 827
$AC_CUTTING_TIME, 1017
$AC_CYCLE_TIME, 1017
$AC_DELAYFST, 878
$AC_ESR_TRIGGER, 1010
$AC_F_TYPE, 146
$AC_FGROUP_MASK, 115
$AC_FZ, 146
$AC_MEA, 628, 636
$AC_OLD_PROG_NET_TIME, 1018
$AC_OLD_PROG_NET_TIME_COUNT, 1018
$AC_OPERATING_TIME, 1017
$AC_OTOL, 897
$AC_OTOL_ G0_ABS, 190
$AC_PROG_NET_TIME_TRIGGER, 1018
$AC_REPOS_PATH_MODE, 886
$AC_REQUIRED_PARTS, 1020
$AC_S_TYPE, 100
$AC_SMAXVELO, 889
$AC_SMAXVELO_INFO, 889
$AC_SPECIAL_PARTS, 1020
$AC_STOLF, 189
$AC_SVC, 100
$AC_TOFF, 89
$AC_TOFFCR, 89
$AC_TOFFL, 89
$AC_TOFFR, 89
$AC_TOTAL_PARTS, 1020
$AC_TRAFO_CORR_ELEM_P, 754
$AC_TRAFO_CORR_ELEM_T, 754
$AC_TRAFO_ORIAX_LOC, 755
$AN_AXCTAS, 914
$AN_AXCTSWA, 913
$AN_ESR_TRIGGER, 1010
$AN_LANGUAGE_ON_HMI, 1396
$AN_POWERON_TIME, 1017
$AN_SETUP_TIME, 1017
$NT_CLOSE_CHAIN_T, 755
$NT_CNTRL, 754
$NT_CORR_ELEM_P, 754
$NT_CORR_ELEM_T, 754
$NT_NAME, 747
$NT_ROT_AX_NAME, 823
$NT_TRAFO_INDEX, 747
$P_ACTBFRAME, 666
$P_AD, 827
$P_AEP, 286
$P_APDV, 286
$P_APR, 286
$P_BFRAME, 666
$P_CHBFRAME, 666
$P_CHBFRMASK, 666
$P_CTOL, 898
$P_CTOL_ G0_ABS, 190
$P_CUT_INV, 828
$P_CUTMOD, 827
$P_CUTMOD_ANG, 827
$P_CUTMOD_ERR, 828
$P_CUTMODK, 827
$P_DELAYFST, 878
$P_F_TYPE, 146
$P_FGROUP_MASK, 116
$P_FZ, 146
$P_GWPS, 106
$P_IFRAME, 667
$P_IS_EES_PATH, 574
$P_NCBFRAME, 665
$P_NCBFRMASK, 666
$P_ORI_DIFF, 818

$P_ORI_POS, 818
$P_ORI_SOL, 819
$P_ORI_STAT, 822
$P_OTOL, 898
$P_OTOL_ G0_ABS, 190
$P_PATH, 573
$P_PFRAME, 667
$P_PROG, 573
$P_PROGPATH, 573
$P_S_TYPE, 100
$P_SIM, 639, 1022
$P_STACK, 573
$P_STOLF, 189
$P_SUBPAR, 507
$P_SVC, 100
$P_TOFF, 89
$P_TOFFCR, 89
$P_TOFFL, 89
$P_TOFFR, 89
$P_TOOLENV, 837
$P_TOOLENVN, 837
$P_WORKAREA_CS_COORD_SYSTEM, 361
$P_WORKAREA_CS_LIMIT_MINUS, 362
$P_WORKAREA_CS_LIMIT_PLUS, 361
$P_WORKAREA_CS_MINUS_ENABLE, 361
$P_WORKAREA_CS_PLUS_ENABLE, 361
$PA_ATOL, 898
$PA_FGREF, 115
$PA_FGROUP, 116
$SA_LEAD_TYPE, 947
$SC_CONTPREC, 871
$SC_MINFEED, 872
$SC_PA_ACTIV_IMMED, 595
$SN_PA_ACTIV_IMMED, 595
$TC_CARR_CORR_ELEM, 810
$TC_CARR1...14, 797
$TC_CARR13, 804
$TC_CARR14, 804
$TC_CARR18...23, 797
$TC_CARR18[m], 801
$TC_DP1 ... 25, 756
$TC_ECPxy, 760
$TC_SCPxy, 760
$TC_TP_MAX_VELO, 96

*****

* (fonction de calcul), 451

/

/ (fonction de calcul), 451

+

+ (fonction de calcul), 451

<

< (opérateur relationnel), 452
<< (opérateur de concaténation), 459
<= (opérateur relationnel), 452
<> (opérateur relationnel), 452

=

== (opérateur relationnel), 452

>

> (opérateur relationnel), 452
>= (opérateur relationnel), 452

A

A effet modal, 46
A effet non modal, 46
ABS, 451
AC, 153
ACC, 130
Accélération sur trajectoire
    Maximale, 893
ACCLIMA, 867
Accostage de butée, 368
ACN, 160
ACOS, 451
À-coup
    Correction, 890
    Limitation, 865
ACP, 160
ACTBLOCNO, 518
ACTFRAME, 645
Adapter à la charge - CYCLE782
    Programmation externe, 1129
ADIS, 301
ADISPOS, 301
ADISPOSA, 640

Adressage, 567
Adresse
  Affectation de valeurs, 48
Adresses, 386
Adresses OEM, 638
Affectation de valeurs, 48
Affichage de bloc
  Inhibition, 518
Alarmes
  Définir dans le programme CN, 1037
Alésage à l'alésoir - CYCLE85
  Programmation externe, 1112
Alésage au tour - CYCLE86
  Programmation externe, 1113
ALF
  Pour relèvement rapide du contour, 558
  Retrait rapide en cours de filetage, 235
AMIRROR, 334
AND, 452
ANG, 215
ANG1, 215
ANG2, 215
Angle de contour
  Arrondir, 250
  Chanfreinage, 250
Angle polaire, 30
Angles d'Euler, 687
Angles RPY, 687
Anticipation (LookAhead), 305
AP, 179
  Axes, 381
Appel de contour - CYCLE62
  Programmation externe, 1083
Appel de sous-programme avec chemin d'accès et paramètres, 545
APR, 420
APRB, 420
APRP, 420
APW, 420
APWB, 420
APWP, 420
AR
  Programmation d'un cercle, 198
ARD, 275
AROT, 322
AROTS, 328
Arrêt
  Assisté par CN, 1014
  Contrôlé par le variateur, 1015
  En fin de cycle, 353
  Facultatif, 353
  Programmé, 352
Arrêt facultatif, 353
Arrêt interne du prétraitement des blocs, 375
Arrêt précis, 299
Arrêt programmé, 352
Arrêt temporisé, 373
Arrondi, 250
Arrondir au chiffre supérieur, 456
AS, 498
ASCALE, 331
ASIN, 451
ASPLINE, 607
ATAN2, 451
ATOL, 894
ATRANS, 316
Attributs de position
  Programmation indirecte, 446
Autres conditions à prendre en compte lors des transformations, 734
AV, 955
Avance
  Avec correction par manivelle, 131
  Correction, 129, 133
  En sens inverse du temps, 111
  Pour axes à interpolation, 110
  pour axes de positionnement, 125
  Pour axes synchrones, 112
  Règles, 108
  Unités de mesure, 113
  Vitesse, 190
avec TCOFRX, 802
avec TCOFRY, 802
AX, 906
AXCTSWE, 912
AXCTSWEC, 912
AXCTSWED, 912
Axe
  Changement, 900
  Conteneur, 382
  Types, 376
Axe asservi
  Couplage de deux axes par valeur pilote, 943
Axe géométrique
  Basculer, 907
Axe pilote
  Couplage de deux axes par valeur pilote, 943
Axe transversal, 173
Axes
  AP, 381
  Axe Link-Lead, 383
  Canal, 379
  Commande, 381
  Conjugué, 924

Fraisage, 379
Géométriques, 376
Link, 381
Machine, 378
Moteur synchrone, 380
Positionnement, 379
Principaux, 376
Supplémentaires, 378
Axes conjugués, 924
Axes de positionnement, 379
Axes d'interpolation, 379
Axes d'orientation, 693
Axes géométriques permutables, 907
Axes rotatifs
Angle rotatif, 797
Vecteurs de direction, 797
Vecteurs distances, 797
Axes supplémentaires, 378
AXIS, 403
AXNAME, 458
AXSTRING, 906
AXTOCHAN, 904
AXTOSPI, 906

B

B_AND, 452
B_NOT, 452
B_OR, 452
B_XOR, 452
BAUTO, 607
BFRAME, 645
Bloc, 44
Fin, 47
Fin LF, 53
Longueur, 47
masquer, 49
numéro, 47
Ordre des instructions, 47
BLOC, 543
Bloc d'arrêt, 879
Bloc par bloc
Inhibition, 513
Blocs optionnels, 49
BLSYNC, 553
BNAT, 607
BNS, 39
BOOL, 403
Boucle de comptage, 489
Boucle de programme
Boucle, 488
Boucle de comptage, 489
Boucle IF, 486
Boucle REPEAT, 491
Boucle WHILE, 490
Boucle sans fin, 488
BOUND, 433
BRISK, 865
BRISKA, 865
Broche
Changement, 900
Fonctions M, 353
Limitation de la vitesse de rotation, 106
Positionnement, 121
Principale, 378
Sens de rotation, 91
Vitesse de rotation, 91
Broche maître, 378
Broche synchrone
Couplage, 954
Définir la paire, 961
BSPLINE, 607
BTAN, 607

C

CAC, 606
CACN, 606
CACP, 606
CADAPTOF, 916
CADAPTON, 916
Calcul de la distance, 385
CALL, 542
CALLPATH, 546
Canal
Axes, 379
Caractère 0, 457
Caractère spécial, 52, 53
CASE, 477
CDC, 606
CDOF, 290
CDOF2, 290
CDON, 290
Centrage - CYCLE81
Programmation externe, 1103
CFC, 135
CFIN, 135
CFINE, 654
CFTCP, 135
Chaîne de caractères
Concaténation, 459
Formater, 465
Longueur, 461
Opérations, 457

Chaîne outil, 753
Chaîne pièce, 753
CHAN, 403
CHANDATA, 574
Chanfrein, 250
Changement d'outil
avec numéro T, 65
CHAR, 403
Chariotage
Fonctions auxiliaires, 1046
Chariotage - CYCLE951
Programmation externe, 1161
Chariotage au contour - CYCLE95
Programmation externe, 1116
Chariotage / Chariotage reste / Plongée / Plongée reste / Plongée G+D / Plongée G+D reste - CYCLE952
Programmation externe, 1164
Chemin d'accès, 568
Chemin de recherche
Appel de sous-programme, 572
Chemin de recherche programmable, 546
Chemin de répertoire, 569
CHF, 250
CHKDNO, 794
CHR, 250
CIC, 606
Cinématique
Décomposée, 801
CIP, 202
CLEARM, 492
CLRINT, 555
COARSE, 955
COARSEA, 640
Code G
Programmation indirecte, 445
Coefficient du polynôme, 620
COLLPAIR, 742
Commande
Axes, 381
Commentaires, 48
COMPCAD, 618
COMPCURV, 618
COMPLETE, 574
COMPOF, 618
COMPON, 618
Composante de frame
FI, 650
MI, 650
RT, 650
SC, 650
TR, 650
COMPSURF, 618
Concaténation
de chaînes de caractères, 459
Concaténation de filetages - CYCLE98
Programmation externe, 1118
Constante, 390
Constante binaire, 391
Constante décimale, 390
Constante hexadécimale, 390
Constante INTEGER, 390
Constante REAL, 390
CONTDCON, 1052
Contour
Accostage/retrait, 266
Calculateur, 216
Code, 1052
Elément, 174
Préparation, 1046
Réaccostage, 880
Table de, 1046
Contournage, 301
CONTPRON, 1046
Contrôle
Structures, 485
Coordonnées
Cartésiennes, 28
Cylindriques, 180
Polaires, 30
Coordonnées cartésiennes, 28
Coordonnées cylindriques, 180
Coordonnées polaires, 30
Correction
Longueur d'outil, 68
Plan, 293
Rayon d'outil, 69
Correction d'outil
Mémoire de correcteurs, 756
Offset, 85
Système de coordonnées pour valeurs d'usure, 764
Correction du rayon d'outil
Aux angles saillants, 272
CUT2DF, 293
décélération aux angles, 639
Correction en ligne de la longueur d'outil, 811
Corrections d'outil
Additives, 758
CORROF, 342
CORRTC, 807
CORRTRAFO, 748
COS, 451
cote absolue, 30
Cote relative, 32

Cotes
Au diamètre, 166
Au rayon, 166
Axes rotatifs et broches, 160
Possibilités, 152
Cotes relatives, 32
COUPDEF, 955
COUPDEL, 955
Couplage
Générique, 965
Couplage de deux axes par valeur pilote, 943
Couplage de vitesse, 958
Couplage par valeur de consigne, 958
Couplage par valeur pilote
Couplage par valeur réelle et couplage par valeur de consigne, 947
Synchronisation de l'axe pilote et de l'axe asservi, 946
Couplage par valeur réelle, 958
Couple de blocage
Accostage de butée, 371
COUPOF, 955
COUPOFS, 955
COUPON, 955
COUPONC, 955
COUPRES, 955
CP, 723
CPBC, 967
CPDEF, 966
CPDEL, 966
CPFMOF, 969
CPFMON, 969
CPFMSON, 968
CPFPOS + CPOF, 969
CPFPOS + CPON, 967
CPFRS, 967
CPLA, 966
CPLCTID, 966
CPLDEF, 966
CPLDEL, 966
CPLDEN, 966
CPLINSC, 971
CPLINTR, 971
CPLNUM, 966
CPLOF, 966
CPLON, 966
CPLOUTSC, 971
CPLOUTTR, 971
CPLPOS, 967
CPLSETVAL, 967
CPMALARM, 972
CPMBRAKE, 972
CPMPRT, 971
CPMRESET, 970
CPMSTART, 971
CPMVDI, 972
CPOF, 966
CPON, 966
CPRECOF, 871
CPRECON, 871
CPROT, 592
CPROTDEF, 588
CPSETTYPE, 972
CPSYNCOP, 971
CPSYNCOP2, 971
CPSYNCOV, 972
CPSYNFIP, 971
CPSYNFIP2, 972
CPSYNFIV, 972
CR, 196
Critère de fin de déplacement
Programmable, 640
CROTS, 328
CSPLINE, 607
CT, 204
CTAB, 937
CTABDEF, 927
CTABDEL, 933
CTABEND, 927
CTABEXISTS, 933
CTABFNO, 942
CTABFPOL, 942
CTABFSEG, 942
CTABID, 936
CTABINV, 937
CTABISLOCK, 936
CTABLOCK, 934
CTABMEMTYP, 936
CTABMPOL, 942
CTABMSEG, 942
CTABNO, 942
CTABNOMEM, 942
CTABPERIOD, 936
CTABPOL, 942
CTABPOLID, 942
CTABSEG, 942
CTABSEGID, 942
CTABSEV, 937
CTABSSV, 937
CTABTEP, 937
CTABTEV, 937
CTABTMAX, 937
CTABTMIN, 937
CTABTSP, 937

CTABTSV, 937
CTABUNLOCK, 934
CTOL, 894
CTOLG0, 186
CTRANS, 654
CUT2D, 291
CUT2DD, 291
CUT2DF, 291
CUT2DFD, 291
CUT3DC, 772
CUT3DCC, 783
CUT3DCCD, 783
CUT3DCD, 772
CUT3DF, 777
CUT3DFD, 777
CUT3DFF, 777
CUT3DFS, 777
CUTCONOF, 294
CUTCONON, 294
CUTMOD, 824
CUTMODK, 824
CYCLE4071
    Programmation externe, 1170
CYCLE4072
    Programmation externe, 1172
CYCLE4073
    Programmation externe, 1176
CYCLE4074
    Programmation externe, 1177
CYCLE4075
    Programmation externe, 1180
CYCLE4077
    Programmation externe, 1183
CYCLE4078
    Programmation externe, 1187
CYCLE4079
    Programmation externe, 1189
CYCLE435 - Définir le système de coordonnées du dresseur
    Programmation externe, 1127
CYCLE495 - Profilage
    Programmation externe, 1127
CYCLE60 - Gravure
    Programmation externe, 1079
CYCLE61 - Surfaçage
    Programmation externe, 1081
CYCLE62 - Appel de contour
    Programmation externe, 1083
CYCLE63 - Fraisage de poche de contour / Matière restante du contour de poche / Fraisage du contour de tourillon / Matière restante du contour de tourillon
    Programmation externe, 1084
CYCLE64 - Perçage ébauche d'une poche de contour
    Programmation externe, 1087
CYCLE70 - Fraisage de filetage
    Programmation externe, 1088
CYCLE72 - Fraisage en contournage
    Programmation externe, 1090
CYCLE76 - Tourillon rectangulaire
    Programmation externe, 1094
CYCLE77 - Tourillon circulaire
    Programmation externe, 1096
CYCLE78 - Fraisage de filetages
    Programmation externe, 1098
CYCLE782 - Adapter à la charge
    Programmation externe, 1129
CYCLE79 - Polyèdre
    Programmation externe, 1101
CYCLE800 - Pivotement plan / Pivotement outil / Orientation outil
    Programmation externe, 1132
CYCLE801 - Modèle de positions Réseau ou cadre
    Programmation externe, 1136
CYCLE802 - Positions quelconques
    Programmation externe, 1137
CYCLE806 - Tournage par interpolation
    Programmation externe, 1141
CYCLE81 - Centrage
    Programmation externe, 1103
CYCLE82 - Perçage
    Programmation externe, 1104
CYCLE83 - Perçage profond 1
    Programmation externe, 1106
CYCLE830 - Perçage profond 2
    Programmation externe, 1141
CYCLE832 - Réglages High Speed
    Programmation externe, 1147
CYCLE84 - Taraudage sans porte-taraud compensateur
    Programmation externe, 1109
CYCLE840 - Taraudage avec porte-taraud compensateur
    Programmation externe, 1151
CYCLE85 - Alésage à l'alésoir
    Programmation externe, 1112
CYCLE86 - Alésage au tour
    Programmation externe, 1113
CYCLE899 - Rainure borgne
    Programmation externe, 1153
CYCLE92 - Tronçonnage
    Programmation externe, 1114
CYCLE930 - Gorge
    Programmation externe, 1156

CYCLE940 - Dégagement forme E et F / Dégagement filet
    Programmation externe, 1159
CYCLE95 - Chariotage au contour
    Programmation externe, 1116
CYCLE951 - Chariotage
    Programmation externe, 1161
CYCLE952 - Chariotage / Chariotage reste / Plongée / Plongée reste / Plongée G+D / Plongée G+D reste
    Programmation externe, 1164
CYCLE98 - Concaténation de filetages
    Programmation externe, 1118
CYCLE99 - Filetage au tour
    Programmation externe, 1122

D

D..., 81
D0, 81
DAC, 168
DC, 160
Début du bloc de programme - GROUP_BEGIN, 1192
Décalage de base, 40
Décalage d'origine
    Externe, 655
    Programmable, 316
    Réglable, 147
Décalage d'origine externe, 655
décalage du point d'attaque
    pour le filetage, 226
Décélération à tous les angles, 640
Décélération aux angles intérieurs, 640
DEF, 403
DEFAULT, 477
DEFINE ... AS, 498
Définir le système de coordonnées du dresseur - CYCLE435
    Programmation externe, 1127
Définition de la pièce brute, 1039
Dégagement forme E et F / Dégagement filet - CYCLE940
    Programmation externe, 1159
DELAYFSTOF, 875
DELAYFSTON, 875
DELDL, 760
DELETE, 581
DELOBJ, 736
DELTOOLENV, 835
Déplacement point à point, 722
Déplacement PTP cartésien, 722
Déplacements conjugués, 922
Descripteur, 43
Détection des violations du contour, 290
Développante, 210
DIACYCOFA, 168
DIAM90, 166
DIAM90A, 168
DIAMCHAN, 168
DIAMCHANA, 168
DIAMCYCOF, 166
DIAMOF, 166
DIAMOFA, 168
DIAMON, 166
DIAMONA, 168
DIC, 168
DILF, 235
DIN 66217, 36
DISABLE, 555
DISC, 272
DISCL, 275
DISPLOF, 518
DISPLON, 518
DISPR, 880
DISR, 275
DISRP, 275
DITE, 233
DITS, 233
DIV, 451
DL, 758
DO, 983
Données de cercles
    Calcul, 1058
DRFOF, 345
DRIVE, 865
DRIVEA, 865
DV, 955
DYNFINISH, 869
DYNNORM, 869
DYNPOS, 869
DYNPREC, 869
DYNROUGH, 869
DYNSEMIFIN, 869

E

Easy XML, 1029
EAUTO, 607
EES, 565
EG
    réducteur électronique, 948
EGDEF, 949
EGDEL, 953
EGOFC, 953

EGOFS, 953
EGON, 950
EGONSYN, 950
EGONSYNE, 950
Elément de contour
    Exécuter, 1057
ELSE, 486, 983
ENABLE, 555
ENAT, 607
ENDFOR, 489
ENDIF, 486
ENDLABEL, 479
ENDLOOP, 488
ENDWHILE, 490
Entrée principale, 171
ESR, 1009
ESRR, 1015
ESRS, 1015
ETAN, 607
Etat du couplage
    Couplage de deux axes par valeur pilote, 948
    Lors de déplacements conjugués, 925
Etiquette, 479
EVERY, 983
EXECSTRING, 448
EXECTAB, 1057
EXECUTE, 1059
Exécution
    Comportement de structures de contrôle, 485
EXP, 451
EXTCALL
    pour SINUMERIK 840D sl, 547
EXTCLOSE, 1033
Extension numérique, 387
EXTERN, 536
EXTOPEN, 1033

F

F...
    Filetage G34 G35, 232
    Interpolation linéaire, 190
    Pour avance, 108
FA, 125
facteur de couplage, 922
facteur d'échelle, 331
FAD, 275
FALSE, 403
FB, 140
FCTDEF, 768
FCUB, 860
FD, 131
FDA, 131
FENDNORM, 639
FFWOF, 870
FFWON, 870
FGREF, 108
FGROUP, 108
Fichier
    Informations, 585
FIFOCTRL, 873
FILEDATE, 585
FILEINFO, 585
FILESIZE, 585
FILESTAT, 585
Filetage
    Concaténation, 226
    G33, 225
    G34 G35, 232
    Multifilet, 226
    Pas, 232
    Sens de rotation, 227
Filetage à droite, 227
Filetage à gauche, 227
Filetage au tour - CYCLE99
    Programmation externe, 1122
Filetage plan, 230
Filetage sur corps conique, 231
Filetage sur corps cylindrique, 230
Filetages bombés, 239
FILETIME, 585
Fin du bloc de programme - GROUP_END
    Programmation externe, 1192
Fin du supplément de mise au point - GROUP_ADDEND
    Programmation externe, 1193
FINE, 955
FINEA, 640
FL, 108
FLIM, 891
FLIN, 860
FMA, 137
FNORM, 860
Fonction
    Prédéfinie, 1382
Fonctions M, 350
Fonctions OEM, 638
FOR, 489
Foret, 74
Format bande perforée, 44
FP, 364
FPO, 860
FPR, 125
FPRAOF, 125
FPRAON, 125

Fraisage de filetage - CYCLE70
Programmation externe, 1088
Fraisage de filetages - CYCLE78
Programmation externe, 1098
Fraisage de poche de contour / Matière restante du contour de poche / Fraisage du contour de tourillon / Matière restante du contour de tourillon – CYCLE63
Programmation externe, 1084
Fraisage en bout, 690
Fraisage en contournage - CYCLE72
Programmation externe, 1090
Fraises, 72
Frame
appel, 651
Concaténation, 669
désélection, 342
Fonction miroir, 334
Homothétie, 331
Instructions, 313
FRAME, 403
Frame neutre, 147
Frames, 311
à définition globale, 663
Attribution, 652
Concaténation de frames, 653
Spécifiques à un canal, 664
Système, 665
Frames système, 665
FRC, 250
FRCM, 250
FROM, 983
FTOCOF, 772
FTOCON, 772

G

G1, 190
G110, 177
G111, 177
G112, 177
G140, 275
G141, 275
G142, 275
G143, 275
G147, 275
G148, 275
G153
Décalage d'origine, 147
Désactiver pour frame, 342
G17, 149
G18, 149
G19, 149
G2, 193
G247, 275
G248, 275
G25
Limitation de la vitesse de rotation de broche, 106
Limitation de la zone de travail, 357
G26
Limitation de la vitesse de rotation de broche, 106
Limitation de la zone de travail, 357
G290, 1043
G291, 1043
G3, 193
G33, 225
G335, 239
G336, 239
G34, 232
G340, 275
G341, 275
G347, 275
G348, 275
G35, 232
G4, 373
G40, 256
G41, 256
G42, 256
G450, 272
G451, 272
G460, 286
G461, 286
G462, 286
G5, 719
G500
Décalage d'origine, 147
G505 ... G599, 147
G53
Décalage d'origine, 147
Désactiver pour frame, 342
G54 ... G57, 147
G58, 320
G59, 320
G60, 299
G601, 299
G602, 299
G603, 299
G62, 639
G621, 639
G64, 301
G641, 301
G642, 301
G643, 301
G644, 301

G645, 301
G7, 719
G70, 162
G700, 162
G71, 162
G710, 162
G74, 363
G75, 364
G810 ... G819, 638
G820 ... G829, 638
G9, 299
G90, 153
G91, 155
G93, 108
G94, 108
G95, 108
G96, 100
G961, 100
G962, 100
G97, 100
G971, 100
G972, 100
G973, 100
GEOAX, 907
Géométrie
    Axes, 376
GET, 900
GETACTTD, 796
GETD, 900
GETDNO, 795
GETTCOR, 837
GETTENV, 836
GETVARAP, 437
GETVARDFT, 439
GETVARDIM, 439
GETVARLIM, 438
GETVARPHU, 436
GETVARTYP, 441
GFRAME0 ... GFRAME100, 347
Gorge - CYCLE930
    Programmation externe, 1156
GOTO, 474
GOTOB, 474
GOTOC, 474
GOTOF, 474
GOTOS, 473
GP, 446
Gravure - CYCLE60
    Programmation externe, 1079
Grignotage, 997
GROUP_ADDEND - Fin du supplément de mise au point
    Programmation externe, 1193
GROUP_BEGIN - Début du bloc de programme
    Programmation externe, 1192
GROUP_END - Fin du bloc de programme
    Programmation externe, 1192
Groupe d'axes à déplacements conjugués, 922
Groupe G
    Technologie, 869
Groupe spline, 617
GUD, 404
GWPSOF, 106
GWPSON, 106

H

High Speed Settings - CYCLE832
    Programmation externe, 1147
HOLES1 - Modèle de positions Rangée
    Programmation externe, 1064
HOLES2 - Modèle de positions Cercle complet ou partiel
    Programmation externe, 1065

I

I...
    Filetage G33, 225
    Filetage G34 G35, 232
    Interpolation circulaire, 193
IC, 155
ID, 983
Identificateur
    pour chaîne de caractères, 53
    Valeurs numériques particulières, 53
    Variables propres au système, 53
IDS, 983
IF, 486
IFRAME, 645
INDEX, 462
Indication de cotes relatives, 155
Indice de tableau, 428
INICF, 403
INIPO, 403
INIRE, 403
INIT, 492
INITIAL, 574
INITIAL_INI, 574
Initialisation
    de tableaux, 426
insensible à la casse, 566
Instruction, 44
Instruction de déplacement, 174

Instructions G, 1337
INT, 403
Interpolation circulaire
avec un point intermédiaire et un point final, 202
Interpolation hélicoïdale, 208
Interpolation du vecteur de rotation, 703
Interpolation hélicoïdale, 208
Interpolation polynomiale, 619
INTERSEC, 1055
INVCCW, 210
INVCW, 210
IPOBRKA, 640
IPOENDA, 640
IPOSTOP, 955
IPTRLOCK, 878
IPTRUNLOCK, 878
IR, 239
ISAXIS, 906
ISFILE, 584
ISNUMBER, 458
ISOCALL, 544
ISVAR, 434

J

J...
Filetage G34 G35, 232
Interpolation circulaire, 193
JERKLIM, 890
JERKLIMA, 867
Jeu de caractères, 52
JR, 239

K

K...
Filetage G33, 225
Filetage G34 G35, 232
Interpolation circulaire, 193
KONT, 266
KONTC, 266
KONTT, 266
KR, 239

L

L..., 534
Langage évolué CN, 45
l'arrêt du prétraitement des blocs,
Niveau de signal, 375
LEAD, 684
LEADOF, 943
LEADON, 943
LENTOAX, 857
Lettres adresses, 1323
LF, 47
LFOF, 235
LFON, 235
LFPOS, 235
LFTXT, 235
LFWP, 235
LIFTFAST, 556
Limitation de la zone de travail
dans le SCB, 357
LIMS, 100
LINE FEED, 47
Linéaire
Interpolation, 190
Link
Axe Link-Lead, 383
Axes, 381
Variables, 402
Lissage
Du tracé d'orientation, 713
LLI, 416
LN, 451
LONGHOLE - Trou oblong
Programmation externe, 1077
LOOP, 488
LUD, 404

M

M..., 350
M0, 350
M1, 350
M17, 522
M19
Fonctions M, 350
Pour positionnement de broches, 121
M2, 350
M3, 91
M30, 522
M4, 91
M40, 350
M41, 350
M42, 350
M43, 350
M44, 350
M45, 350
M5, 91
M6, 66
M70, 121

Machines
Axes, 378
Macro, 498
Manivelle
Correction par, 131
Marque de saut
Répétitions de sections de programme, 479
Sauts de programme, 475
MASLDEF, 979
MASLDEL, 979
MASLOF, 979
MASLOFS, 979
MASLON, 979
MATCH, 462
MAXVAL, 433
MCALL, 540
MD10010, 492
MD10210, 807
MD10240, 164
MD10260, 162
MD10280, 492
MD10710, 235
MD24658, 844
MD9440, 83
MEAC, 629
MEAFRAME, 659
MEAS, 627
MEASA, 629
MEAW, 627
MEAWA, 629
Mémoire
Programme, 563
Tampon, 873
Travail, 574
Mémoire de correcteurs, 756
Mémoire de programmes
Répertoires standard, 564
Types de fichiers, 564
Mémoire de travail, 574
Mémoire globale de programmes pièce (GDIR), 565
Messages, 354
MINDEX, 462
MINVAL, 433
MIRROR, 334
MMC, 1028
MOD, 451
MODAXVAL, 906
Mode d'accélération, 865
Mode de langage, 1043
Modèle de positions Cercle complet ou partiel - HOLES2
Programmation externe, 1065
Modèle de positions Rangée - HOLES1
Programmation externe, 1064
Modèle de positions Réseau ou cadre - CYCLE801
Programmation externe, 1136
MPF, 564
MSG, 354

N

NAMETOINT, 740
NCK, 403
NEWCONF, 1024
Niveaux d'imbrication
de structures de contrôle, 485
Niveaux d'inhibition, 50
NOC, 955
Nom de fichier, 570
Nom de sous-programme DIN, 571
Nom du lecteur, 568
NORM, 266
NOT, 452
Notation à adresse étendue, 387
Notation EES, 567
Notation NCK, 567
NPROT, 592
NPROTDEF, 588
NUMBER, 458
Numéro D
attribuer librement, 794
Numéro de tranchant, 794
Numéros D
contrôler, 794
renommer, 795
NUT, 695

O

OEMIPO1/2, 638
OFFN, 256
Offset
Longueur d'outil, 85
Rayon d'outil, 85
OMA1 ... OMA5, 638
Opérateurs logiques, 452
Opérateurs relationnels, 452
OR, 452
Organe porte-outil
cinématique, 797
Orientable, 802
Point de référence, 35
Organes porte-outil orientables, 797

ORIAXES, 693
ORIC, 788
ORICONCCW, 695
ORICONCW, 695
ORICONIO, 695
ORICONTO, 695
ORICURVE, 698
ORID, 788
Orientation de l'outil
  Par rapport à la trajectoire, 704
Orientation de l'outil par rapport à la trajectoire, 704
ORIEULER, 693
Origine
  machine, 34
  pièce, 34
Origines
  Pour le tournage, 172
ORIMKS, 691
ORIPATH, 706
ORIPATHS, 706
ORIPLANE, 695
ORIRESET(A, B, C), 682
ORIROTA, 702
ORIROTC
  Interpolation de la rotation de l'outil, 707
  Rotation de l'orientation de l'outil, 702
ORIROTR, 702
ORIROTT, 702
ORIRPY, 693
ORIRPY2, 693
ORIS, 788
ORISOF, 713
ORISOLH, 813
ORISON, 713
ORIVECT, 693
ORIVIRT1, 693
ORIVIRT2, 693
ORIWKS, 691
OS, 984
OSB, 984
OSC, 788
OSCILL, 989
Oscillation
  Asynchrone, 984
  Oscillation asynchrone, 984
  Oscillation synchrone, 989
  Pénétration au point d'inversion de sens, 994
  Pénétration partielle, 992
  pilotée par des actions synchrones, 989
  Point d'inversion, 992
  Zone d'inversion, 992
Oscillation asynchrone, 984
Oscillation synchrone
  Actions synchrones, 993
  affectation des axes d'oscillation et de pénétration, 992
  définir les pénétrations, 992
  Exploitation Période d'appel de l'interpolateur, 995
  Mouvement de pénétration, 993
  Pénétration dans la zone d'inversion de sens, 993
  Pénétration partielle suivante, 995
OSCTRL, 984
OSD, 788
OSE, 984
OSNSC, 984
OSOF, 788
OSP1, 984
OSP2, 984
OSS, 788
OSSE, 788
OST, 788
OST1, 984
OST2, 984
OTOL, 894
OTOLG0, 186
Outil
  Changement avec M6, 66
  Correction de longueur, 68
  Correction de rayon, 69
  Groupe, 71
  Mémoire de corrections, 70
  Numéro de type, 71
  Orientation, 788
  Paramètre, 756
  Point de changement, 35
  Pointe, 70
  Tranchant, 81
  Type, 71
Outils de rectification, 75
Outils de tournage, 77
Outils spéciaux, 79
OVR, 129
OVRA, 129
OVRRAP, 129

P

P..., 538
P_ACTFRAME, 667
PACCLIM, 893
Paramètre
  Outil, 756

Paramètre de calcul
Spécifiques à un canal, 398
Paramètres
Effectifs, 505
Formels, 505
Transfert à l'appel d'un sous-programme, 505, 536
Paramètres de calcul
Globaux, 400
Paramètres de cycle de mesure
CYCLE961, 1215, 1257
CYCLE971, 1234, 1276
CYCLE973, 1195, 1238
CYCLE974, 1197, 1240
CYCLE976, 1204, 1246
CYCLE977, 1212, 1254
CYCLE978, 1206, 1248
CYCLE979, 1218, 1260
CYCLE982, 1231, 1273
CYCLE994, 1201, 1243
CYCLE995, 1224, 1266
CYCLE996, 1226, 1268
CYCLE9960, 1229, 1271
CYCLE997, 1221, 1263
CYCLE998, 1209, 1251
PAROT, 339
PAROTOF, 339
PCALL, 545
PDELAYOF, 997
PDELAYON, 997
Perçage - CYCLE82
Programmation externe, 1104
Perçage ébauche d'une poche de contour - CYCLE64
Programmation externe, 1087
Perçage profond 1 - CYCLE83
Programmation externe, 1106
Perçage profond 2 - CYCLE830
Programmation externe, 1141
PFRAME, 645
PHI
Orientation le long d'une enveloppe de cône, 695
Polynômes d'orientation, 701
PHU, 418
Pièce
Catalogue, 564
Compteurs, 1020
Contour, 175
Répertoires, 564
Pivotement plan / Pivotement outil / Orientation outil – CYCLE800
Programmation externe, 1132
PL
Interpolation de type spline, 607
Interpolation polynomiale, 619
Plan de travail, 33
PM, 275
PM10651, 240
PM15800, 401
PM18104, 834
PM18116, 835
PM18156, 401
PM20360, 842
PM24558, 844
PO, 619
PO[PHI]
Orientation le long d'une enveloppe de cône, 695
Polynômes d'orientation, 701
Rotation de l'orientation de l'outil, 706
PO[PSI]
Orientation le long d'une enveloppe de cône, 695
Polynômes d'orientation, 701
Rotation de l'orientation de l'outil, 706
PO[THT]
Polynômes d'orientation, 701
Rotation de l'orientation de l'outil, 706
PO[XH]
Orientation avec indication de deux points de contact, 698
Polynômes d'orientation, 701
PO[YH]
Orientation avec indication de deux points de contact, 698
Polynômes d'orientation, 701
PO[ZH]
Orientation avec indication de deux points de contact, 698
Polynômes d'orientation, 701
Poche circulaire - POCKET4
Programmation externe, 1069
Poche rectangulaire - POCKET3
Programmation externe, 1067
POCKET3 - Poche rectangulaire
Programmation externe, 1067
POCKET4 - Poche circulaire
Programmation externe, 1069
Poinçonnage, 997
Point / Angle d'accostage, 268
Point auxiliaire de la fraise, 776
Point de butée, 34
Point de départ, 35
Point de départ - Point de destination, 174
Point de destination, 174
Point de référence, 35

Point d'usinage de la fraise, 776
Point fixe
Accostage, 364
Pointe de la fraise, 776
Pointeur d'interruption automatique, 880
Points de référence, 34
Pôle, 177
POLF
Retrait assisté par CN, 1010
Retrait rapide en cours de filetage, 235
POLFA, 1010
POLFMASK
Retrait assisté par CN, 1010
Retrait rapide en cours de filetage, 235
POLFMLIN
Retrait assisté par CN, 1010
Retrait rapide en cours de filetage, 235
POLY, 619
Polyèdre - CYCLE79
Programmation externe, 1101
polynôme dénominateur, 623
POLYPATH, 619
PON, 1006
PONS, 997
POS, 116
POSA, 116
POSBA, 955
Position initiale de l'orientation de l'outil
ORIRESET, 683
Positions quelconques - CYCLE802
Programmation externe, 1137
Positions singulières, 692
POSP, 116
POT, 451
PR, 275
Précision de contour
Programmable, 871
Préparation du contour
Signalisation d'erreur, 1059
PREPRO, 521
PRESETON, 656
PRESETONS, 658
PRIO, 553
Prise de référence, 363
PRLOC, 403
Procédure
Prédéfinie, 1358
Process DataShare, 1033
Profilage - CYCLE495
Programmation externe, 1127
Profondeur de pénétration, 775
Programmation CN
Jeu de caractères, 52
Programmation de l'orientation, 693
Programmation d'éléments de contour, 215
Programmation diamétrale, 166
Programmation d'un cercle
Avec des coordonnées polaires, 200
avec un angle au centre et un centre, 198
avec un centre et un point final, 194
avec un rayon et un point final, 196
Modes d'interpolation, 193
Programmation externe, 1192
Programmation indirecte
Attributs de position, 446
d'adresses, 443
des codes G, 445
Lignes de programme pièce, 448
Programmation radiale, 166
Programme
Adressage, 567
En-tête, 53
Fin, 46
Initialisation, 575
Mémoire, 564
Nom, 43
Répétition, 538
Saut, 477
Sauts, 474
simulation, 1022
Temps d'exécution, 1017
Programme CN
Création, 51
Programme d'initialisation, 575
PROTA, 743
PROTD, 745
PROTS, 744
PSI
Orientation le long d'une enveloppe de cône, 695
Polynômes d'orientation, 701
PTP, 723
PTPG0, 723
PTPWOC, 723
PUD, 404
PUNCHACC, 997
PUTFTOC, 771
PUTFTOCF, 770
PW, 607

Q

QU, 349

R

RAC, 168
Rainure borgne - CYCLE899
    Programmation externe, 1153
Rainure rectiligne - SLOT1
    Programmation externe, 1072
Rainure sur cercle - SLOT2
    Programmation externe, 1074
Rayon
    Effectif, 114
Rayon de référence, 114
Rayon polaire, 30
READ, 582
REAL, 403
Rectification en plongée oblique, 719
REDEF, 409
réducteur électronique, 948
Règle des trois doigts, 36
RELEASE, 900
Relèvement rapide du contour, 556
REP, 426
REPEAT, 479
REPEATB, 479
Répétition de sections de programme
    avec programmation indirecte CALL, 543
REPOSA, 880
REPOSH, 880
REPOSHA, 880
REPOSL, 880
REPOSQ, 880
REPOSQA, 880
RET, 523
RET (paramétrable), 524
RETB (paramétrable), 530
Retrait
    Assisté par CN, 1010
    Contrôlé par le variateur, 1015
    direction pour le filetage, 236
Retrait rapide
    Filetage, 235
RG, 400
RIC, 168
RINDEX, 462
RMBBL, 880
RMEBL, 880
RMIBL, 880
RMNBL, 880
RND, 250
RNDM, 250
ROT, 322
Rotation
    Programmable, 322
    Vecteur d'orientation, 702
ROTS, 328
ROUND, 451
ROUNDUP, 456
Routine d'interruption
    Déplacement de retrait, 558
    mettre hors tension/sous tension, 555
    Nouvelle affectation, 554
    Relèvement rapide du contour, 556
    sens de déplacement programmable, 558
    Suppression, 555
RP, 179
RPL, 322
Run MyScreens, 1029

S

S, 91
Saisie et recherche de plages interdites à la recherche, 879
Saut
    au début du programme, 473
    Repères de saut, 474
SAVE, 512
SBLOF, 513
SBLON, 513
SCALE, 331
SCB, 38
SCC, 100
Scie à rainurer, 80
SCM, 36
SCP, 41
    orientation sur la pièce, 339
SCPARA, 915
SCR, 40
SD, 607
SD41610, 755
SD41611, 755
SD42010, 234
SD42440, 156
SD42442, 156
SD42465, 307
SD42466, 307
SD42475, 712
SD42476, 712
SD42477, 712
SD42900, 763
SD42910, 763
SD42920, 763
SD42930, 764

SD42935, 766
SD42940, 767
SD42984, 826
SD43240, 122
SD43250, 122
Section de programme
    Répétition, 479
Segmentation automatique des déplacements, 1002
SET, 426
SETAL, 1037
SETDNO, 795
SETINT, 553
SETM, 492
SETMS, 91
SETTCOR, 844
SF, 225
Simulation, 1022
Simulation de valeurs pilotes, 947
SIN, 451
SLOT1 - Rainure rectiligne
    Programmation externe, 1072
SLOT2 - Rainure sur cercle
    Programmation externe, 1074
SOFT, 865
SOFTA, 865
SON, 997
SONS, 997
Sortie
    Vers un périphérique/fichier externe, 1033
Sortie de fonction auxiliaire
    en contournage, 350
    Rapide, 349
Sortie des fonctions d'aide
    Propriétés des fonctions d'aide, 348
Sous-programme
    Appel avec transfert de paramètres, 536
    Appel indirect, 542
    Appel modal, 540
    Appel sans transfert de paramètres, 534
    Application, 501
    Chemin de recherche programmable, 546
    Nom, 502
    Répétition, 538
    Retour paramétrable (RET ...), 524
    Retour paramétrable (RETB ...), 530
SPCOF, 120
SPCON, 120
SPF, 564
SPI, 906
SPIF1, 997
SPIF2, 997
Spline
    interpolation, 607
    Types, 613
Spline A, 613
Spline B, 614
Spline C, 615
SPLINEPATH, 617
SPN, 1002
SPOF, 997
SPOS, 121
SPOSA, 121
SPP, 1002
SPRINT, 465
SQRT, 451
SR, 137
SRA, 137
ST, 137
STA, 137
START, 492
STARTFIFO, 873
STAT, 724
STOLF, 186
STOPFIFO, 873
STOPRE, 873
STRING, 403
STRINGIS, 1025
STRLEN, 461
SUBSTR, 463
SUPA
    Décalage d'origine, 147
    Désactiver pour frame, 342
Surfaçage - CYCLE61
    Programmation externe, 1081
SVC, 94
Synchrone
    Axes, 380
Synchronisme
    fin, 958
    grossier, 958
Synchronisme de position, 955
Synchronisme de position avec décalage angulaire, 955
Synchronisme de vitesse de rotation, 955
SYNR, 403
SYNRW, 403
SYNW, 403
Système de coordonnées
    de base, 38
    Vue d’ensemble, 36
Système de coordonnées de base, 38

Système de coordonnées décalé
de base, 39
Réglable, 40
Système de coordonnées décalé de base, 39
Système de coordonnées machine, 36
Système de coordonnées pièce, 41
Système de mesure, 162

T

T0, 65
Tableau, 426
Définition, 426
Elément, 426
Tampon
Mémoire, 873
TAN, 451
TANG, 973
TANGDEL, 978
TANGOF, 977
TANGON, 976
Taraudage avec porte-taraud compensateur - CYCLE840
Programmation externe, 1151
Taraudage sans porte-taraud compensateur - CYCLE84
Programmation externe, 1109
TCARR, 802
TCOABS, 802
TCOFR, 802
TCOFRZ, 802
Temps d'usinage, 1017
Temps restant
pour une pièce, 1019
THETA
Interpolation de la rotation de l'outil, 707
Rotation de l'orientation de l'outil, 702
TILT, 684
TLIFT, 975
TMOF, 1008
TMON, 1008
TOFF, 85
TOFFL, 85
TOFFLR, 85
TOFFOF, 811
TOFFON, 811
TOFFR, 85
TOFRAME, 339
TOFRAMEX, 339
TOFRAMEY, 339
TOFRAMEZ, 339
Tolérances G0, 186
TOLOWER, 460
TOOLENV, 832
TOROT, 339
TOROTOF, 339
TOROTX, 339
TOROTY, 339
TOROTZ, 339
TOUPPER, 460
Tourillon circulaire - CYCLE77
Programmation externe, 1096
Tourillon rectangulaire - CYCLE76
Programmation externe, 1094
Tournage par interpolation - CYCLE806
Programmation externe, 1141
TOWBCS, 764
TOWKCS, 764
TOWMCS, 764
TOWSTD, 764
TOWTCS, 764
TOWWCS, 764
TRAANG
avec angle programmable, 718
TRACON, 721
TRACYL, 715
TRAFOOF, 735
TRAFOON, 747
TRAILOF, 922
TRAILON, 922
Tranchant
Centre, 70
Numéro, 82
Point de référence, 296
Position, 70
Position définie, 296
Rayon, 70
Tranchants
Nombre de tranchants des outils de contour, 292
TRANS, 316
Transformation à angle oblique (TRAANG)
avec angle programmable, 718
Transformation avec axe linéaire pivotant, 680
Transformation de l'orientation TRAORI
Cinématique de machine, 673
Déplacements d'orientation, 672
Programmation d'orientation, 682
Transformation 5/6 axes générique, 673
Variantes de programmation d'orientation, 682
Transformation de surfaces cylindriques, 674
Transformation polaire, 674
Transformations
concaténées, 721
Transformation d'orientation, 670
Transformation trois, quatre ou cinq axes, 681

Transformations cinématiques, 671
Transformations concaténées, 672
Transformations des coordonnées (Frames), 41
TRANSMIT, 714
TRAORI, 681
Tronçonnage - CYCLE92
Programmation externe, 1114
Trou oblong - LONGHOLE
Programmation externe, 1077
TRUE, 403
TRUNC, 451
TU, 728
TURN, 208
Type de cinématique, 801
Type de couplage, 958

U

ULI, 416
UNTIL, 491

V

Valeur de réglage, 760
Valeur d'usure, 760
Valeur S
Interprétation, 93
Variable
conversion de type, 442
Variable frame
affecter des valeurs, 648
Appel de transformations de coordonnées, 643
Variable frame prédéfinie, 645
Variables
Conversion de type, 458
Définies par l'utilisateur, 403
VCM, 106
Vecteur de direction, 687
Vecteur d'orientation THETA, 702
VELOLIM, 888
VELOLIMA, 867
Vitesse de coupe, 94
Vitesse de coupe (constante), 100
Vitesse de rotation d'outil
Maximale, 96
Vitesse tangentielle
Maximale, 891

W

WAITC, 955
WAITE, 492
WAITENC, 914
WAITM, 492
WAITMC, 492
WAITP, 116
WAITS, 121
WALCS<n>, 360
WALCS0, 360
WALIMOF, 357
WALIMON, 357
WHEN, 983
WHEN-DO, 993
WHENEVER, 983
WHENEVER-DO, 993
WHILE, 490
WORKPIECE, 1039
WRITE, 578
WRTPR, 355

X

X..., 176
XML utilisateur, 1029
XOR, 452

Y

Y..., 176

Z

Z..., 176
Zones de protection, 588