SIEMENS

SINUMERIK

SINUMERIK 840D sl Programación CN

Manual de programación

Prólogo

Consignas básicas de seguridad 1

Fundamentos 2

Preparación del trabajo 3

Tablas 4

Anexo: A

Válido para

Control
SINUMERIK 840D sl / 840DE sl

Software de CNC versión 4.93

12/2019
6FC5398-2EP40-0EA0

Промэлектроника, sales@prom-elec.com

Notas jurídicas

Filosofía en la señalización de advertencias y peligros

Este manual contiene las informaciones necesarias para la seguridad personal así como para la prevención de daños materiales. Las informaciones para su seguridad personal están resaltadas con un triángulo de advertencia; las informaciones para evitar únicamente daños materiales no llevan dicho triángulo. De acuerdo al grado de peligro las consignas se representan, de mayor a menor peligro, como sigue.

⚠ PELIGRO
Significa que si no se adoptan las medidas preventivas adecuadas **se producirá** la muerte o bien lesiones corporales graves.

⚠ ADVERTENCIA
Significa que si no se adoptan las medidas preventivas adecuadas **puede producirse** la muerte o bien lesiones corporales graves.

⚠ PRECAUCIÓN
Significa que si no se adoptan las medidas preventivas adecuadas pueden producirse lesiones corporales.

ATENCIÓN
Significa que si no se adoptan las medidas preventivas adecuadas pueden producirse daños materiales.

Si se dan varios niveles de peligro se usa siempre la consigna de seguridad más estricta en cada caso. Si en una consigna de seguridad con triángulo de advertencia de alarma de posibles daños personales, la misma consigna puede contener también una advertencia sobre posibles daños materiales.

Personal cualificado

El producto/sistema tratado en esta documentación sólo deberá ser manejado o manipulado por **personal cualificado** para la tarea encomendada y observando lo indicado en la documentación correspondiente a la misma, particularmente las consignas de seguridad y advertencias en ella incluidas. Debido a su formación y experiencia, el personal cualificado está en condiciones de reconocer riesgos resultantes del manejo o manipulación de dichos productos/sistemas y de evitar posibles peligros.

Uso previsto de los productos de Siemens

Considere lo siguiente:

⚠ ADVERTENCIA
Los productos de Siemens sólo deberán usarse para los casos de aplicación previstos en el catálogo y la documentación técnica asociada. De usarse productos y componentes de terceros, éstos deberán haber sido recomendados u homologados por Siemens. El funcionamiento correcto y seguro de los productos exige que su transporte, almacenamiento, instalación, montaje, manejo y mantenimiento hayan sido realizados de forma correcta. Es preciso respetar las condiciones ambientales permitidas. También deberán seguirse las indicaciones y advertencias que figuran en la documentación asociada.

Marcas registradas

Todos los nombres marcados con ® son marcas registradas de Siemens AG. Los restantes nombres y designaciones contenidos en el presente documento pueden ser marcas registradas cuya utilización por terceros para sus propios fines puede violar los derechos de sus titulares.

Exención de responsabilidad

Hemos comprobado la concordancia del contenido de esta publicación con el hardware y el software descritos. Sin embargo, como es imposible excluir desviaciones, no podemos hacernos responsable de la plena concordancia. El contenido de esta publicación se revisa periódicamente; si es necesario, las posibles correcciones se incluyen en la siguiente edición.

Siemens AG
Digital Industries
Postfach 48 48
90026 NÜRNBERG
ALEMANIA

Referencia del documento: 6FC5398-2EP40-0EA0
℗ 12/2019 Sujeto a cambios sin previo aviso

Copyright © Siemens AG 2019.
Reservados todos los derechos

Prólogo

Documentación SINUMERIK

La documentación SINUMERIK se estructura en las siguientes categorías:

- Documentación general y catálogos
- Documentación para el usuario
- Documentación para el fabricante o servicio técnico

Información adicional

En la siguiente dirección (https://support.industry.siemens.com/cs/de/en/view/108464614) encontrará información relativa a los siguientes temas:

- pedir documentación/lista de publicaciones;
- otros enlaces para la descarga de documentos;
- uso online de documentación (manuales/búsqueda y exploración de información).

Para cualquier consulta con respecto a la documentación técnica (p. ej. sugerencias, correcciones), envíe un e-mail a la siguiente dirección (mailto:docu.motioncontrol@siemens.com).

mySupport/Documentación

En la siguiente dirección (https://support.industry.siemens.com/My/ww/es/documentation) encontrará información sobre cómo recopilar de manera personalizada documentación basada en los contenidos de Siemens y adaptarla a la documentación propia de la máquina.

Formación

La siguiente dirección (http://www.siemens.com/sitrain) contiene información sobre SITRAIN, el programa de capacitación y formación de Siemens en torno a los productos, sistemas y soluciones de accionamientos y automatización.

FAQ

Encontrará preguntas frecuentes en las páginas de Service&Support, en Product Support (https://support.industry.siemens.com/cs/de/en/ps/faq).

SINUMERIK

Encontrará información sobre SINUMERIK en la siguiente dirección (http://www.siemens.com/sinumerik).

Destinatarios

La presente documentación está destinada a:

- Programadores
- Proyectistas

Finalidad

El Manual de programación capacita a los destinatarios para diseñar, escribir y probar programas e interfaces de software y para resolver errores.

Alcance estándar

El presente manual de programación contiene una descripción de la funcionalidad estándar. Los suplementos o las modificaciones realizados por el fabricante de la máquina son documentadas por el mismo.

En el control pueden ejecutarse otras funciones adicionales no descritas en la presente documentación. Sin embargo, no se pueden reclamar por derecho estas funciones en nuevos suministros o en intervenciones de mantenimiento.

Asimismo, por razones de claridad expositiva, esta documentación no detalla toda la información relativa a las variantes completas del producto descrito ni tampoco puede considerar todos los casos imaginables de instalación, de explotación ni de mantenimiento.

Nota relativa al Reglamento general de protección de datos

Siemens respeta los principios básicos de la protección de datos, en especial los preceptos de la minimización de datos (privacy by design). Para este producto, esto significa:

El producto no procesa/almacena datos personales, únicamente datos técnicos asociados a sus funciones (p. ej. sellos de tiempo). Si el usuario enlaza estos datos con otros datos (p. ej. horarios de turnos) o almacena datos personales en el mismo medio (p. ej. el disco duro), creando de esta manera un vínculo con personas específicas, deberá cumplir él mismo las prescripciones legales relativas a la protección de datos.

Technical Support

Los números de teléfono específicos de cada país para el asesoramiento técnico se encuentran en Internet en la siguiente dirección (https://support.industry.siemens.com/sc/ww/es/sc/asistencia-tecnica/oid2090), en el apartado "Contacto".

Para formular una pregunta técnica, utilice el formulario online en el área "Support Request".

Información sobre la estructura y el contenido de la documentación

La documentación sobre la programación CN están divididas en dos capítulos principales:

1. **Fundamentos**
   El capítulo Fundamentos está destinado al operario de máquinas cualificado y presupone los correspondientes conocimientos en mecanizados de taladrado, fresado y torneado. Mediante ejemplos de programación sencillos se explican también los comandos e instrucciones conocidos según DIN 66025.
2. **Preparación del trabajo**
   El capítulo Preparación del trabajo está destinado al técnico ya familiarizado con todas las posibilidades de programación. El control SINUMERIK permite programar, con un lenguaje de programación especial, un programa de pieza complejo (p. ej., superficies de forma libre, coordinación de canales, etc.) y facilita la laboriosa programación del técnico.

Índice

Prólogo **3**

1 Consignas básicas de seguridad **23**

1.1 Consignas generales de seguridad 23

1.2 Garantía y responsabilidad para ejemplos de aplicación 24

1.3 Seguridad industrial 25

2 Fundamentos **27**

2.1 Fundamentos geométricos 27
2.1.1 Posiciones de pieza 27
2.1.1.1 Sistema de referencia de los datos de posición 27
2.1.1.2 Coordenadas cartesianas 27
2.1.1.3 Coordenadas polares 29
2.1.1.4 Acotado absoluto 30
2.1.1.5 Acotado incremental 32
2.1.2 Planos de trabajo 33
2.1.3 Orígenes y puntos de referencia 34
2.1.4 Sistemas de coordenadas 36
2.1.4.1 Sistema de coordenadas de máquina (MKS) 36
2.1.4.2 Sistema de coordenadas básico (BKS) 38
2.1.4.3 Sistema de origen básico (BNS) 39
2.1.4.4 Sistema de origen ajustable (ENS) 40
2.1.4.5 Sistema de coordenadas de pieza (WKS) 41
2.1.4.6 ¿Cómo se relacionan los diferentes sistemas de coordenadas? 42

2.2 Fundamentos de la programación CN 43
2.2.1 Denominación de un programa CN 43
2.2.2 Estructura y contenido de un programa CN 44
2.2.2.1 Secuencias y componentes de secuencia 44
2.2.2.2 Normas de secuencia 47
2.2.2.3 Asignaciones de valores 48
2.2.2.4 Comentarios 48
2.2.2.5 Omisión de secuencias 49

2.3 Creación de un programa CN 51
2.3.1 Procedimiento básico 51
2.3.2 Caracteres disponibles 52
2.3.3 Cabecera de programa 53
2.3.4 Ejemplos de programa 55
2.3.4.1 Ejemplo 1: primeros pasos de programación 55
2.3.4.2 Ejemplo 2: programa CN para torneado 55
2.3.4.3 Ejemplo 3: programa CN para fresado 57

2.4 Cambio de herramienta 60
2.4.1 Cambio de herramienta con gestión de herramientas activa 61
2.4.1.1 Cambio de herramienta con WZV activa con la función T=puesto 62
2.4.1.2 Cambio de herramienta con WZV activa con M6 64

2.4.2 Cambio de herramienta con gestión de herramientas no activa 65
2.4.2.1 Cambio de herramienta con WZV no activa con número T 65
2.4.2.2 Cambio de herramienta con WZV no activa con M6 66
2.4.3 Comportamiento en caso de programación T errónea 67

2.5 Correcciones de herramientas 68
2.5.1 Contorno y trayectoria de herramienta programados 68
2.5.2 Corrección longitudinal de herramienta 68
2.5.3 Corrección del radio de herramienta 69
2.5.4 Memoria de corrección de herramienta 70
2.5.5 Tipos de herramientas 71
2.5.5.1 Número de tipo de herramienta y grupos de herramientas 71
2.5.5.2 Fresas 72
2.5.5.3 Broca 74
2.5.5.4 Muelas rectificadoras 75
2.5.5.5 Cuchillas de tornear 77
2.5.5.6 Herramientas especiales 79
2.5.6 Activación/desactivación de correcciones de herramientas (D, D0) 81
2.5.7 Supresión de correcciones de herramienta (SUPD) 83
2.5.8 Offset programable de corrección de herramienta (TOFFL, TOFF, TOFFR, TOFFLR) 85

2.6 Movimiento de cabezales 91
2.6.1 Velocidad de giro del cabezal (S), sentido de giro del cabezal (M3/M4/M5) 91
2.6.2 Velocidad de corte de la herramienta (SVC) 94
2.6.3 Velocidad de corte constante (G96/G961/G962, G97/G971/G972, G973, LIMS, SCC) 101
2.6.4 Conexión/desconexión de la velocidad periférica de muela constante (GWPSON, GWPSOF) 106
2.6.5 Límite programable de la velocidad del cabezal (G25, G26) 107

2.7 Ajuste de avance 108
2.7.1 Avance (G93, G94, G95, F, FGROUP, FL, FGREF) 108
2.7.2 Desplazar ejes de posicionado (POS, POSA, POSP, FA, WAITP, WAITMC) 116
2.7.3 Servicio de cabezal regulado en posición (SPCON, SPCOF) 120
2.7.4 Posicionamiento de cabezales (SPOS, SPOSA, M19, M70, WAITS) 121
2.7.5 Avance para ejes de posicionado/cabezales (FA, FPR, FPRAON, FPRAOF) 126
2.7.6 Corrección programable del avance (OVR, OVRRAP, OVRA) 129
2.7.7 Corrección programable de la aceleración (ACC) 130
2.7.8 Avance con corrección del volante (FD, FDA) 131
2.7.9 Optimización del avance para contornos de curvatura pronunciada (CFTCP, CFC, CFIN) 135
2.7.10 Varios valores de avance en una secuencia (F, ST, SR, FMA, STA, SRA) 137
2.7.11 Avance por secuencia (FB) 140
2.7.12 Avance por diente (G95 FZ) 141

2.8 Ajustes de geometría 147
2.8.1 Decalaje de origen ajustable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153) 147
2.8.2 Decalaje de origen ajustable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153): Información adicional 148
2.8.3 Selección del plano de trabajo (G17/G18/G19) 149
2.8.4 Acotado 152
2.8.4.1 Acotado absoluto (G90, AC) 153
2.8.4.2 Acotado incremental (G91, IC) 155
2.8.4.3 Acotado absoluto e incremental en el torneado y el fresado (G90/G91) 159
2.8.4.4 Acotado absoluto para ejes giratorios (DC, ACP, ACN) 160
2.8.4.5 Sistema de medida métrico/imperial (G70/G71, G700/G710) 162

2.8.4.6 Programación por diámetro/radio específica del canal (DIAMON, DIAM90, DIAMOF, DIAMCYCOF) 166
2.8.4.7 Programación por diámetro/radio específica del eje (DIAMONA, DIAM90A, DIAMOFA, DIACYCOFA, DIAMCHANA, DIAMCHAN, DAC, DIC, RAC, RIC) 168
2.8.5 Posición de la pieza en el torneado 172

2.9 Instrucciones de desplazamiento 174
2.9.1 Información general sobre los comandos de desplazamiento 174
2.9.2 Comandos de desplazamiento con coordenadas cartesianas (G0, G1, G2, G3, X..., Y..., Z...) 176
2.9.3 Comandos de desplazamiento con coordenadas polares 177
2.9.3.1 Punto de referencia de las coordenadas polares (G110, G111, G112) 177
2.9.3.2 Comandos de desplazamiento con coordenadas polares (G0, G1, G2, G3, AP, RP) 179
2.9.4 Desplazamiento en rápido 183
2.9.4.1 Activar desplazamiento en rápido (G0) 183
2.9.4.2 Activar/desactivar la interpolación lineal para desplazamientos en rápido (RTLION, RTLIOF) 185
2.9.4.3 Adaptación de tolerancias para desplazamientos en rápido (STOLF, CTOLG0 y OTOLG0) 187
2.9.5 Interpolación lineal (G1) 191
2.9.6 Interpolación circular 194
2.9.6.1 Vista general 194
2.9.6.2 Interpolación circular con centro y punto final (G2/G3, X... Y... Z..., I... J... K...) 195
2.9.6.3 Interpolación circular con radio y punto final (G2/G3, X... Y... Z..., CR) 197
2.9.6.4 Interpolación circular con ángulo en el vértice y punto final / centro (G2/G3, X... Y... Z... / I... J... K..., AR) 199
2.9.6.5 Interpolación circular con coordenadas polares (G2/G3, AP, RP) 201
2.9.6.6 Interpolación circular con punto intermedio y punto final (CIP, X... Y... Z..., I1... J1... K1...) 203
2.9.6.7 Interpolación circular con transición tangencial (CT, X... Y... Z...) 205
2.9.7 Interpolación helicoidal (G2/G3, TURN) 209
2.9.8 Interpolación de evolutas (INVCW, INVCCW) 211
2.9.9 Sucesión de elementos de contorno 216
2.9.9.1 Programación de la sucesión de elementos de contorno 216
2.9.9.2 Sucesiones de elementos de contorno: una recta 217
2.9.9.3 Sucesiones de elementos de contorno: dos rectas 219
2.9.9.4 Sucesiones de elementos de contorno: tres rectas 222
2.9.9.5 Sucesiones de elementos de contorno: Programación de punto final con ángulo 226
2.9.10 Tallado de roscas 226
2.9.10.1 Tallado de roscas con paso constante (G33, SF) 226
2.9.10.2 Tallado de roscas con paso creciente o decreciente (G34, G35) 233
2.9.10.3 Trayectos de entrada y salida programados en G33, G34 y G35 (DITS, DITE) 234
2.9.10.4 Retirada rápida durante el tallado de roscas (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN) 236
2.9.10.5 Roscas convexas (G335, G336) 240
2.9.11 Roscado con macho sin mandril de compensación 246
2.9.11.1 Roscado con macho sin mandril de compensación y movimiento de retirada (G331, G332) 246
2.9.11.2 Ejemplo: Roscado con macho con G331/G332 247
2.9.11.3 Ejemplo: emitir la velocidad de taladrado programada en el escalón de reducción actual 248
2.9.11.4 Ejemplo: aplicación del segundo juego de datos de escalón de reducción 248
2.9.11.5 Ejemplo: sin programación de velocidad de giro, vigilancia del escalón de reducción 249
2.9.11.6 Ejemplo: cambio del escalón de reducción no posible, vigilancia del escalón de reducción 249

2.9.11.7 Ejemplo: programación sin SPOS....249
2.9.12 Roscado con macho con mandril de compensación....250
2.9.12.1 Roscado con macho con mandril de compensación y movimiento de retirada (G63)....250
2.9.13 Chaflán, redondeo (CHF, CHR, RND, RNDM, FRC, FRCM)....251

2.10 Correcciones del radio de herramienta....257
2.10.1 Corrección del radio de herramienta (G40, G41, G42, OFFN)....257
2.10.2 Aproximación y retirada del contorno (NORM, KONT, KONTC, KONTT)....266
2.10.3 Corrección en las esquinas exteriores (G450, G451, DISC)....273
2.10.4 Aproximación y retirada suaves....276
2.10.4.1 Aproximación y retirada (G140 a G143, G147, G148, G247, G248, G347, G348, G340, G341, DISR, DISCL, DISRP, FAD, PM, PR)....276
2.10.4.2 Aproximación y retirada con estrategias de retirada ampliada (G460, G461, G462)....288
2.10.5 Activación/desactivación de la vigilancia de colisión ("detección de cuellos de botella") (CDON, CDOF, CDOF2)....292
2.10.6 Corrección de herramienta en 2 1/2 D (CUT2D, CUT2DD, CUT2DF, CUT2DFD)....294
2.10.7 Mantener constante la corrección del radio de herramienta (CUTCONON, CUTCONOF)....297
2.10.8 Herramientas con posición fija de filo....299

2.11 Influencia sobre la trayectoria....302
2.11.1 Parada precisa (G60, G9, G601, G602, G603)....302
2.11.2 Modo de contorneado (G64, G641, G642, G643, G644, G645, ADIS, ADISPOS)....304

2.12 Transformaciones de coordenadas (frames)....314
2.12.1 Frames....314
2.12.2 Instrucciones frame....316
2.12.3 Decalaje de origen programable (TRANS, ATRANS)....319
2.12.4 Decalaje de origen programable (G58, G59)....323
2.12.5 Rotación programable (ROT, AROT, RPL)....325
2.12.6 Rotaciones de frames progr. con áng. espaciales (ROTS, AROTS, CROTS)....331
2.12.7 Factor de escala programable (SCALE, ASCALE)....334
2.12.8 Simetría programable (MIRROR, AMIRROR)....337
2.12.9 Creación de frame por orientación de herramienta (TOFRAME, TOROT, PAROT)....342
2.12.10 Cancelar frame (G53, G153, SUPA, G500)....345
2.12.11 Programación: Cancelar superposiciones específicas de eje (CORROF)....345
2.12.12 Cancelación de decalajes de origen aditivos (DRFOF)....348
2.12.13 Decalajes de origen específicos de rectificado (GFRAME0, GFRAME1 ... GFRAME100)....350

2.13 Emisión de funciones auxiliares....351
2.13.1 Funciones M....354

2.14 Comandos complementarios....358
2.14.1 Salida de avisos (MSG)....358
2.14.2 Escribir cadena de caracteres en variable BTSS (WRTPR)....359
2.14.3 Limitación del campo de trabajo....361
2.14.3.1 Limitación del campo de trabajo en BKS (G25/G26, WALIMON, WALIMOF)....361
2.14.3.2 Limitación del campo de trabajo en WKS/ENS (WALCS0 ... WALCS10)....364
2.14.4 Búsqueda del punto de referencia (G74)....367
2.14.5 Desplazamiento a punto fijo (G75)....368
2.14.6 Desplazamiento a tope fijo (FXS, FXST, FXSW)....372
2.14.7 Tiempo de espera (G4)....377
2.14.8 Parada de decodificación interna....379

2.15 Otra información....380
2.15.1 Ejes....380

2.15.1.1 Ejes principales/Ejes geométricos 380
2.15.1.2 Ejes adicionales 382
2.15.1.3 Cabezal, cabezal maestro....382
2.15.1.4 Ejes de máquina 382
2.15.1.5 Ejes de canal....383
2.15.1.6 Ejes de contorneado 383
2.15.1.7 Ejes de posicionado 383
2.15.1.8 Ejes síncronos....384
2.15.1.9 Ejes de comando 385
2.15.1.10 Ejes PLC 385
2.15.1.11 Ejes lincados 385
2.15.1.12 Ejes lincados guía 387
2.15.2 Del comando de desplazamiento al movimiento de la máquina 389
2.15.3 Cálculo del recorrido 389
2.15.4 Direcciones 390
2.15.5 Nombres....392
2.15.6 Constantes 394
2.15.7 Operadores y funciones de cálculo 395

3 Preparación del trabajo 399

3.1 Programación flexible de CN 399
3.1.1 Variables 399
3.1.1.1 Datos de sistema 400
3.1.1.2 Variables de usuario predefinidas: Parámetros de cálculo específicos de canal (R) 402
3.1.1.3 Variables de usuario predefinidas: Parámetros de cálculo globales (RG)....404
3.1.1.4 Variables de usuario predefinidas: Variables de lincado 405
3.1.1.5 Definición de variables de usuario (DEF)....407
3.1.1.6 Redefinición de datos de sistema, datos de usuario e instrucciones de programación del CN (REDEF) 413
3.1.1.7 Atributo: Valor de inicialización 416
3.1.1.8 Atributo: Valores límite (LLI, ULI) 419
3.1.1.9 Atributo: Unidad física (PHU) 421
3.1.1.10 Atributo: derechos de acceso (APR, APW, APRP, APWP, APRB, APWB)....423
3.1.1.11 Vista general de atributos definibles y redefinibles 428
3.1.1.12 Definición e inicialización de variables de matriz (DEF, SET, REP) 430
3.1.1.13 Tipos de datos....436
3.1.1.14 Mínimo, máximo y rango de variables (MINVAL, MAXVAL, BOUND) 437
3.1.1.15 Comprobar la presencia de una variable (ISVAR) 438
3.1.1.16 Leer valores de atributo/tipo de datos (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP) 440
3.1.1.17 Conversiones de tipos posibles 445
3.1.2 Programación indirecta 446
3.1.2.1 Programación indirecta de direcciones 446
3.1.2.2 Programación indirecta de comandos G 448
3.1.2.3 Programación indirecta de atributos de posición (GP)....450
3.1.2.4 Programación indirecta de líneas de programa de pieza (EXECSTRING)....452
3.1.3 Operaciones....453
3.1.3.1 Funciones de cálculo 453
3.1.3.2 Operaciones de comparación y operaciones lógicas....455
3.1.3.3 Prioridad de las operaciones....457
3.1.3.4 Corrección de precisión en caso de errores de comparación (TRUNC)....458
3.1.3.5 Redondeo (ROUNDUP) 460

3.1.4 Operaciones con cadenas de caracteres 461
3.1.4.1 Conversión de tipos a STRING (AXSTRING) 461
3.1.4.2 Conversión de tipos de STRING (NUMBER, ISNUMBER, AXNAME) 462
3.1.4.3 Concatenación de cadenas (<<) 463
3.1.4.4 Conversión minúsculas/mayúsculas (TOLOWER, TOUPPER) 464
3.1.4.5 Definir la longitud de una cadena (STRLEN) 465
3.1.4.6 Búsqueda de caracteres/cadena en una cadena (INDEX, RINDEX, MINDEX, MATCH) 466
3.1.4.7 Selección de una cadena parcial (SUBSTR) 467
3.1.4.8 Lectura y escritura de caracteres individuales 468
3.1.4.9 Formatear cadena (SPRINT) 469
3.1.5 saltos y derivaciones del programa 477
3.1.5.1 Salto atrás hasta el inicio del programa (GOTOS) 477
3.1.5.2 Saltos de programa a marcas de salto (GOTOB, GOTOF, GOTO, GOTOC) 478
3.1.5.3 Bifurcación de programa (CASE ... OF ... DEFAULT ...) 481
3.1.6 Repetición de secciones de programa (REPEAT, REPEATB, ENDLABEL, P) 483
3.1.7 Estructuras de control 488
3.1.7.1 Instrucción condicional y bifurcación (IF, ELSE, ENDIF) 490
3.1.7.2 Bucle de programa sin fin (LOOP, ENDLOOP) 492
3.1.7.3 Bucle contador (FOR ... TO ..., ENDFOR) 493
3.1.7.4 Bucle de programa con la condición en el inicio del bucle (WHILE, ENDWHILE) 494
3.1.7.5 Bucle de programa con la condición al final del bucle (REPEAT, UNTIL) 495
3.1.7.6 Ejemplo de programa con estructuras de control imbricadas 495
3.1.8 Coordinación de programa en todos los canales (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM) 496
3.1.9 Técnica de macros (DEFINE ... AS) 502

3.2 Uso de subprogramas 505
3.2.1 Generalidades 505
3.2.1.1 Subprograma 505
3.2.1.2 Nombres de subprogramas 506
3.2.1.3 Imbricación de subprogramas 507
3.2.1.4 Ruta de búsqueda 508
3.2.1.5 Parámetros formales y actuales 508
3.2.1.6 Transferencia de parámetros 509
3.2.2 Definición de un subprograma 511
3.2.2.1 Subprograma sin transferencia de parámetros 511
3.2.2.2 Subprograma con transferencia de parámetros Call-By-Value (PROC) 512
3.2.2.3 Subprograma con transferencia de parámetros Call-By-Reference (PROC, VAR) 513
3.2.2.4 Almacenamiento de las funciones G modales (SAVE) 515
3.2.2.5 Suprimir la ejecución de secuencia individual (SBLOF, SBLON) 516
3.2.2.6 Suprimir la visualización de secuencia actual (DISPLOF, DISPLON, ACTBLOCNO) 521
3.2.2.7 Identificación de subprogramas con preparación (PREPRO) 524
3.2.2.8 Salto atrás al subprograma M17 525
3.2.2.9 Salto atrás al subprograma RET 526
3.2.2.10 Salto atrás parametrizable al subprograma (RET ...) 527
3.2.2.11 Salto atrás parametrizable al subprograma (RETB ...) 533
3.2.3 Llamada de un subprograma 537
3.2.3.1 Llamada de subprograma sin transferencia de parámetros 537
3.2.3.2 Llamada de subprograma con transferencia de parámetros (EXTERN) 539
3.2.3.3 Cantidad de repeticiones de programa (P) 541
3.2.3.4 Llamada de subprograma modal (MCALL) 543
3.2.3.5 Llamada de subprograma indirecta (CALL) 545

3.2.3.6 Llamada de programa indirecta con indicación de la sección de programa que se va a ejecutar (CALL BLOCK ... TO ...) 546
3.2.3.7 Llamada indirecta de un programa programado en lenguaje ISO (ISOCALL) 547
3.2.3.8 Llamada a subprograma con ruta y parámetros (PCALL) 548
3.2.3.9 Ampliación de la ruta de búsqueda en la llamada de subprograma (CALLPATH) 549
3.2.3.10 Ejecutar subprograma externo (EXTCALL) 550

3.3 Rutina de interrupción (ASUP) 555
3.3.1 Función de una rutina de interrupción 555
3.3.2 Creación de una rutina de interrupción 556
3.3.3 Asignar e iniciar una rutina de interrupción (SETINT, PRIO, BLSYNC) 556
3.3.4 Desactivación/reactivación de la asignación de una rutina de interrupción (DISABLE, ENABLE) 559
3.3.5 Borrado de la asignación de una rutina de interrupción (CLRINT) 559
3.3.6 Retirada rápida del contorno (SETINT LIFTFAST, ALF) 560
3.3.7 Sentido de desplazamiento en la retirada rápida del contorno 562
3.3.8 Desarrollo del movimiento en rutinas de interrupción 565

3.4 Administración de programas y ficheros 567
3.4.1 Memoria de programas 567
3.4.1.1 Memoria de programas en el NCK 567
3.4.1.2 Memoria de programas externa 569
3.4.1.3 Direccionamiento de ficheros de la memoria de programas 570
3.4.1.4 Ruta de búsqueda en la llamada de subprogramas 575
3.4.1.5 Consulta de ruta y nombre de fichero 577
3.4.2 Memoria de trabajo (CHANDATA, COMPLETE, INITIAL) 578

3.5 Manejo de ficheros 582
3.5.1 Escribir fichero (WRITE) 582
3.5.2 Borrar fichero (DELETE) 585
3.5.3 Leer líneas en fichero (READ) 586
3.5.4 Comprobar la presencia de un fichero (ISFILE) 588
3.5.5 Leer información de fichero (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) 589

3.6 Zonas protegidas 592
3.6.1 Definición de las zonas protegidas (CPROTDEF, NPROTDEF) 592
3.6.2 Activar/desactivar zonas protegidas (CPROT, NPROT) 595
3.6.3 Comprobación de vulneración de zonas protegidas, limitación del campo de trabajo y finales de carrera de software (CALCPOSI) 599

3.7 Órdenes de desplazamiento especiales 609
3.7.1 Desplazamiento a posiciones codificadas (CAC, CIC, CDC, CACP, CACN) 609
3.7.2 Interpolación spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL) 610
3.7.3 Conjunto spline (SPLINEPATH) 620
3.7.4 Activar/desactivar compresión de secuencias CN (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF) 621
3.7.5 Interpolación polinómica (POLY, POLYPATH, PO, PL) 622
3.7.6 Referencia de trayectoria ajustable (SPATH, UPATH) 627
3.7.7 Medición específica de canal (MEAS, MEAW) 629
3.7.8 Medición específica de eje (MEASA, MEAWA, MEAC) (opcional) 632
3.7.9 Funciones especiales para el usuario de OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829) 641
3.7.10 Reducción del avance con deceleración en los dos vértices (FENDNORM, G62, G621) 642

3.7.11 Criterio de fin del movimiento programable (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA) ... 643
3.8 Transformaciones de coordenadas (frames) ... 646
3.8.1 Transformada de coordenadas a través de variables frame ... 646
3.8.1.1 Variable frame predefinida ($P_CHBFRAME, $P_IFRAME, $P_PFRAME, $P_ACTFRAME) ... 648
3.8.2 Asignación de valores a frames ... 651
3.8.2.1 Asignar valores directos (valor de eje, ángulo, escala) ... 651
3.8.2.2 Lectura y modificación de componentes de frame (TR, FI, RT, SC, MI) ... 653
3.8.2.3 Calcular con frames ... 654
3.8.2.4 Definición de variables frame (DEF FRAME) ... 656
3.8.3 Decalaje basto y fino (CTRANS, CFINE) ... 657
3.8.4 Decalaje de origen externo ($AA_ETRANS) ... 658
3.8.5 Ajuste de valor real con pérdida del estado de referenciado (PRESETON) ... 659
3.8.6 Ajuste de valor real sin pérdida del estado de referenciado (PRESETONS) ... 661
3.8.7 Cálculo del frame a partir de 3 puntos de medida en el espacio (MEAFRAME) ... 662
3.8.8 Frames globales ... 666
3.8.8.1 Frames específicos de canal ($P_CHBFR, $P_UBFR) ... 667
3.8.8.2 Frames activos en el canal ... 668
3.9 Transformadas ... 673
3.9.1 Programación general de los tipos de transformada ... 673
3.9.1.1 Movimientos de orientación en las transformadas ... 675
3.9.1.2 Resumen de la transformada de orientación TRAORI ... 679
3.9.2 Transformada de tres, cuatro y cinco ejes (TRAORI) ... 681
3.9.2.1 Relaciones generales, cabezal de herramienta cardánico ... 681
3.9.2.2 Transformada de tres, cuatro y cinco ejes (TRAORI) ... 684
3.9.2.3 Variantes de la programación de la orientación y el estado inicial (ORIRESET) ... 685
3.9.2.4 Programación de la orientación de herramienta (A..., B..., C..., LEAD, TILT) ... 687
3.9.2.5 Fresado frontal (A4, B4, C4, A5, B5, C5) ... 693
3.9.2.6 Referencia de los ejes de orientación (ORIWKS, ORIMKS) ... 694
3.9.2.7 Programación de los ejes de orientación (ORIAXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2) ... 696
3.9.2.8 Programación de la orientación a lo largo de una superficie de cono (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) ... 698
3.9.2.9 Especificación de orientación de dos puntos de contacto (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=) ... 701
3.9.3 Polinomios de orientación (PO[ángulo], PO[coordenada]) ... 703
3.9.4 Giros de la orientación de la herramienta (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) ... 705
3.9.5 Orientaciones relativas a la trayectoria ... 707
3.9.5.1 Tipos de orientación relativos a la trayectoria ... 707
3.9.5.2 Giro relativo a la trayectoria de la orientación de herramienta (ORIPATH, ORIPATHS, ángulo de giro) ... 709
3.9.5.3 Interpolación relativa a la trayectoria del giro de herramienta (ORIROTC, THETA) ... 710
3.9.5.4 Alisamiento del desarrollo de la orientación (ORIPATHS A8=, B8=, C8=) ... 712
3.9.6 Compresión de la orientación (COMPON, COMPCURV, COMPCAD, COMPSURF) ... 713
3.9.7 Compresión de la orientación (COMPCAD, COMPSURF) ... 716
3.9.8 Activar/desactivar alisado del desarrollo de la orientación (ORISON, ORISOF) ... 719
3.9.9 Transformación cinemática ... 720
3.9.9.1 Conexión de la transformada de lados frontales (TRANSMIT) ... 720
3.9.9.2 Conexión de la transformada de envolvente de cilindro (TRACYL) ... 720

3.9.9.3 Activar transformada de ángulo inclinado con ángulo programable (TRAANG) 723
3.9.9.4 Ranurado oblicuo en rectificadoras (G5, G7) 724
3.9.10 Conexión de la transformada concatenada (TRACON) 726
3.9.11 Desplazamiento PTP cartesiano 727
3.9.11.1 Conexión/desconexión del desplazamiento cartesiano PTP (PTP, PTPG0, PTPWOC, CP) 727
3.9.11.2 Indicación de la posición de las articulaciones (STAT) 729
3.9.11.3 Indicación de los signos de los ángulos de eje (TU) 733
3.9.11.4 Ejemplo 1: desplazamiento PTP de un robot de 6 ejes con transformada ROBX 736
3.9.11.5 Ejemplo 2: Desplazamiento PTP en transformada genérica de 5 ejes 737
3.9.11.6 Ejemplo 3: PTPG0 y TRANSMIT 737
3.9.12 Limitaciones en la selección de una transformada 739
3.9.13 Cancelar transformada (TRAFOOF) 740

3.10 Cadenas cinemáticas 741
3.10.1 Eliminación de componentes (DELOBJ) 741
3.10.2 Determinación del índice a partir del nombre (NAMETOINT) 744

3.11 Prevención de colisiones con cadenas cinemáticas 746
3.11.1 Comprobación de par de colisión (COLLPAIR) 746
3.11.2 Solicitar nuevo cálculo del modelo de máquina de la prevención de colisiones (PROTA) 747
3.11.3 Asignar estado a la zona protegida (PROTS) 748
3.11.4 Determinación de la distancia de dos zonas protegidas (PROTD) 749

3.12 Transformadas con cadenas cinemáticas 751
3.12.1 Activar transformada (TRAFOOF) 751
3.12.2 Modificar la transformada de orientación después de la medición de la máquina (CORRTRAFO) 752

3.13 Correcciones de herramientas 760
3.13.1 Memoria de corrección 760
3.13.2 Correcciones aditivas 762
3.13.2.1 Seleccionar correcciones aditivas (DL) 762
3.13.2.2 Definir valores de desgaste y de ajuste ($TC_SCPxy[t,d], $TC_ECPxy[t,d]) 764
3.13.2.3 Borrar correcciones aditivas (DELDL) 764
3.13.3 Acciones especiales de corrección de herramienta 765
3.13.3.1 Simetrizar longitudes de herramienta 767
3.13.3.2 Evaluación de signos de desgaste 767
3.13.3.3 Sistema de coordenadas del mecanizado activo (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS) 768
3.13.3.4 Longitud de herramienta y cambio de plano 771
3.13.4 Corrección de herramienta online 772
3.13.4.1 Definición de la función polinómica (FCTDEF) 772
3.13.4.2 Escritura continua de corrección de herramienta online (PUTFTOCF) 774
3.13.4.3 Escritura discreta de corrección de herramienta online (PUTFTOC) 775
3.13.4.4 Activación/desactivación de la corrección de herramienta online (FTOCON/FTOCOF) 776
3.13.5 Corrección de radio de herramienta 3D 776
3.13.5.1 Seleccionar la corrección de radio de herramienta 3D para fresado periférico 3D (CUT3DC, CUT3DCD, ISD) 776
3.13.5.2 Seleccionar la corrección de radio de herramienta 3D para el fresado frontal 3D (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD) 781
3.13.5.3 Fresado periférico 3D considerando una superficie de limitación (CUT3DCC, CUT3DCCD) 787
3.13.6 Orientación de herramienta (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST) 792

3.13.7 Asignación libre de números D, número de filo 798
3.13.7.1 Asignación libre de números D, número de filo (dirección CE)....798
3.13.7.2 Asignación libre de números D: Comprobar números D (CHKDNO)....798
3.13.7.3 Asignación libre de números D: Cambiar nombre de números D (GETDNO, SETDNO) 799
3.13.7.4 Asignación libre de números D: Determinar el número T para el número D predeterminado (GETACTTD) 800
3.13.7.5 Asignación libre de números D: Fijación como no válidos de los números D (DZERO)....800
3.13.8 Cinemática del portaherramientas 801
3.13.9 Corrección longitudinal de herramienta para portaherramientas orientables (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)....807
3.13.10 Modificar los portaherramientas orientables después de la medición de la máquina (CORRTC) 810
3.13.11 Corrección longitudinal de herramienta online (TOFFON, TOFFOF) 814
3.13.12 Modificación de los datos de corrección en herramientas giratorias 816
3.13.12.1 Cálculo de orientaciones (ORISOLH) 816
3.13.12.2 Activar la modificación de los datos de corrección en herramientas giratorias (CUTMOD, CUTMODK)....827
3.13.13 Trabajar con entornos de herramientas....835
3.13.13.1 Guardar entorno de herramienta (TOOLENV) 835
3.13.13.2 Borrar entorno de herramienta (DELTOOLENV) 838
3.13.13.3 Leer números T, D y DL (GETTENV) 839
3.13.13.4 Lectura de la información sobre los entornos de herramienta guardados ($P_TOOLENVN, ($P_TOOLENV) 840
3.13.13.5 Lectura de longitudes o componentes longitudinales de herramienta (GETTCOR) 840
3.13.13.6 Modificar componentes de herramienta (SETTCOR) 847
3.13.14 Leer la asignación de las longitudes de herramienta L1, L2, L3 a los ejes de coordenadas (LENTOAX) 859

3.14 Comportamiento de contorneado....863
3.14.1 Variación del avance (FNORM, FLIN, FCUB, FPO) 863
3.14.2 Comportamiento en aceleración 868
3.14.2.1 Modo de aceleración (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA) 868
3.14.2.2 Influencia en la aceleración en ejes de seguimiento (VELOLIMA, ACCLIMA, JERKLIMA)....870
3.14.2.3 Activación de los valores de dinámica específicos de tecnología (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH, DYNPREC)....871
3.14.3 Desplazamiento con mando anticipativo (FFWON, FFWOF) 873
3.14.4 Precisión del contorno programable (CPRECON, CPRECOF) 874
3.14.5 Ejecución del programa con memoria de pretratamiento (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE) 875
3.14.6 Definir las áreas Stop-Delay (DELAYFSTON, DELAYFSTOF) 878
3.14.7 Evitar lugar del programa para SERUPRO (IPTRLOCK, IPTRUNLOCK) 880
3.14.8 Reposicionamiento a contorno (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL) 882
3.14.9 Influencia en el guiado de movimiento....890
3.14.9.1 Adaptación de tirón por eje máximo (JERKLIM)....890
3.14.9.2 Adaptación de la velocidad máxima de eje o cabezal (VELOLIM) 891
3.14.10 Programar tolerancia de contorno/orientación (CTOL, OTOL, ATOL)....893
3.14.11 Comportamiento en cambio de secuencia con acoplamiento activo (CPBC)....898

3.15 Funciones de ejes....899
3.15.1 Intercambio de eje, intercambio de cabezales (RELEASE, GET, GETD) 899
3.15.2 Transferir el eje a otro canal (AXTOCHAN)....903
3.15.3 Funciones de eje (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)....904

3.15.4 Ejes geométricos conmutables (GEOAX) 906
3.15.5 Contenedor de ejes (AXCTSWE, AXCTSWED, AXCTSWEC) 911
3.15.6 Esperar a posición de eje válida (WAITENC) 913
3.15.7 Conmutación programable entre secuencias de parámetros (SCPARA) 914
3.15.8 Activar/desactivar adaptación (CADAPTON, CADAPTOF) 915
3.15.9 Adaptación del filtro antitirones FIR al modo dinámico (CALCFIR) 918

3.16 Acoplamientos de ejes 921
3.16.1 Arrastre de ejes (TRAILON, TRAILOF) 921
3.16.2 Tablas de levas (CTAB) 925
3.16.2.1 Definición de tablas de levas (CTABDEF, CATBEND) 925
3.16.2.2 Comprobación de la existencia de una tabla de levas (CTABEXISTS) 932
3.16.2.3 Borrado de tablas de levas (CTABDEL) 932
3.16.2.4 Bloqueo de las tablas de levas contra borrado y sobrescritura (CTABLOCK, CTABUNLOCK) 933
3.16.2.5 Tablas de levas: determinación de las propiedades de la tabla (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD) 934
3.16.2.6 Lectura de valores de las tablas de levas (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX) 936
3.16.2.7 Tablas de levas: comprobación del uso de los recursos (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL) 940
3.16.3 Acoplamiento entre ejes de valores maestros (LEADON, LEADOF) 942
3.16.4 Reductor electrónico (EG) 947
3.16.4.1 Definir reductor electrónico (EGDEF) 947
3.16.4.2 Activación del reductor electrónico (EGON, EGONSYN, EGONSYNE) 948
3.16.4.3 Desactivación del reductor electrónico (EGOFS, EGOFC) 952
3.16.4.4 Borrar la definición de un reductor electrónico (EGDEL) 953
3.16.4.5 Avance por vuelta (G95)/Reductor electrónico (FPR) 953
3.16.5 Cabezal síncrono 953
3.16.5.1 Cabezal síncrono: Programación (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC) 954
3.16.6 Acoplamiento genérico (CP...) 965
3.16.7 Control tangencial 973
3.16.7.1 Definición del acoplamiento (TANG) 973
3.16.7.2 Conexión de la creación de secuencia intermedia (TLIFT) 975
3.16.7.3 Activación de un acoplamiento (TANGON) 976
3.16.7.4 Desactivación del acoplamiento (TANGOF) 977
3.16.7.5 Borrado de un acoplamiento (TANGDEL) 977
3.16.8 Acoplamiento maestro-esclavo (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS) 979

3.17 Acciones síncronas 983
3.17.1 Definición de una acción síncrona 983

3.18 Vaivén 984
3.18.1 Vaivén asíncrono (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB) 984
3.18.2 Vaivén controlado mediante acciones síncronas (OSCILL) 989

3.19 Troquelado y punzonado 997
3.19.1 Activación/desactivación 997
3.19.1.1 Activar o desactivar punzonado y troquelado (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC) 997
3.19.2 División automática del recorrido en segmentos 1002

3.19.2.1 División en ejes de contorneado ... 1005
3.19.2.2 División en ejes individuales ... 1006

3.20 Rectificado ... 1008
3.20.1 Activación/desactivación de la vigilancia específica de muelas (TMON, TMOF) ... 1008

3.21 Parada y retirada ampliada (ESR) ... 1009
3.21.1 ESR controlada por CN ... 1010
3.21.1.1 Retirada controlada por CN (POLF, POLFA, POLFMASK, POLFMLIN) ... 1010
3.21.1.2 Parada controlada por CN ... 1014
3.21.2 ESR integrada en el accionamiento ... 1015
3.21.2.1 Configurar la parada ejecutada en el propio accionamiento (ESRS) ... 1015
3.21.2.2 Configurar la retirada ejecutada en el propio accionamiento (ESRS) ... 1015

3.22 Tiempo de ejecución de programa/contador de piezas ... 1017
3.22.1 Tiempo de ejecución del programa ... 1017
3.22.2 Contador de piezas ... 1020

3.23 Otras funciones ... 1022
3.23.1 Activar los datos de máquina (NEWCONF) ... 1022
3.23.2 Comprobar el repertorio del lenguaje CN disponible (STRINGIS) ... 1023
3.23.3 Llamar interactivamente la ventana del programa de pieza (MMC) ... 1026
3.23.4 Process DataShare - Salida a un equipo/fichero externo (EXTOPEN, WRITE, EXTCLOSE): ... 1031
3.23.5 Alarmas (SETAL) ... 1035
3.23.6 Definir pieza en bruto (WORKPIECE) ... 1037
3.23.7 Cambiar modo de lenguaje (G290, G291) ... 1041

3.24 Programas de desbaste propios ... 1044
3.24.1 Funciones de apoyo para el desbaste ... 1044
3.24.2 Creación de tabla para el contorno (CONTPRON) ... 1044
3.24.3 Creación de tabla para el contorno codificada (CONTDCON) ... 1050
3.24.4 Determinación del punto de intersección entre dos elementos de contorno (INTERSEC) ... 1054
3.24.5 Retirada de los elementos de contorno de una tabla secuencia a secuencia (EXECTAB) ... 1056
3.24.6 Cálculo de los datos de circunferencia (CALCDAT) ... 1056
3.24.7 Desactivación de la preparación del contorno (EXECUTE) ... 1058

3.25 Programar ciclos externamente ... 1060
3.25.1 Ciclos tecnológicos ... 1060
3.25.1.1 Introducción ... 1060
3.25.1.2 Vista general específica de tecnología ... 1061
3.25.1.3 HOLES1, patrón de posiciones línea ... 1063
3.25.1.4 HOLES2, patrón de posiciones arco circular o arco ... 1064
3.25.1.5 POCKET3, caja rectangular ... 1066
3.25.1.6 POCKET4, caja circular ... 1068
3.25.1.7 SLOT1, ranura longitudinal ... 1071
3.25.1.8 SLOT2, ranura circular ... 1074
3.25.1.9 LONGHOLE, agujero rasgado ... 1076
3.25.1.10 CYCLE60, grabado ... 1078
3.25.1.11 CYCLE61, planear ... 1081
3.25.1.12 CYCLE62, llamada de contorno ... 1083
3.25.1.13 CYCLE63, fresar caja de contorno/material sobrante caja de contorno/fresar saliente de contorno/material sobrante saliente de contorno ... 1084
3.25.1.14 CYCLE64, pretaladrar caja de contorno ... 1086
3.25.1.15 CYCLE70, fresado de roscas ... 1088

3.25.1.16 CYCLE72, fresado en contorneado 1090
3.25.1.17 CYCLE76, saliente rectangular 1094
3.25.1.18 CYCLE77, saliente circular 1096
3.25.1.19 CYCLE78, fresado de roscas 1098
3.25.1.20 CYCLE79, poliedro 1100
3.25.1.21 CYCLE81, taladrado, punteado 1102
3.25.1.22 CYCLE82, taladrado, avellanado 1103
3.25.1.23 CYCLE83, taladrado profundo 1 1106
3.25.1.24 CYCLE84, roscado sin mandril de compensación 1109
3.25.1.25 CYCLE85, escariado 1112
3.25.1.26 CYCLE86, mandrinar 1113
3.25.1.27 CYCLE92, tronzar 1115
3.25.1.28 CYCLE95, mecanizado de contorno 1116
3.25.1.29 CYCLE98, cadena de roscas 1118
3.25.1.30 CYCLE99, tallado de roscas 1122
3.25.1.31 CYCLE435 - Definir el sistema de coordenadas del diamante 1127
3.25.1.32 CYCLE495, perfilar 1127
3.25.1.33 CYCLE782: Adaptar a la carga 1129
3.25.1.34 CYCLE800, orientar plano/orientar herramienta/alinear herramienta 1132
3.25.1.35 CYCLE801, patrón de posiciones rejilla o marco 1136
3.25.1.36 CYCLE802, posiciones libres 1137
3.25.1.37 CYCLE830, taladrado profundo 2 1140
3.25.1.38 CYCLE832, High Speed Settings 1146
3.25.1.39 CYCLE840, roscado con mandril de compensación 1150
3.25.1.40 CYCLE899, ranura abierta 1152
3.25.1.41 CYCLE930, ranurado 1155
3.25.1.42 CYCLE940, garganta forma E y F/garganta rosca 1157
3.25.1.43 CYCLE951, desbaste 1160
3.25.1.44 CYCLE952, desbastar/desbastar resto/ranurar/ranurar resto/ranurar derecha-izquierda/ranurar derecha-izquierda resto 1163
3.25.1.45 CYCLE4071 - Rectificado longitudinal con penetración en el punto de inversión 1169
3.25.1.46 CYCLE4072 - Rectificado longitudinal con penetración en el punto de inversión y señal de cancelación 1171
3.25.1.47 CYCLE4073 - Rectificado longitudinal con penetración continua 1175
3.25.1.48 CYCLE4074 - Rectificado longitudinal con penetración continua y señal de cancelación 1176
3.25.1.49 CYCLE4075 - Rectificado plano con penetración en el punto de inversión 1179
3.25.1.50 CYCLE4077 - Rectificado plano con penetración en el punto de inversión y señal de cancelación 1182
3.25.1.51 CYCLE4078 - Rectificado plano con penetración continua 1186
3.25.1.52 CYCLE4079 - Rectificado plano con penetración intermitente 1188
3.25.1.53 GROUP_BEGIN - Inicio bloque de programa 1191
3.25.1.54 GROUP_END - Fin bloque de programa 1191
3.25.1.55 GROUP_ADDEND - Fin adición aprox. 1192
3.25.1.56 Condiciones marginales 1192
3.25.2 Ciclos de medida 1194
3.25.3 Vista general de parámetros de ciclos de medida 1194
3.25.3.1 Parámetros de ciclo de medida CYCLE973 1194
3.25.3.2 Parámetros de ciclo de medida CYCLE974 1197
3.25.3.3 Parámetros de ciclo de medida CYCLE994 1199
3.25.3.4 Parámetros de ciclo de medida CYCLE976 1202
3.25.3.5 Parámetros de ciclo de medida CYCLE978 1205
3.25.3.6 Parámetros de ciclo de medida CYCLE998 1208

3.25.3.7 Parámetros de ciclo de medida CYCLE977 ... 1210
3.25.3.8 Parámetros de ciclo de medida CYCLE961 ... 1214
3.25.3.9 Parámetros de ciclo de medida CYCLE979 ... 1217
3.25.3.10 Parámetros de ciclo de medida CYCLE997 ... 1220
3.25.3.11 Parámetros de ciclo de medida CYCLE995 ... 1223
3.25.3.12 Parámetros de ciclo de medida CYCLE996 ... 1224
3.25.3.13 Parámetros de ciclo de medida CYCLE9960 ... 1227
3.25.3.14 Parámetros de ciclo de medida CYCLE982 ... 1229
3.25.3.15 Parámetros de ciclo de medida CYCLE971 ... 1231
3.25.3.16 Parámetros de ciclo de medida CYCLE150 ... 1234

4 Tablas ... 1237

4.1 Instrucciones ... 1237

4.2 Direcciones ... 1280
4.2.1 Letras para direcciones ... 1280
4.2.2 Direcciones fijas ... 1281
4.2.3 Direcciones ajustables ... 1286

4.3 Comandos G ... 1293
4.3.1 Comandos G ... 1293
4.3.2 Grupo G 1: Comandos de desplazamiento modales ... 1293
4.3.3 Grupo G 2: Desplazamientos por secuencia, tiempo de espera ... 1294
4.3.4 Grupo G 3: Frame programable, limitación del campo de trabajo y programación del polo ... 1294
4.3.5 Grupo G 4: FIFO ... 1295
4.3.6 Grupo G 6: Selección del plano ... 1295
4.3.7 Grupo G 7: Corrección de radio de herramienta ... 1296
4.3.8 Grupo G 8: Decalaje de origen ajustable ... 1296
4.3.9 Grupo G 9: Supresión de la corrección de frames y de herramientas ... 1297
4.3.10 Grupo G 10: Parada precisa - Modo Contorneado ... 1297
4.3.11 Grupo G 11: Parada precisa por secuencia ... 1298
4.3.12 Grupo G 12: Criterios de cambio de secuencia para parada precisa (G60/G9) ... 1298
4.3.13 Grupo G 13: Acotado de piezas en métrico/pulgadas ... 1298
4.3.14 Grupo G 14: Acotado de piezas absoluto/incremental ... 1298
4.3.15 Grupo G 15: Tipo de avance ... 1299
4.3.16 Grupo G 16: Corrección de avance en curvatura interna y externa ... 1299
4.3.17 Grupo G 17: Comportamiento en aproximación y retirada, corrección de herramienta ... 1300
4.3.18 Grupo G 18: Comportamiento en esquinas, corrección de herramienta ... 1300
4.3.19 Grupo G 19: Transición de curva al inicio de Spline ... 1300
4.3.20 Grupo G 20: Transición de curva al final de Spline ... 1301
4.3.21 Grupo G 21: Perfil de aceleración ... 1301
4.3.22 Grupo G 22: Tipo de corrección de herramienta ... 1301
4.3.23 Grupo G 23: Vigilancia de colisión en contornos internos ... 1302
4.3.24 Grupo G 24: Control anticipativo ... 1302
4.3.25 Grupo G 25: Relación orientación de herramienta ... 1302
4.3.26 Grupo G 26: Modo de reposicionamiento para REPOS (de efecto modal) ... 1303
4.3.27 Grupo G 27: Corrección de herramienta en cambios de orientación en esquinas exteriores ... 1303
4.3.28 Grupo G 28: Limitación del campo de trabajo ... 1303
4.3.29 Grupo G 29: Programación por radio/diámetro ... 1303
4.3.30 Grupo G 30: Compresión de secuencias CN ... 1304
4.3.31 Grupo G 31: Comandos G OEM ... 1304
4.3.32 Grupo G 32: Comandos G OEM ... 1305

4.3.33 Grupo G 33: Corrección de herramienta fina ajustable 1305
4.3.34 Grupo G 34: Alisado de la orientación de herramienta 1305
4.3.35 Grupo G 35: Troquelado y punzonado 1306
4.3.36 Grupo G 36: Troquelado con retardo 1306
4.3.37 Grupo G 37: Perfil de avance 1306
4.3.38 Grupo G 38: Asignación de entradas/salidas rápidas para punzonado/troquelado 1306
4.3.39 Grupo G 39: Precisión de contorno programable 1307
4.3.40 Grupo G 40: Corrección de radio de corte/herramienta constante 1307
4.3.41 Grupo G 41: Tallado de roscas con posibilidad de interrupción 1307
4.3.42 Grupo G 42: Portaherramientas 1307
4.3.43 Grupo G 43: Dirección de aproximación/retirada suaves 1308
4.3.44 Grupo G 44: División de segmentos aprox./retirada suaves 1308
4.3.45 Grupo G 45: Trayectoria de referencia de los ejes FGROUP 1308
4.3.46 Grupo G 46: Selección de planos para retirada rápida 1309
4.3.47 Grupo G 47: Conmutación de modo para código CN externo 1309
4.3.48 Grupo G 48: Comportamiento en aproximación y retirada en la corrección de radio de herramienta 1309
4.3.49 Grupo G 49: Desplazamiento punto a punto 1309
4.3.50 Grupo G 50: Programación de la orientación de la herramienta 1310
4.3.51 Grupo G 51: Tipo de interpolación en la programación de la orientación 1310
4.3.52 Grupo G 52: Rotación de frame relativa a la pieza 1311
4.3.53 Grupo G 53: Rotación de frame relativa a la herramienta 1311
4.3.54 Grupo G 54: Rotación vectorial en la programación polinómica 1312
4.3.55 Grupo G 55: Desplazamiento en rápido con/sin interpolación lineal 1312
4.3.56 Grupo G 56: Inclusión del desgaste de la herramienta 1312
4.3.57 Grupo G 57: Deceleración en los dos vértices 1313
4.3.58 Grupo G 59: Modo dinámico para interpolación de la trayectoria 1313
4.3.59 Grupo G 60: Limitación del campo de trabajo 1313
4.3.60 Grupo G 61: Alisado de la orientación de herramienta 1314
4.3.61 Grupo G 62: Modo de reposicionamiento para REPOS (efectivo por secuencia) 1314
4.3.62 Grupo G 64: Frames de rectificado 1314
4.4 Procedimientos predefinidos 1316
4.5 Procedimientos predefinidos en acciones síncronas 1338
4.6 Funciones predefinidas 1340
4.7 Idioma actual del HMI 1355
A Anexo: 1357
A.1 Lista de abreviaturas 1357
A.2 Vista general de la documentación 1366
Índice alfabético 1367

Consignas básicas de seguridad

1

1.1 Consignas generales de seguridad

⚠ ADVERTENCIA

Peligro de muerte en caso de incumplimiento de las consignas de seguridad e inobservancia de los riesgos residuales

Si no se cumplen las consignas de seguridad ni se tienen en cuenta los riesgos residuales de la documentación de hardware correspondiente, pueden producirse accidentes con consecuencias mortales o lesiones graves.

- Respete las consignas de seguridad de la documentación de hardware.
- Tenga en cuenta los riesgos residuales durante la evaluación de riesgos.

⚠ ADVERTENCIA

Fallos de funcionamiento de la máquina a consecuencia de una parametrización errónea o modificada

Una parametrización errónea o modificada puede provocar en máquinas fallos de funcionamiento que pueden producir lesiones graves o la muerte.

- Proteja la parametrización del acceso no autorizado.
- Controle los posibles fallos de funcionamiento con medidas apropiadas, p. ej., DESCONEXIÓN o PARADA DE EMERGENCIA.

1.2 Garantía y responsabilidad para ejemplos de aplicación

Los ejemplos de aplicación no son vinculantes y no pretenden ser completos en cuanto a la configuración y al equipamiento, así como a cualquier eventualidad. Los ejemplos de aplicación tampoco representan una solución específica para el cliente; simplemente ofrecen una ayuda para tareas típicas.

El usuario es responsable del correcto manejo y uso de los productos descritos. Los ejemplos de aplicación no le eximen de la obligación de trabajar de forma segura durante la aplicación, la instalación, el funcionamiento y el mantenimiento.

1.3 Seguridad industrial

Nota

Seguridad industrial

Siemens ofrece productos y soluciones con funciones de seguridad industrial con el objetivo de hacer más seguro el funcionamiento de instalaciones, sistemas, máquinas y redes.

Para proteger las instalaciones, los sistemas, las máquinas y las redes de amenazas cibernéticas, es necesario implementar (y mantener continuamente) un concepto de seguridad industrial integral que sea conforme a la tecnología más avanzada. Los productos y las soluciones de Siemens constituyen únicamente una parte de este concepto.

Los clientes son responsables de impedir el acceso no autorizado a sus instalaciones, sistemas, máquinas y redes. Dichos sistemas, máquinas y componentes solo deben estar conectados a la red corporativa o a Internet cuando y en la medida que sea necesario y siempre que se hayan tomado las medidas de protección adecuadas (p. ej. uso de cortafuegos y segmentación de la red).

Para obtener información adicional sobre las medidas de seguridad industrial que podrían ser implementadas, por favor visite:

Seguridad industrial (https://www.siemens.com/industrialsecurity)

Los productos y las soluciones de Siemens están sometidos a un desarrollo constante con el fin de mejorar todavía más su seguridad. Siemens recomienda expresamente realizar actualizaciones en cuanto estén disponibles y utilizar únicamente las últimas versiones de los productos. El uso de versiones anteriores o que ya no se soportan puede aumentar el riesgo de amenazas cibernéticas.

Para mantenerse siempre informado de las actualizaciones de productos, suscríbase al Siemens Industrial Security RSS Feed en:

Seguridad industrial (https://www.siemens.com/industrialsecurity)

Encontrará más información en Internet:

Manual de configuración de Industrial Security (https://support.industry.siemens.com/cs/ww/es/view/108862708/en)

⚠ ADVERTENCIA

Estados operativos no seguros debidos a una manipulación del software

Las manipulaciones del software (p. ej. mediante virus, troyanos o gusanos) pueden provocar estados operativos inseguros en la instalación, con consecuencias mortales, lesiones graves o daños materiales.

- Mantenga actualizado el software.
- Integre los componentes de automatización y accionamiento en un sistema global de seguridad industrial de la instalación o máquina conforme a las últimas tecnologías.
- En su sistema global de seguridad industrial, tenga en cuenta todos los productos utilizados.
- Proteja los archivos almacenados en dispositivos de almacenamiento extraíbles contra software malicioso tomando las correspondientes medidas de protección, p. ej. programas antivirus.
- Al finalizar la puesta en marcha, compruebe todos los ajustes relevantes para la seguridad.
- Proteja el accionamiento contra modificaciones no autorizadas activando la función "Protección de know-how" del convertidor.

Fundamentos

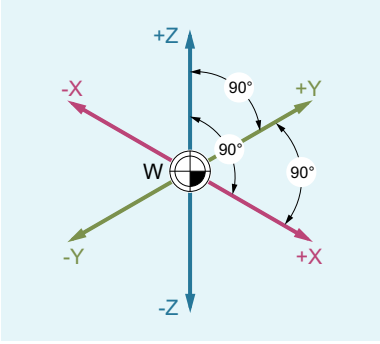
2

2.1 Fundamentos geométricos

2.1.1 Posiciones de pieza

2.1.1.1 Sistema de referencia de los datos de posición

Para que la máquina o el control puedan trabajar con las posiciones especificadas en el programa CN, estos datos de posición deben estar vinculados a un sistema de referencia que pueda transferirse a las direcciones de desplazamiento de los ejes de la máquina. Con este fin, en las máquinas herramienta se utilizan sistemas de coordenadas cartesianas, es decir, dextrógiros y ortogonales según DIN 66217, para el sistema de coordenadas de pieza.



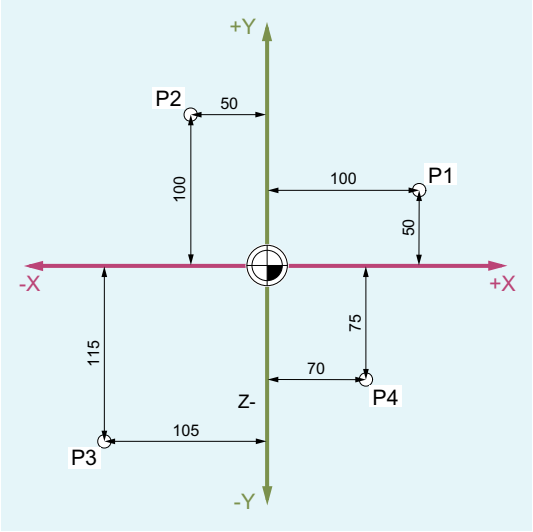
El origen de pieza (W) es el origen del sistema de coordenadas de la pieza.

2.1.1.2 Coordenadas cartesianas

Los ejes del sistema de coordenadas están acotados. De este modo es posible describir unívocamente todos los puntos del sistema de coordenadas y, con ello, todas las posiciones de pieza mediante la dirección (X, Y y Z) y tres valores numéricos. El origen siempre tiene las coordenadas X0, Y0 y Z0.

Indicaciones de posición en forma de coordenadas cartesianas

Para mayor simplicidad vamos a considerar en el siguiente ejemplo un solo plano del sistema de coordenadas, el plano X/Y:

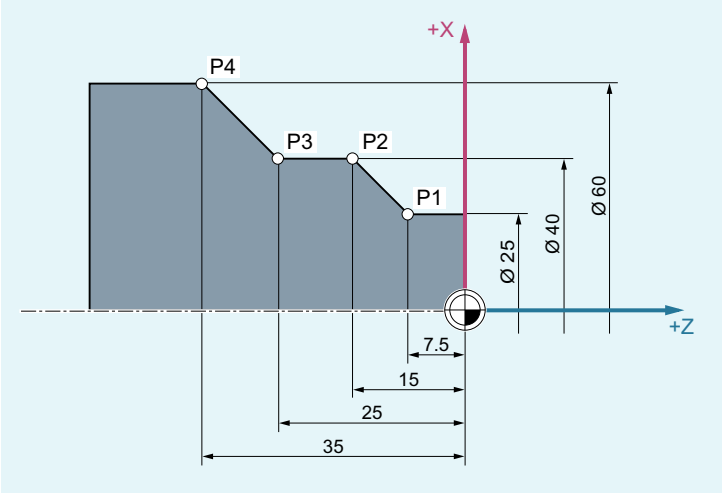


Los puntos P1 a P4 tienen las siguientes coordenadas:

Posición	Coordenadas
P1	X100 Y50
P2	X-50 Y100
P3	X-105 Y-115
P4	X70 Y-75

Ejemplo: Posiciones de pieza en el torneado

En tornos basta un plano para describir el contorno:

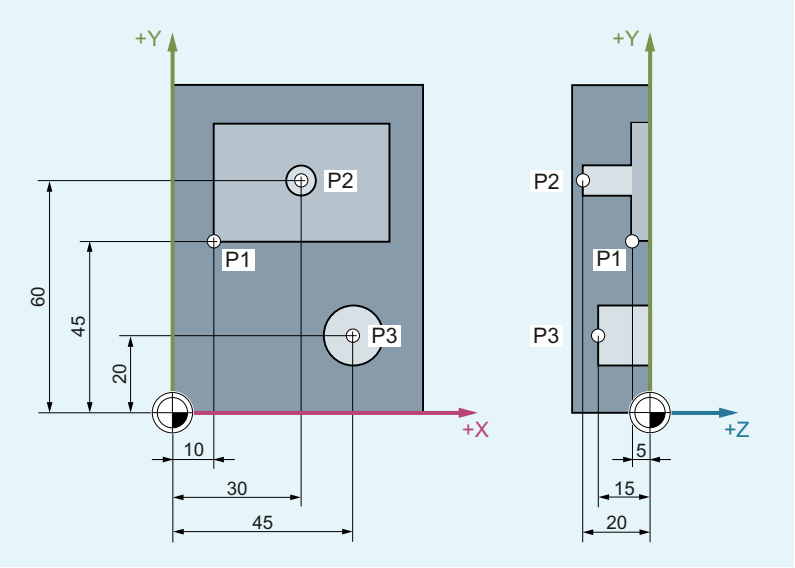


Los puntos P1 a P4 tienen las siguientes coordenadas:

Posición	Coordenadas
P1	X25 Z-7.5
P2	X40 Z-15
P3	X40 Z-25
P4	X60 Z-35

Ejemplo: Posiciones de pieza en el fresado

En los mecanizados de fresado debe describirse también la profundidad de penetración, es decir, también debe asignarse un valor numérico a la tercera coordenada (en este caso, Z).



Los puntos P1 a P3 tienen las siguientes coordenadas:

Posición	Coordenadas
P1	X10 Y45 Z-5
P2	X30 Y60 Z-20
P3	X45 Y20 Z-15

2.1.1.3 Coordenadas polares

Para describir posiciones de pieza pueden utilizarse también coordenadas polares, en vez de coordenadas cartesianas. Se utilizan cuando una pieza o una parte de ella está acotada con radios y ángulos. El punto donde se cortan los radios se denomina "polo".

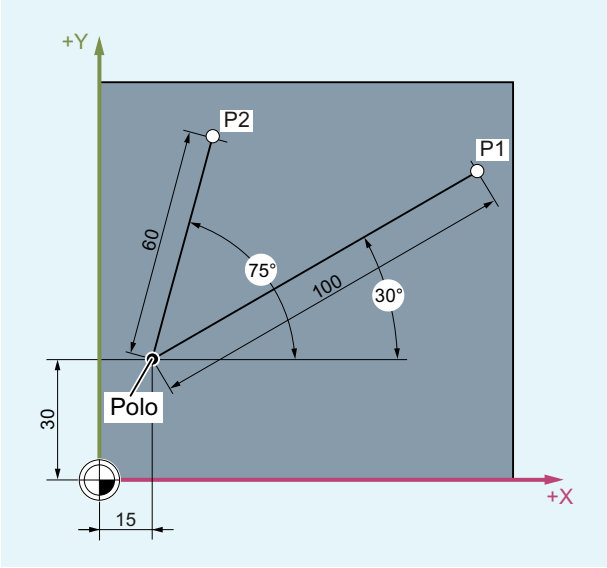
Indicaciones de posición en forma de coordenadas polares

Las coordenadas polares se componen del **radio polar** y el **ángulo polar**.

El radio polar es la distancia entre el polo y la posición.

El ángulo polar es el ángulo entre el radio polar y el eje horizontal del plano de trabajo. Los ángulos polares negativos discurren en sentido horario, mientras que los positivos lo hacen en sentido antihorario.

Ejemplo



Los puntos P1 y P2 pueden describirse como sigue en relación con el polo:

Posición	Coordenadas polares
P1	RP=100 AP=30
P2	RP=60 AP=75
RP: Radio polar AP: Ángulo polar	

2.1.1.4 Acotado absoluto

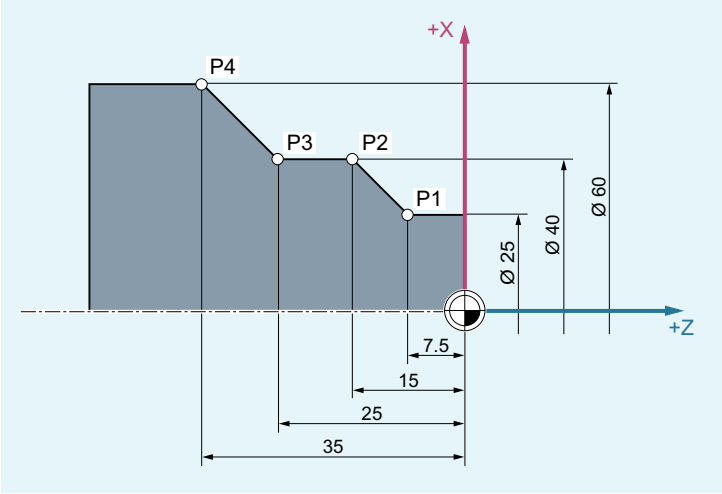
Indicaciones de posición en cotas absolutas

Cuando se programa utilizando cotas absolutas, todas las posiciones introducidas están referidas al origen válido en ese momento.

Esto se traduce en que para el desplazamiento de la herramienta:

La cota absoluta describe la posición a la cual se deberá desplazar la herramienta.

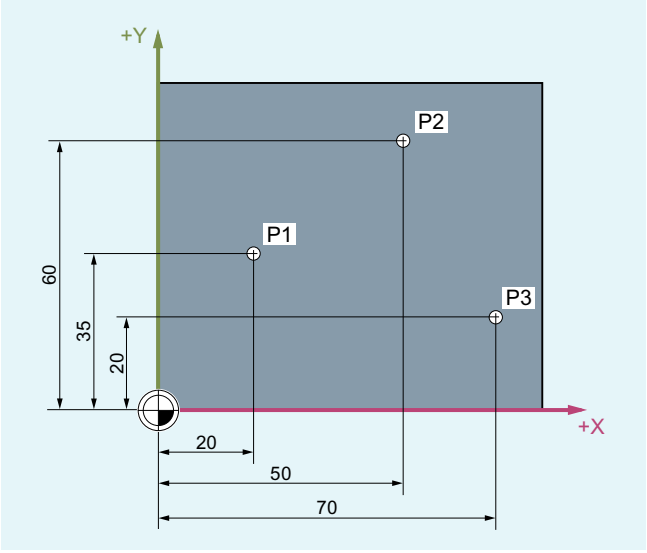
Ejemplo: Torneado



En el acotado absoluto se dan las siguientes indicaciones de posición para los puntos P1 a P4:

Posición	Indicación de posición en cotas absolutas
P1	X25 Z-7,5
P2	X40 Z-15
P3	X40 Z-25
P4	X60 Z-35

Ejemplo: Fresado



En el acotado absoluto se dan las siguientes indicaciones de posición para los puntos P1 a P3:

Posición	Indicación de posición en cotas absolutas
P1	X20 Y35
P2	X50 Y60
P3	X70 Y20

2.1.1.5 Acotado incremental

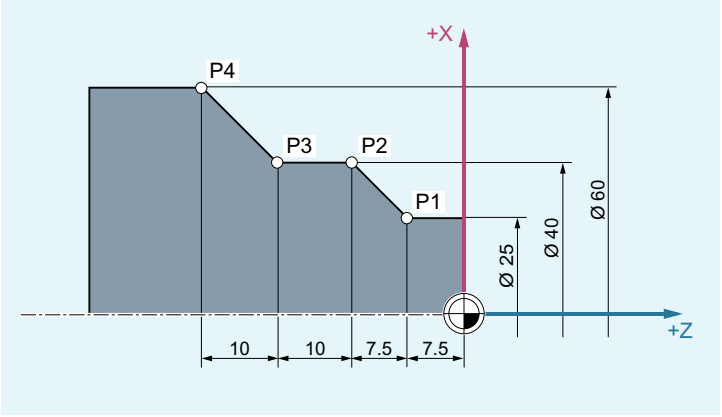
Indicaciones de posición en cotas incrementales

En los planos de fabricación, a menudo las cotas no se refieren al origen, sino a otro punto de la pieza. Para no tener que recalcular estas cotas, existe la posibilidad de utilizar un acotado incremental. En este tipo de acotado, una indicación de posición hace referencia al punto anterior.

Esto se traduce en que para el desplazamiento de la herramienta:

El acotado incremental indica qué distancia se debe desplazar la herramienta.

Ejemplo: torneado



En el acotado incremental se dan las siguientes indicaciones de posición para los puntos P2 a P4:

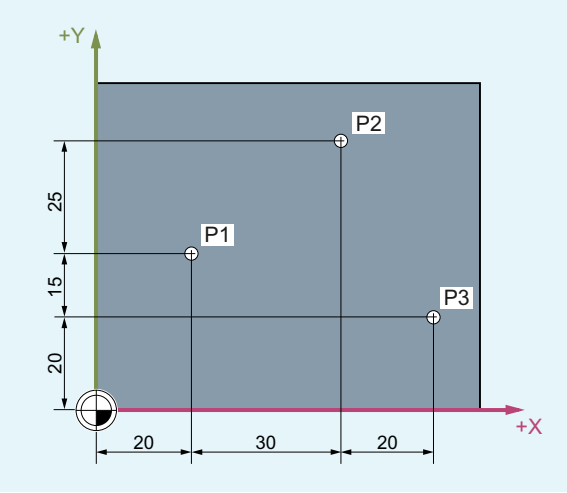
Posición	Indicación de posición en cotas incrementales	La indicación se refiere a:
P2	X15 Z-7,5	P1
P3	Z-10	P2
P4	X20 Z-10	P3

Nota

Si está activado DIAMOF o DIAM90 (Página 166), el trayecto teórico en cotas incrementales (G91) se programa en radios.

Ejemplo: fresado

Las cotas incrementales para los puntos P1 a P3 son:



En el acotado incremental se dan las siguientes indicaciones de posición para los puntos P1 a P3:

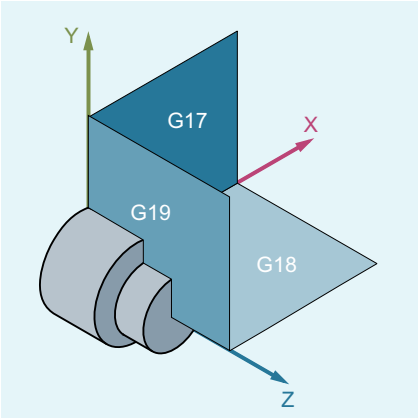
Posición	Indicación de posición en cotas incrementales	La indicación se refiere a:
P1	X20 Y35	Origen
P2	X30 Y20	P1
P3	X20 Y-35	P2

2.1.2 Planos de trabajo

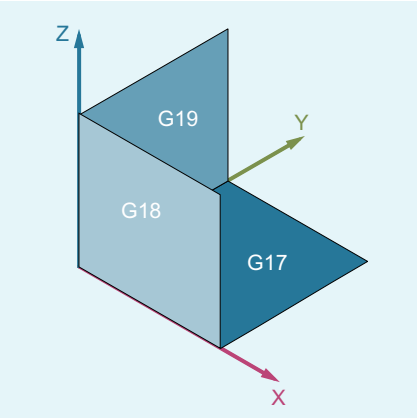
Un programa CN necesita conocer el plano en el que se realizará el mecanizado. Esto es imprescindible para que el control pueda, p. ej., calcular correctamente los valores de corrección de herramienta. Además, para determinados tipos de programación de círculos y en caso de usar coordenadas polares, es necesario indicar el plano de trabajo.

El plano de trabajo se define en el sistema de coordenadas de pieza cartesiano subyacente mediante dos ejes de coordenadas. El tercer eje de coordenadas es perpendicular a dicho plano de trabajo y define la dirección de penetración de la herramienta (p. ej., para mecanizado 2D).

Planos de trabajo en el torneado/fresado



Planos de trabajo en el torneado



Planos de trabajo en el fresado

Activación de un plano de trabajo

Los planos de trabajo se activan en el programa CN con los comandos G G17, G18 y G19. La relación se define de la siguiente manera:

Comando G	Plano de trabajo	Abscisa	Ordenada	Eje de aplicadas ≙ dirección de penetración
G17	X/Y	X	Y	Z
G18	Z/X	Z	X	Y
G19	Y/Z	Y	Z	X

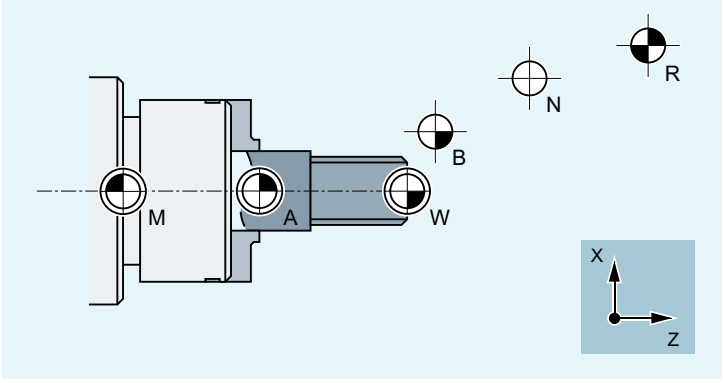
2.1.3 Orígenes y puntos de referencia

En una máquina CN hay definidos diferentes orígenes y puntos de referencia:

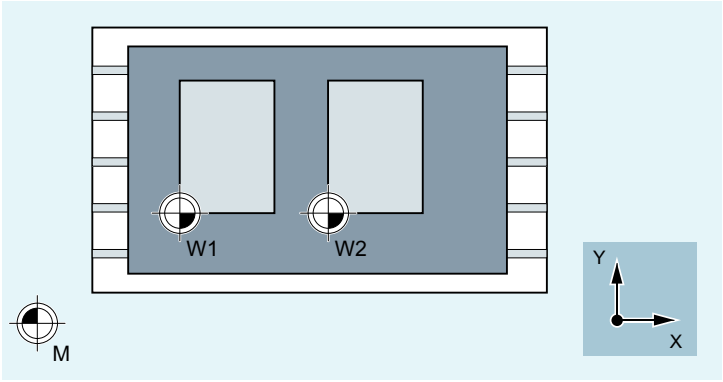
Orígenes		
	M	Origen de la máquina El origen de la máquina sirve para establecer el sistema de coordenadas de máquina (MKS). Todos los demás puntos de referencia se refieren al origen de la máquina.
	W	Origen de pieza = Origen del programa El origen de la pieza establece el sistema de coordenadas de pieza en relación con el origen de la máquina.
	A	Punto de tope Puede coincidir con el origen de la pieza (sólo en tornos).

Puntos de referencia		
	R	Punto de referencia Posición establecida por la leva y el sistema de medida. Es preciso conocer la distancia respecto al origen de la máquina **M** para poder ajustar la posición del eje en este punto exactamente a este valor.
	B	Punto inicial Definible mediante programa. Aquí comienza la 1.ª herramienta del mecanizado.
	T	Punto de referencia del portaherramientas Se encuentra en el alojamiento del portaherramientas. Si se introducen las longitudes de herramienta, el control calcula la distancia de la punta de la herramienta respecto al punto de referencia del portaherramientas.
	N	Punto de cambio de la herramienta

Orígenes y puntos de referencia en el torneado



Orígenes en el fresado



2.1.4 Sistemas de coordenadas

Se distinguen los siguientes sistemas de coordenadas:

- Sistema de coordenadas de máquina (MKS) (Página 36) con el origen de máquina **M**
- Sistema de coordenadas básico (BKS) (Página 38)
- Sistema de origen básico (BNS) (Página 39)
- Sistema de origen ajustable (ENS) (Página 40)
- Sistema de coordenadas de pieza (WKS) (Página 41) con el origen de pieza **W**

2.1.4.1 Sistema de coordenadas de máquina (MKS)

El sistema de coordenadas de máquina se forma utilizando los ejes físicos existentes en la máquina.

En el sistema de coordenadas de la máquina se definen puntos de referencia, puntos de cambio de soporte de pieza y de herramienta (puntos fijos predefinidos).

Cuando se programa con cotas referidas al sistema de coordenadas de máquina (utilizando los comandos G que así lo permiten) se están desplazando directamente los ejes físicos de la máquina. No se considera un eventual elemento de amarre de pieza.

Nota

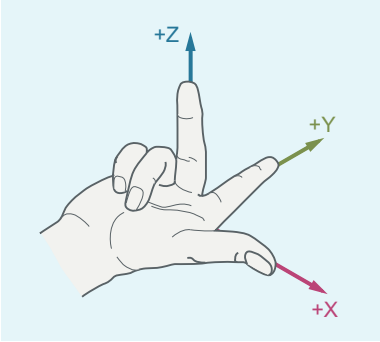
Si existen diferentes sistemas de coordenadas de máquina (p. ej., transformación de 5 ejes), la cinemática de la máquina se reproduce, por transformación interna, en el sistema de coordenadas en el cual se realiza la programación.

Regla de los tres dedos

La posición del sistema de coordenadas asociado a la máquina depende del tipo de máquina. Las direcciones de eje siguen la llamada "regla de los tres dedos" de la mano **derecha** (según DIN 66217).

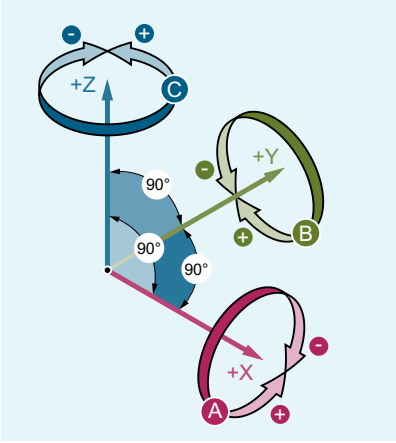
Situándose delante de la máquina con el dedo corazón de la mano derecha en dirección opuesta al la dirección de penetración del cabezal. Entonces:

- El pulgar indica el sentido +X
- El índice indica el sentido +Y
- El dedo corazón indica el sentido +Z



Los movimientos giratorios en torno a los ejes de coordenadas X, Y y Z se identifican con A, B y C. El sentido de giro se obtiene a partir del sentido del movimiento giratorio mirando hacia el sentido positivo del eje de coordenadas:

Sentido del movimiento giratorio	Sentido de giro
Sentido horario	Positivo
Sentido antihorario	Negativo

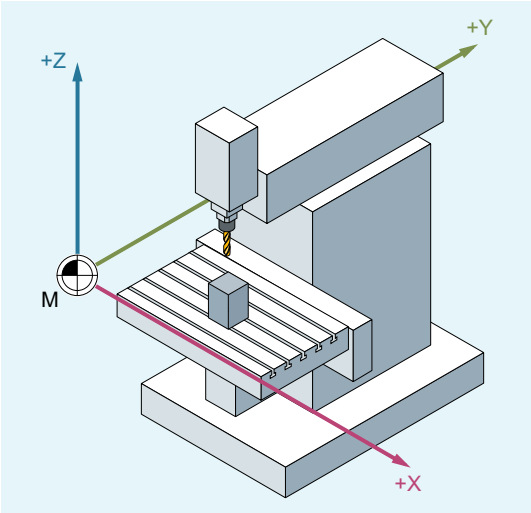


X, Y, Z Ejes de coordenadas perpendiculares entre sí

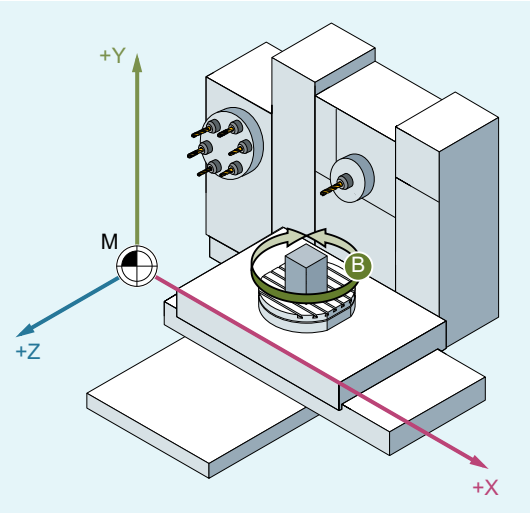
A, B, C Ejes giratorios alrededor de X, Y y Z

Posición del sistema de coordenadas en diferentes tipos de máquinas

La posición del sistema de coordenadas resultante de la "regla de los tres dedos" puede estar orientada de diferentes maneras en los diferentes tipos de máquinas, tal como se muestra en los dos siguientes ejemplos:



Fresadora vertical de 3 ejes



Fresadora horizontal de 4 ejes

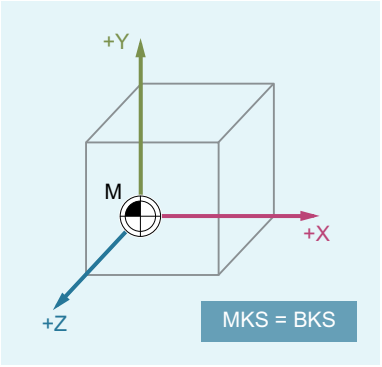
2.1.4.2 Sistema de coordenadas básico (BKS)

El sistema de coordenadas básico (BKS) consta de tres ejes dispuestos perpendicularmente (ejes geométricos) y otros ejes (ejes adicionales) sin relación geométrica.

Máquinas herramienta sin transformada cinemática

El BKS y el MKS coinciden siempre que el BKS puede derivarse del MKS sin transformada cinemática (p. ej., transformada de 5 ejes, TRANSMIT/TRACYL/TRAANG).

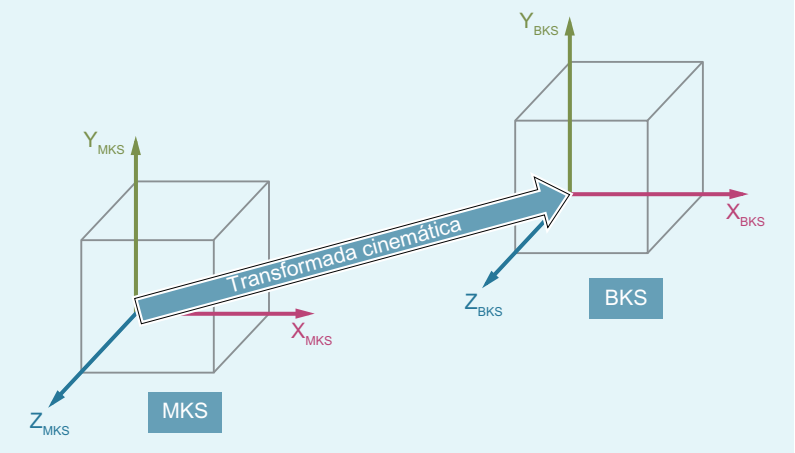
En estas máquinas, los ejes de máquina y los ejes geométricos pueden tener el mismo nombre.



Máquinas herramienta con transformada cinemática

El BKS y el MKS no coinciden cuando el BKS puede derivarse del MKS con transformada cinemática (p. ej., transformada de 5 ejes, TRANSMIT/TRACYL/TRAANG).

En estas máquinas, los ejes de máquina y los ejes geométricos deben tener nombres diferentes.

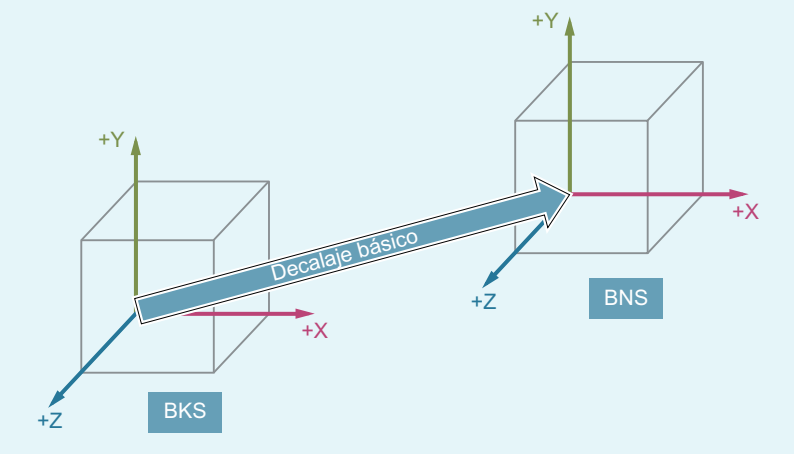


Cinemática de máquina

La pieza se programa siempre en un sistema de coordenadas cartesiano (WKS) bidimensional o tridimensional. Sin embargo, para mecanizar estas piezas se utilizan cada vez con más frecuencia máquinas herramienta con ejes giratorios o ejes lineales dispuestos de manera no cartesiana. Para reproducir las coordenadas programadas en el WKS (cartesianamente) en movimientos de eje de máquina reales se utiliza la transformada cinemática.

2.1.4.3 Sistema de origen básico (BNS)

El sistema de origen básico (BNS) se origina a partir del sistema de coordenadas básico por medio del decalaje básico.



Decalaje básico

El decalaje básico describe la transformación de coordenadas entre el BKS y el BNS. Permite establecer, p. ej., el origen de las paletas.

El decalaje básico se compone de:

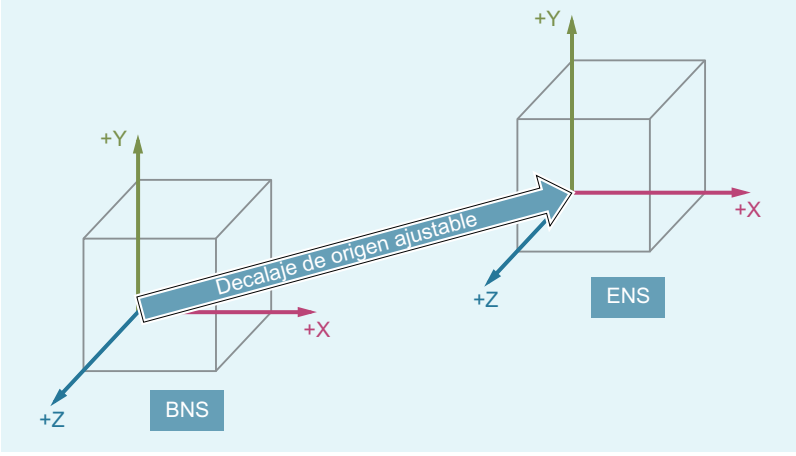
- Decalaje de origen externo
- Decalaje DRF
- Movimiento superpuesto
- Frames de sistema concatenados
- Frames básicos concatenados

2.1.4.4 Sistema de origen ajustable (ENS)

Decalaje de origen ajustable

Mediante el decalaje de origen ajustable se origina a partir del sistema de origen básico (BNS) el "sistema de origen ajustable" (ENS).

Los decalajes de origen ajustables se activan en el programa CN con los comandos G G54...G57 y G505...G599.



Si no hay activa ninguna transformación de coordenadas (frames) programable, el "sistema de origen ajustable" es el sistema de coordenadas de pieza (WKS).

Transformaciones de coordenadas (frames) programables

A veces es necesario o conveniente dentro de un mismo programa CN desplazar a otro sitio y, en caso necesario, girar, invertir en simetría especular o escalar el sistema de coordenadas de pieza seleccionado originalmente (o el "sistema de origen ajustable"). Esto se realiza mediante transformaciones de coordenadas (frames) (Página 314) programables.

Nota

Las transformaciones de coordenadas programables (frames) se refieren siempre al "sistema de origen ajustable".

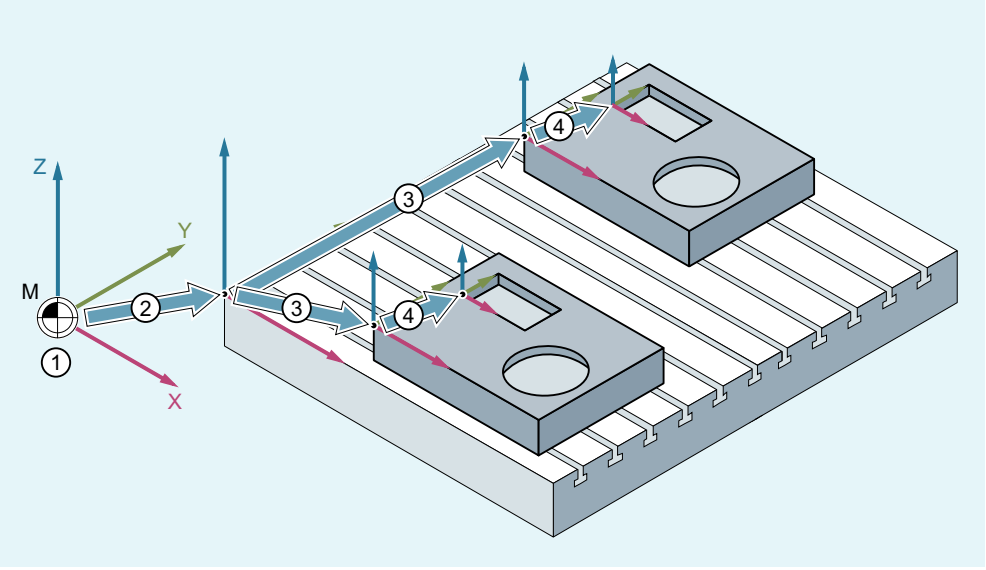
2.1.4.5 Sistema de coordenadas de pieza (WKS)

En el sistema de coordenadas de pieza (WKS) se describe la geometría de una pieza. O, expresado de otra manera: Los datos del programa CN están referidos al sistema de coordenadas de pieza.

El sistema de coordenadas de pieza es un sistema de coordenadas cartesiano asociado a una determinada pieza.

2.1.4.6 ¿Cómo se relacionan los diferentes sistemas de coordenadas?

El siguiente ejemplo ilustra las correspondencias entre los diferentes sistemas de coordenadas:



① La transformada cinemática no está activa. Esto significa que el sistema de coordenadas de máquina y el sistema de coordenadas básico coinciden.

② Mediante el decalaje básico se obtiene el sistema de origen básico (BNS) con el origen de paletas.

③ Mediante el decalaje de origen ajustable G54 o G55 se establece el "sistema de origen ajustable" (ENS) para la pieza 1 o la pieza 2.

④ Mediante la transformación de coordenadas programables se origina el sistema de coordenadas de pieza (WKS).

2.2 Fundamentos de la programación CN

Nota

La directriz para la programación CN es la norma DIN 66025.

2.2.1 Denominación de un programa CN

Reglas

Al crear cada programa CN, debe recibir un nombre (identificador). El nombre de programa puede elegirse libremente de acuerdo con las siguientes reglas:

- Caracteres permitidos:
  - Letras: A ... Z, a ... z
  - Números: 0 ... 9
  - Carácter de subrayado: _
- Los dos primeros caracteres deberían ser dos letras o un carácter de subrayado seguido de una letra.

  Nota

  Si se cumple esta condición, un programa CN puede llamarse como subprograma desde otro programa con solo indicar su nombre. Si, por el contrario, el nombre de programa empieza con números, la llamada de subprograma solo es posible mediante la instrucción CALL.

- Longitud máxima: 24 caracteres

Nota

Mayúsculas/minúsculas

En el lenguaje CN de SINUMERIK **no** se diferencia entre mayúsculas y minúsculas.

Nota

Nombres de programa no permitidos

Para evitar problemas con aplicaciones Windows, **no** deben utilizarse los siguientes nombres de programa:

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

Encontrará más información sobre limitaciones en "Nombres (Página 392)".

Ampliaciones internas del control

El nombre de programa asignado durante la creación del programa se amplía en el interior del control con un prefijo y una extensión:

- Prefijo: _N_
- Extensión:
  - Programas principales: _MPF
  - Subprogramas: _SPF

Ficheros en formato de cinta perforada

Los ficheros de programa creados externamente que deban cargarse a través del puerto V.24 deben estar en formato de cinta perforada.

Para los nombres de fichero en formato de cinta perforada se aplican las siguientes reglas adicionales:

- Primer carácter: %
- Para finalizar, una identificación de fichero de cuatro caracteres: _xxx

Ejemplos:

- %_N_EJE 123_MPF
- %Brida3_MPF

2.2.2 Estructura y contenido de un programa CN

2.2.2.1 Secuencias y componentes de secuencia

Secuencias

Un programa CN consta de una serie de secuencias CN. Cada secuencia contiene los datos para ejecutar un paso de trabajo en el mecanizado de una pieza.

Componentes de secuencia

Las secuencias CN están formadas por los siguientes componentes:

- Comandos (instrucciones) según DIN 66025
- Elementos del lenguaje de alto nivel CN

Comandos según DIN 66025

Los comandos según DIN 66025 se componen de un carácter de dirección y una cifra o una sucesión de cifras que simbolizan un valor aritmético.

Carácter de dirección (dirección)

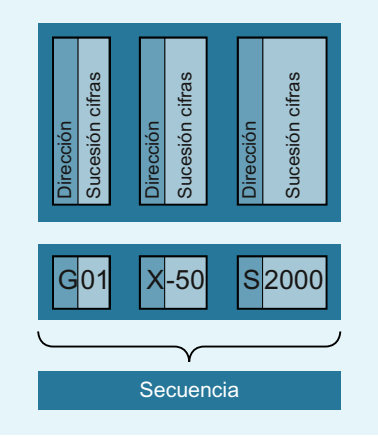
El carácter de dirección (por lo general una letra) define el significado del comando.

Ejemplos:

Carácter de dirección	Descripción
G	Comando G (condición de desplazamiento)
X	Información de recorrido para el eje X
S	Velocidad de giro del cabezal

Sucesión de cifras

La sucesión de cifras es el valor asignado al carácter de dirección. La sucesión de cifras puede llevar signo (+ ó -) y coma (punto) decimal; el signo tiene que encontrarse siempre entre la letra de dirección y la sucesión de cifras. No hace falta escribir los signos positivos (+) ni los ceros (0) a la izquierda.



Elementos del lenguaje de alto nivel CN

Dado que la secuencia de comandos según DIN 66025 ya no es suficiente para programar los complejos procesos de mecanizado en las máquinas herramienta actuales, se ha completado con los elementos del lenguaje de alto nivel CN.

Así encontramos, entre otros:

- Comandos del lenguaje de alto nivel CN
  A diferencia de los comandos según DIN 66025, los comandos del lenguaje de alto nivel CN están formados por varias letras de dirección, p. ej.:
  - `OVR` para la corrección de velocidad (Override)
  - `SPOS` para posicionado de cabezales
- Identificadores (nombres definidos) para:
  - Variables del sistema
  - Variables definidas por el usuario
  - Subprogramas
  - Palabras reservadas
  - Metas (etiquetas) de salto
  - específicas de canal

  Nota

  Un identificador debe ser unívoco y no puede utilizarse para diferentes objetos.
- Operadores de comparación
- Operadores lógicos
- Funciones de cálculo
- Estructuras de control

Efecto de los comandos

Los comandos pueden actuar de forma modal o secuencial:

- Modal
  Los comandos modalmente activos mantienen su validez con el valor programado (en todas las secuencias sucesivas) hasta que:
  - se programe un nuevo valor bajo el mismo comando.
  - se programe un comando que anula el efecto del comando válido hasta ahora.
- Secuencia a secuencia
  Los comandos secuencialmente activos sólo son válidos para la secuencia en la que se programan:

Fin del programa

La última secuencia en el orden de ejecución contiene una palabra especial para el fin del programa: `M2`, `M17` o `M30`.

2.2.2.2 Normas de secuencia

Principio de secuencia

Las secuencias CN pueden identificarse al principio con números de secuencia. Éstos están formados por el carácter "N" y un número entero positivo, p. ej.:
`N40 ...`

El orden de los números de secuencia no es fijo, aunque se aconseja utilizar números en orden ascendente.

Nota

Los números de secuencia dentro de un mismo programa deben ser unívocos para conseguir un resultado también unívoco cuando se efectúa una búsqueda de secuencia.

Fin de secuencia

Las secuencias terminan con el carácter "LF" (LINE FEED = nueva línea).

Nota

No es necesario escribir el carácter "LF". Se crea automáticamente al cambiar de línea.

Longitud de la secuencia

Una secuencia puede contener como máx. **512 caracteres** (incluyendo los comentarios y el carácter de fin de secuencia "LF").

Nota

En general, en la ventana de secuencia actual se visualizan tres secuencias, cada una de ellas con un máximo de 66 caracteres. Los comentarios se visualizan también. Los avisos se visualizan en la ventana de avisos.

Orden de las instrucciones

Con el fin de conseguir una estructura de secuencia clara, las instrucciones de una secuencia deben disponerse en el siguiente orden:
`N... G... X... Y... Z... F... S... T... D... M... H...`

Dirección	Descripción
`N`	Dirección del número de secuencia
`G`	Condición de desplazamiento
`X,Y,Z`	Información de recorrido
`F`	Avance
`S`	Velocidad de giro

T	Herramienta
D	Número de corrección de herramienta
M	Función adicional
H	Función auxiliar

Nota

Algunas direcciones pueden utilizarse varias veces dentro de la misma secuencia, p. ej.:

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

2.2.2.3 Asignaciones de valores

Se pueden asignar valores a las direcciones. Se aplican las siguientes reglas:

- Debe escribirse el carácter "=" entre la dirección y el valor cuando:
  - La dirección consta de más de una letra.
  - El valor consta de más de una constante

  Puede omitirse el carácter "=" cuando la dirección consta de una sola letra y el valor está formado por una única constante.
- Se permiten los signos.
- Se admiten los caracteres separadores tras la letra de dirección.

Ejemplos:

X10	Asignación del valor (10) a la dirección X, "=" no necesario
X1=10	Asignación del valor (10) a una dirección (X) con dirección numérica ampliada (1), "=" es necesario
X=10*(5+SIN(37.5))	Asignación del valor mediante una expresión numérica, "=" es necesario

Nota

Para diferenciar la dirección con extensión numérica de una letra de dirección con valor debe aparecer siempre uno de los signos especiales "=", "(", "[", ")", "]", "," o bien un operador después de las extensiones numéricas.

2.2.2.4 Comentarios

Para aumentar la comprensibilidad de un programa CN es posible añadir comentarios a las secuencias CN.

Los comentarios van al final de la secuencia separados del resto de la secuencia CN mediante el carácter separador (";").

Ejemplo 1:

Código de programa	Comentarios
N10 G1 F100 X10 Y20	; Comentarios para la aclaración de la secuencia CN

Ejemplo 2:

Código de programa	Comentarios
N10	; Empresa G&S, Pedido n.° 12A71
N20	; Programa creado por H. Müller, Dep. TV 4, el 21.11.94
N50	; Pieza n.° 12, caja para bomba sumergible tipo TP23A

Nota

Los comentarios se almacenan en la memoria y se visualizan en la ventana de secuencia actual durante la ejecución del programa.

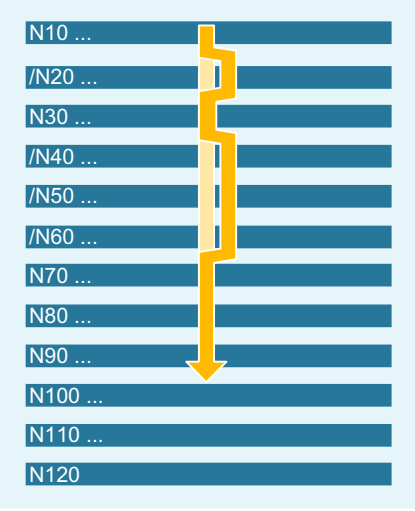
2.2.2.5 Omisión de secuencias

Las secuencias CN que no se deben ejecutar en cada pasada de programa se pueden ocultar para la ejecución. Esta función puede utilizarse, p. ej., al probar nuevos programas.

Secuencias opcionales

Las secuencias que se han de omitir llevan el carácter "/" (barra inclinada) antes del número de secuencia. Se pueden omitir también varias secuencias sucesivas. Las instrucciones escritas en las secuencias opcionales no son ejecutadas y el programa continúa con la primera secuencia que siga y que no esté marcada como omisible.

Ejemplo:



Código de programa	Comentarios
N10 …	; Se ejecuta
/N20 …	; No se ejecuta
N30 …	; Se ejecuta
/N40 …	; No se ejecuta
/N50 …	; No se ejecuta
/N60 …	; No se ejecuta
N70 …	; Se ejecuta
...	

Niveles de omisión

Es posible asignar a las secuencias niveles de omisión (máx. 10), que pueden activarse por medio de la interfaz de usuario o el programa de usuario del PLC.

La asignación se efectúa en el programa CN anteponiendo una barra inclinada seguida del número del nivel de omisión. Para cada secuencia solo puede definirse un nivel de omisión.

Ejemplo:

Código de programa	Comentarios
/ ...	; La secuencia se omite (1.er nivel de omisión)
/0 ...	; La secuencia se omite (1.er nivel de omisión)
/1 N010...	; La secuencia se omite (2.° nivel de omisión)
/2 N020...	; La secuencia se omite (3.er nivel de omisión)
...	
/7 N100...	; La secuencia se omite (8.° nivel de omisión)
/8 N080...	; La secuencia se omite (9.° nivel de omisión)
/9 N090...	; La secuencia se omite (10.° nivel de omisión)

Nota

Solamente se pueden modificar los niveles de omisión cuando el control está en el estado STOP/Reset.

Nota

La cantidad de niveles de omisión utilizables depende de un dato de máquina de visualización.

Nota

La omisión de secuencias también está activa en caso de búsqueda de secuencia.

Nota

También es posible crear programas de ejecución variable utilizando variables de sistema y de usuario para saltos condicionados.

2.3 Creación de un programa CN

2.3.1 Procedimiento básico

Generalmente, la creación del programa de pieza, es decir, el traslado de las distintas operaciones al lenguaje CN, sólo es una pequeña parte del trabajo de programación.

Antes de comenzar con la programación se debe de realizar de forma prioritaria una planificación y preparación de los procesos de trabajo. Cuanto más precisa sea la preparación de cómo se debe de estructurar el programa de pieza, más rápida y sencilla será la propia tarea de programación; esto también contribuye a que la estructura del programa sea clara y legible incluso para otros programadores. Los programas estructurados y legibles facilitan además la implementación de futuras modificaciones.

Dado que todas las piezas no son idénticas, no es conveniente crear todos los programas utilizando exactamente el mismo método. Sin embargo, para la mayoría de los casos resultará útil el siguiente procedimiento:

Procedimiento

1. **Preparar el plano de la pieza**
   - Definir el origen de la pieza
   - Marcar el sistema de coordenadas
   - Calcular los eventuales coordenadas que faltan
2. **Definir el orden de las operaciones de mecanizado**
   - ¿Qué herramientas se utilizan cuándo y para mecanizar qué contorno?
   - ¿En qué orden se fabrican los elementos individuales de la pieza?
   - ¿Qué elementos individuales se repiten (ev. también girados) y se deberían guardar en un subprograma?
   - ¿Hay en otros programas de pieza o subprogramas contornos de pieza que puedan reutilizarse para la pieza actual?
   - ¿Dónde son útiles o necesarios el decalaje de origen, la rotación, la simetría especular o el escalado (concepto frame)?

3. **Definir el plan de trabajo**
   Definir para cada proceso las distintas condiciones tecnológicas, p.ej.:
   - Desplazamientos en rápido para el posicionamiento
   - Cambio de herramienta
   - Definición del plano de mecanizado
   - Retirada para medida de comprobación
   - Cabezal, conectar/desconectar refrigerante
   - Llamada de datos de herramienta
   - Penetración
   - Corrección de trayectoria
   - Posicionamiento en el contorno
   - Retirada del contorno
   - etc.
4. **Traducir cada uno de las opciones al lenguaje de programación**
   - Se debe de definir cada uno de los pasos como una (o varias) secuencia de control numérico.
5. **Agrupar en un programa de pieza los distintos pasos**

2.3.2 Caracteres disponibles

Para la creación de programas CN se dispone del siguiente juego de caracteres:

- Letras mayúsculas:
  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
- Letras minúsculas:
  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
- Números:
  0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Caracteres especiales:
  Ver tabla siguiente

Caracteres especiales	Descripción
%	Carácter de inicio de programa (sólo para editar programas en un PC externo)
(	Paréntesis para parámetros o expresiones
)	Paréntesis para parámetros o expresiones
[	Paréntesis para direcciones o índices de matriz
]	Paréntesis para direcciones o índices de matriz
<	Menor que
>	Mayor que

Caracteres especiales	Descripción
:	Secuencia principal, definición de lábel, operador de concatenación
=	Asignación, equivalencia
/	División, supresión de secuencia
*	Multiplicación
+	Suma
-	Resta, signo negativo
"	Comillas, identificador para cadena de caracteres
'	Apóstrofe, identificador para valores numéricos especiales: hexadecimal, binario
$	Identificación de variable propia del sistema
s_	Subrayado, asociado a letras
?	Reservado
!	Reservado
.	Punto decimal
,	Coma, separador de parámetros
;	Inicio de comentario
&	Carácter para formato (mismo efecto que espacio en blanco)
LF	Fin de secuencia
Tabulador	Separador
Espacio en blanco	Separador, espacio

Nota

¡No confundir la letra "O" con el número "0"!

Nota

No se diferencia entre letras mayúsculas y minúsculas (excepción: llamada a herramienta).

Nota

Los caracteres no representables se tratan como si fueran espacios en blanco.

2.3.3 Cabecera de programa

Las secuencias CN antepuestas a las secuencias de desplazamiento propiamente dichas para la creación del contorno de pieza se denominan cabeceras de programa.

La cabecera de programa contiene información e instrucciones relativas a:

- Cambio de herramienta
- Correcciones de herramientas
- Giro del cabezal

- Ajuste de avance
- Ajustes geométricos (decalaje de origen, elección del plano de trabajo)

Cabecera de programa en el torneado

El siguiente ejemplo muestra cuál es la estructura típica de la cabecera de un programa CN para torneado:

Código de programa	**Comentarios**
N10 G0 G153 X200 Z500 T0 D0	; Retirada del portaherramientas antes de orientar el revólver de herramientas.
N20 T5	; Meter herramienta 5.
N30 D1	; Activar juego de datos de filos de la herramienta.
N40 G96 S300 LIMS=3000 M4 M8	; Velocidad de corte constante (Vc) = 300 m/min, limitación de velocidad = 3000 r/min, sentido de giro a la izquierda, refrigeración activada.
N50 DIAMON	; El eje X se programa en el diámetro.
N60 G54 G18 G0 X82 Z0.2	; Llamar a decalaje de origen y plano de trabajo, aproximar a posición inicial.
...	

Cabecera de programa en el fresado

El siguiente ejemplo muestra cuál es la estructura típica de la cabecera de un programa CN para fresado:

Código de programa	**Comentarios**
N10 T="SF12"	; Alternativa: T123
N20 M6	; Activar cambio de herramienta
N30 D1	; Activar juego de datos de filos de la herramienta
N40 G54 G17	; Decalaje de origen y plano de trabajo
N50 G0 X0 Y0 Z2 S2000 M3 M8	; Movimiento de aproximación a la pieza, cabezal y refrigerante activados
...	

Si se trabaja con orientación de herramienta/transformaciones de coordenadas, deben borrase las transformaciones que aún pueda haber activas al principio del programa:

Código de programa	**Comentarios**
N10 CYCLE800()	; Reseteo del plano orientado
N20 TRAFOOF	; Reseteo de TRAORI, TRANSMIT, TRACYL...
...	

2.3.4 Ejemplos de programa

2.3.4.1 Ejemplo 1: primeros pasos de programación

El ejemplo de programación 1 sirve para llevar a cabo y comprobar los primeros pasos de programación.

Procedimiento

1. Crear nuevo programa de pieza (nombre)
2. Editar programa de pieza
3. Seleccionar programa de pieza
4. Activar secuencia individual
5. Iniciar programa de pieza

Nota

Para que el programa pueda ejecutarse en la máquina, los datos de máquina deben estar ajustados como corresponda (→ ver las indicaciones del fabricante de la máquina).

Nota

Al probar un programa se pueden producir alarmas que deban resetearse primero.

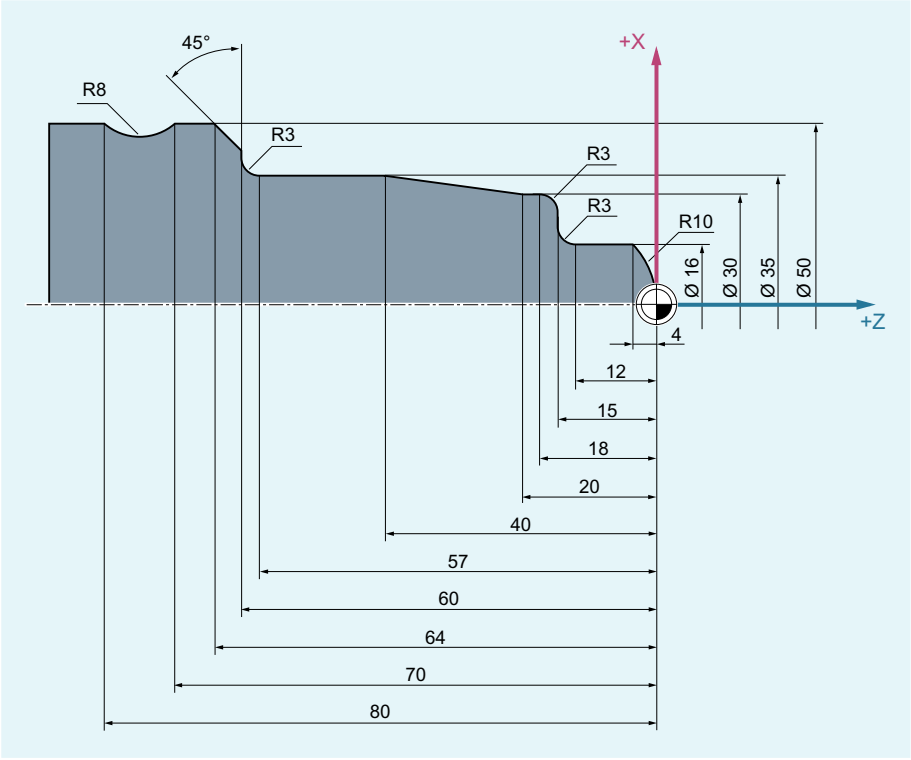
Programa CN

Código de programa	**Comentarios**
N10 MSG("ESTE ES MI PRIMER PROGRAMA CN")	; Emisión del aviso "ESTE ES MI PRIMER PROGRAMA CN" en la línea de alarmas
N20 F200 S900 T1 D2 M3	; Avance, cabezal, herramienta, corrección de herramienta, cabezal a derechas
N30 G0 X100 Y100	; Desplazamiento rápido a la posición
N40 G1 X150	; Rectángulo con avance, recta en X
N50 Y120	; Recta en Y
N60 X100	; Recta en X
N70 Y100	; Recta en Y
N80 G0 X0 Y0	; Retorno con desplazamiento rápido
N100 M30	; Fin de secuencia

2.3.4.2 Ejemplo 2: programa CN para torneado

El ejemplo de programa 2 está enfocado al mecanizado de una pieza en un torno. Incluye la programación por radio y la corrección del radio de herramienta.

Plano acotado de la pieza



Programa CN

Código de programa	Comentarios
N5 G0 G53 X280 Z380 D0	; Punto inicial
N10 TRANS X0 Z250	; Decalaje de origen
N15 LIMS=4000	; Limitación de la velocidad de giro (G96)
N20 G96 S250 M3	; Seleccionar velocidad de corte constante
N25 G90 T1 D1 M8	; Selección de la herramienta y la corrección
N30 G0 G42 X-1.5 Z1	; Posicionar herramienta con corrección del radio de herramienta
N35 G1 X0 Z0 F0.25	
N40 G3 X16 Z-4 I0 K-10	; Tornear radio 10
N45 G1 Z-12	
N50 G2 X22 Z-15 CR=3	; Tornear radio 3
N55 G1 X24	
N60 G3 X30 Z-18 I0 K-3	; Tornear radio 3
N65 G1 Z-20	
N70 X35 Z-40	
N75 Z-57	
N80 G2 X41 Z-60 CR=3	; Tornear radio 3
N85 G1 X46	

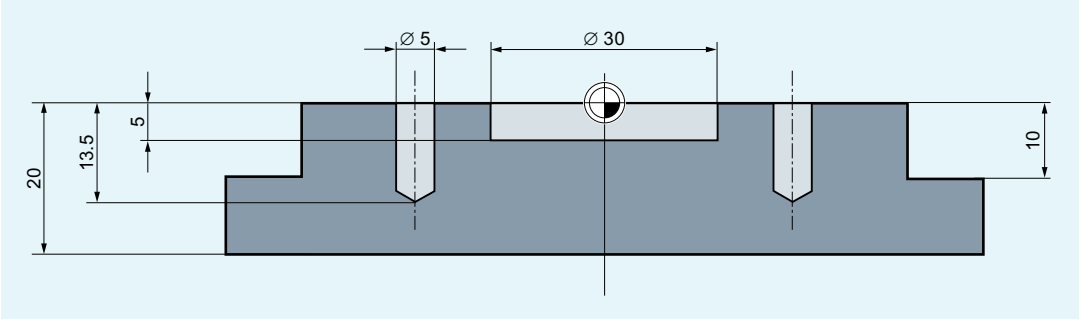
Código de programa	**Comentarios**
N90 X52 Z-63	
N95 G0 G40 G97 X100 Z50 M9	; Desactivar corrección del radio de herramienta y desplazar a punto de cambio de herramienta
N100 T2 D2	; Llamada de herramienta y selección de corrección
N105 G96 S210 M3	; Seleccionar velocidad de corte constante
N110 G0 G42 X50 Z-60 M8	; Posicionar herramienta con corrección del radio de herramienta
N115 G1 Z-70 F0.12	; Tornear diámetro 50
N120 G2 X50 Z-80 I6.245 K-5	; Tornear radio 8
N125 G0 G40 X100 Z50 M9	; Retirar la herramienta y desactivar la corrección del radio de herramienta
N130 G0 G53 X280 Z380 D0 M5	; Desplazamiento a punto de cambio de herramienta
N135 M30	; Fin de programa

2.3.4.3 Ejemplo 3: programa CN para fresado

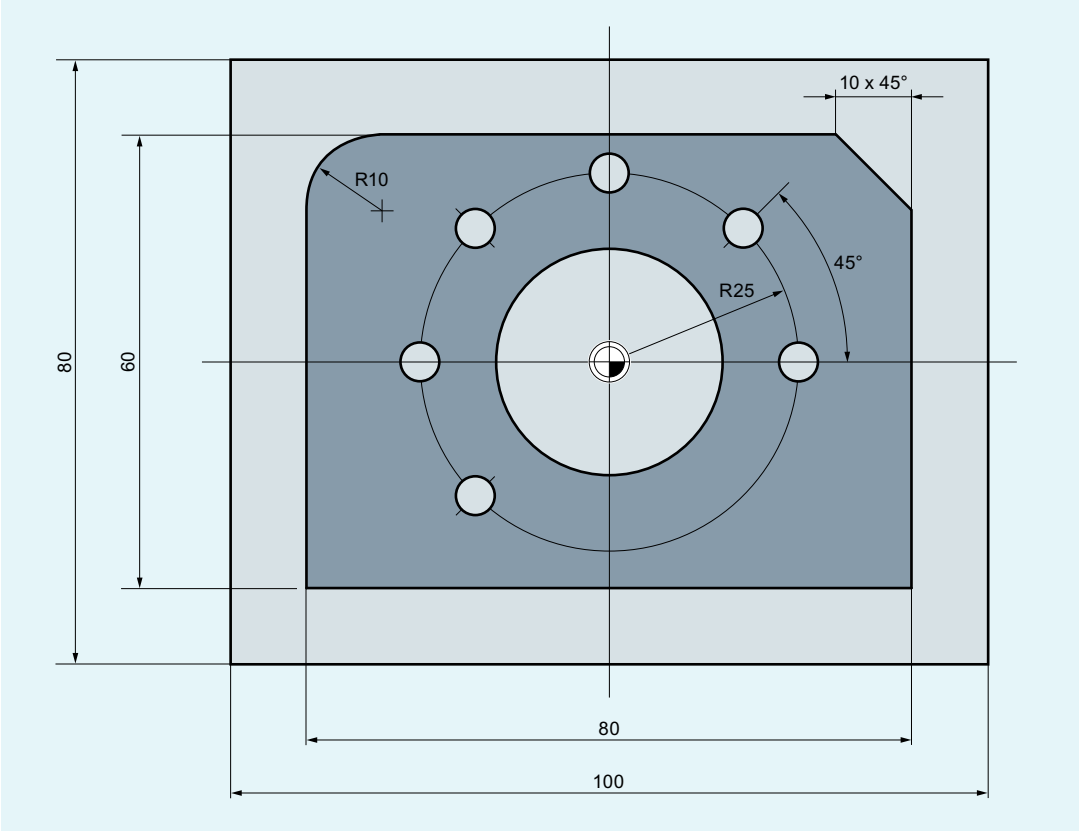
El ejemplo de programa 3 está enfocado al mecanizado de una pieza en una fresadora vertical. Incluye tanto fresado superficial y lateral como taladrado.

Plano acotado de la pieza

Vista lateral



Vista en planta



Programa CN

Código de programa	Comentarios
N10 T="PF60"	; Preselección de la herramienta con el nombre PF60.
N20 M6	; Carga de la herramienta en el cabezal.
N30 S2000 M3 M8	; Velocidad de giro, sentido de giro, refrigeración activados.
N40 G90 G64 G54 G17 G0 X-72 Y-72	; Ajuste básico de la geometría y aproximación al punto inicial.
N50 G0 Z2	; Eje Z a la distancia de seguridad.
N60 G450 CFTCP	; Comportamiento con G41/G42 activo.
N70 G1 Z-10 F3000	; Fresa a la profundidad de penetración con avance = 3000 mm/min.
N80 G1 G41 X-40	; Activación de la corrección del radio de fresa.
N90 G1 X-40 Y30 RND=10 F1200	; Desplazamiento en el contorno con avance = 1200 mm/min.
N100 G1 X40 Y30 CHR=10	
N110 G1 X40 Y-30	
N120 G1 X-41 Y-30	

Código de programa	**Comentarios**
N130 G1 G40 Y-72 F3000	; Cancelación de la corrección del radio de fresa.
N140 G0 Z200 M5 M9	; Elevación de fresa, cabezal y refrigeración desactivados.
N150 T="SF10"	; Preselección de la herramienta con el nombre SF10.
N160 M6	; Carga de la herramienta en el cabezal.
N170 S2800 M3 M8	; Velocidad de giro, sentido de giro, refrigeración activados.
N180 G90 G64 G54 G17 G0 X0 Y0	; Ajuste básico para la geometría y aproximación al punto inicial.
N190 G0 Z2	
N200 POCKET4(2,0,1,-5,15,0,0,0,0,0,800,1300,0,21,5,,,2,0.5)	; Llamada al ciclo de fresado de caja.
N210 G0 Z200 M5 M9	; Elevación de fresa, cabezal y refrigeración desactivados.
N220 T="ZB6"	; Llamada a broca de puntear 6 mm.
N230 M6	
N240 S5000 M3 M8	
N250 G90 G60 G54 G17 X25 Y0	; Parada precisa G60 debido a posicionamiento exacto.
N260 G0 Z2	
N270 MCALL CYCLE82(2,0,1,-2.6,,0)	; Llamada modal del ciclo de taladrado.
N280 POSITION:	; Meta de salto para repetición.
N290 HOLES2(0,0,25,0,45,6)	; Patrón de posiciones para figura de taladro.
N300 ENDLABEL:	; Identificación final para repetición.
N310 MCALL	; Reseteo de la llamada modal.
N320 G0 Z200 M5 M9	
N330 T="SPB5"	; Llamada a broca helicoidal 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)	; Llamada modal del ciclo de taladrado.
N380 REPEAT POSITION	; Repetición de la descripción de posición desde el punteado.
N390 MCALL	; Reseteo del ciclo de taladrado.
N400 G0 Z200 M5 M9	
N410 M30	; Fin del programa.

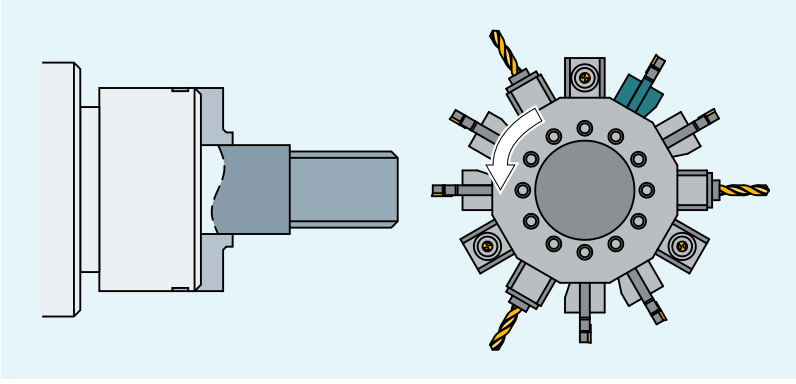
2.4 Cambio de herramienta

Tipo de cambio de herramienta

Nota

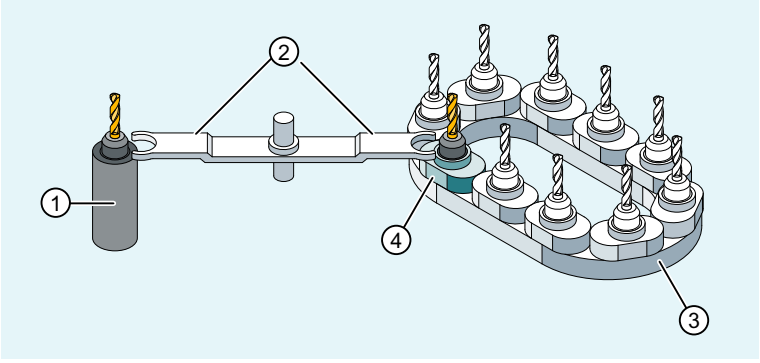
El tipo de cambio de herramienta lo fija el fabricante de la máquina durante la puesta en marcha.

Cambio de herramienta en tornos con torreta revólver



En las torretas revólver de los tornos, el cambio de herramienta, es decir, la búsqueda y el cambio de herramienta, se llama únicamente con la orden T.

En fresadoras con cargadores de cadena, almacenes de herramientas con disco giratorio o almacenes de tambor.



① Cabezal
② Pinzas
③ Almacén (aquí: cargador de cadena)
④ Punto de cambio para cabezal

En los almacenes de cadena, de herramientas con disco giratorio o de tambor, el cambio de herramienta se realiza normalmente en dos pasos:

1. Con la orden T se busca la herramienta en el almacén.
2. A continuación se realiza el cambio de las herramientas en el cabezal con el comando M.

Programación del cambio de herramienta con gestión de herramientas activa o no

La programación del cambio de herramienta se diferencia según la gestión de herramientas esté activa o no. Se describen por tanto las dos variantes:

→ Cambio de herramienta con gestión de herramientas activa (Página 61)

→ Cambio de herramienta con gestión de herramientas no activa (Página 65)

Programación del plano de trabajo

Con el cambio de herramienta debe programarse también el correspondiente plano de trabajo (Página 33) (posición básica: G18). De esta forma se garantiza que la corrección longitudinal de la herramienta esté asignada al eje correcto.

Activación de la corrección de herramienta

El cambio de herramienta activa los valores de corrección de herramienta memorizados en un número D (Página 81).

2.4.1 Cambio de herramienta con gestión de herramientas activa

Gestión de herramientas

La función "Gestión de herramientas" asegura que en la máquina siempre está disponible la herramienta correcta en el instante y lugar adecuados, y que los datos asignados a una herramienta son los actuales. Además posibilita un rápido cambio de herramienta, evita piezas defectuosas gracias a la vigilancia del tiempo de uso de la herramienta, así como paradas de máquina al tener en cuenta las herramientas de repuesto.

Nombres de herramienta

En una máquina herramienta con gestión de herramientas activa, las herramientas se deben dotar de nombres y números que permitan identificarlas de manera inequívoca (p. ej., "Broca", "3").

Entonces puede realizarse la llamada de herramienta por medio del nombre de herramienta, p. ej.:

```
T="Broca"
```

Nota

El nombre de herramienta no puede contener caracteres especiales.

2.4.1.1 Cambio de herramienta con WZV activa con la función T=puesto

Al programar a función T=puesto se realiza un cambio directo de herramienta.

Aplicación

En tornos con torreta revólver.

Sintaxis

Seleccionar herramienta

```
T=<N.°>
T=<Nombre>
T<n>=<N.°>
T<n>=<Nombre>
```

Deseleccionar herramienta

```
T0
```

Descripción

T=:	Comando para el cambio de herramienta y la activación de la corrección de herramienta Como datos se admiten:	
	<N.°>:	Número del puesto de almacén
	<Nombre>:	Nombre de la herramienta **Nota:** Al programar un nombre de herramienta se debe prestar atención a la correcta notación (mayúsculas/minúsculas).
<n>:	Número de cabezal como dirección extendida **Nota:** La posibilidad de programar un número de cabezal como dirección extendida depende de la configuración de la máquina (→ ver las indicaciones del fabricante de la máquina).	
T0:	Deselección de herramienta (puesto de almacén no ocupado)	

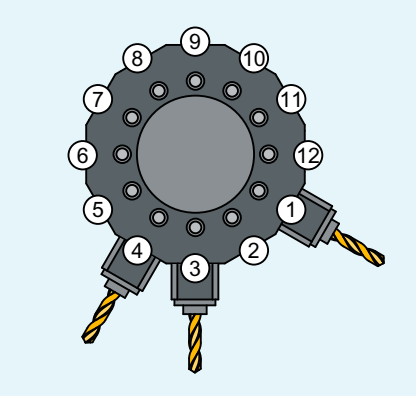
Nota

Si, en un almacén de herramientas, el puesto de almacén seleccionado no está ocupado, el comando de herramienta actúa como T0. La selección del puesto de almacén sin ocupar se puede utilizar para posicionar el puesto vacío.

Ejemplo

Un almacén de torreta revólver tiene los puestos 1 a 12 con la siguiente ocupación de herramientas:

Puesto	Herramienta	Número T interno	Estado
1	Broca, n.º duplo = 1	T15	Bloqueado
2	No ocupado		
3	Broca, n.º duplo = 2	T10	Disponible
4	Broca, n.º duplo = 3	T1	Activo
5 ... 12	No ocupado		



① ... ⑫ Números de puesto de almacén

En el programa CN se ha programado la siguiente llamada de herramienta:

```
N10 T=1
```

La llamada se procesa como sigue:

1. Se considera el puesto de almacén 1 y se determina el identificador de la herramienta.
2. La gestión de herramientas detecta que esta herramienta está bloqueada y, por tanto, no está lista para ser utilizada.
3. Se inicia una búsqueda de herramienta de T="Broca" conforme a la estrategia de búsqueda ajustada:
   "Buscar herramienta activa; si no, utilizar el siguiente n.º duplo más alto".
4. Como herramienta lista para ser utilizada se encuentra:
   "Broca" n.º duplo 3 (en el puesto de almacén 4)
   De este modo concluye la selección de herramienta y se inicia el cambio de herramienta.

Nota

Con la estrategia de búsqueda "Tomar la primera herramienta libre del grupo" no está definido el orden de las herramientas. En tal caso se elige T10 **o** T1.

Con la estrategia de búsqueda "Tomar la primera herramienta con estado 'activo' del grupo", se carga T1.

2.4.1.2 Cambio de herramienta con WZV activa con M6

La herramienta se selecciona programando el comando T. La herramienta pasa a estar activa solo con M6 (incluida la corrección de herramienta).

Aplicación

En máquinas herramienta con cargadores de cadena, almacenes de herramientas con disco giratorio o almacenes de tambor.

Sintaxis

Seleccionar herramienta
```
T=<N.°>
T=<Nombre>
T<n>=<N.°>
T<n>=<Nombre>
```

Cambio de herramienta
```
M6
```

Deseleccionar herramienta
```
T0
```

Descripción

T=:	Dirección para la selección de herramientas Como datos se admiten:	
	<N.°>:	Número del puesto de almacén
	<Nombre>:	Nombre de la herramienta **Nota:** Al programar un nombre de herramienta se debe prestar atención a la correcta notación (mayúsculas/minúsculas).
<n>:	Número de cabezal como dirección extendida **Nota:** La posibilidad de programar un número de cabezal como dirección extendida depende de la configuración de la máquina (→ ver las indicaciones del fabricante de la máquina).	
M6:	Función M para el cambio de herramienta (según DIN 66025) Con M6 se activan la herramienta seleccionada (T…) y la corrección de herramienta (D...).	
T0:	Deselección de herramienta (puesto de almacén no ocupado)	

Ejemplo

Código de programa	Comentarios
N10 T=1 M6	; Carga de la herramienta del puesto de almacén 1.

Código de programa	Comentarios
N20 D1	; Activación de la corrección longitudinal de herramienta.
N30 G1 X10 ...	; Trabajo con herramienta T=1.
...	
N70 T="Broca"	; Preselección de herramienta con el nombre "Broca".
N80 ...	; Trabajo con herramienta T=1.
...	
N100 M6	; Carga de la broca.
N140 D1 G1 X10 ...	; Trabajo con broca.
...	

2.4.2 Cambio de herramienta con gestión de herramientas no activa

2.4.2.1 Cambio de herramienta con WZV no activa con número T

Al programar el número T se realiza un cambio directo de herramienta.

Aplicación

En tornos con torreta revólver.

Sintaxis

Seleccionar herramienta

```
T<N.°>
T=<N.°>
T<n>=<N.°>
```

Deseleccionar herramienta

```
T0
T0=<N.°>
```

Descripción

T:	Dirección para la selección de la herramienta incluido el cambio de herramienta y la activación de la corrección de herramienta	
<n>:	Número de cabezal como dirección extendida **Nota:** La posibilidad de programar un número de cabezal como dirección extendida depende de la configuración de la máquina (→ ver las indicaciones del fabricante de la máquina).	
<N.°>:	Número de la herramienta	
	Rango de valores:	0 ... 32000
T0:	Deselección de la herramienta activa	

Ejemplo

Código de programa	Comentarios
N10 T1 D1	; Carga de la herramienta T1 y activación de la corrección de herramienta D1.
...	
N70 T0	; Deseleccionar herramienta T1.
...	

2.4.2.2 Cambio de herramienta con WZV no activa con M6

La herramienta se selecciona programando el comando T. La herramienta pasa a estar activa solo con M6 (incluida la corrección de herramienta).

Aplicación

En máquinas herramienta con cargadores de cadena, almacenes de herramientas con disco giratorio o almacenes de tambor.

Sintaxis

Seleccionar herramienta
```
T<N.°>
T=<N.°>
T<n>=<N.°>
```

Cambio de herramienta
```
M6
```

Deseleccionar herramienta
```
T0
T0=<N.°>
```

Descripción

T:	Dirección para la selección de herramientas	
<n>:	Número de cabezal como dirección extendida **Nota:** La posibilidad de programar un número de cabezal como dirección extendida depende de la configuración de la máquina (→ ver las indicaciones del fabricante de la máquina).	
<N.°>:	Número de la herramienta	
	Rango de valores:	0 ... 32000
M6:	Función M para el cambio de herramienta (según DIN 66025) Con M6 se activan la herramienta seleccionada (T…) y la corrección de herramienta (D...).	
T0:	Deselección de la herramienta activa	

Ejemplo

Código de programa	Comentarios
N10 T1 M6	; Carga de la herramienta T1.
N20 D1	; Activación de la corrección longitudinal de herramienta.
N30 G1 X10 ...	; Trabajo con T1.
...	
N70 T5	; Preselección de herramienta T5.
N80 ...	; Trabajo con T1.
...	
N100 M6	; Carga de herramienta T5.
N110 D1 G1 X10 ...	; Trabajo con herramienta T5.
...	

2.4.3 Comportamiento en caso de programación T errónea

El comportamiento en caso de programación T incorrecta varía en función de la configuración de la máquina:

MD22562 TOOL_CHANGE_ERROR_MODE		
Bit	**Valor**	**Descripción**
7	0	Posición básica En la programación T se comprueba inmediatamente si el número T es conocido en el CN. De lo contrario se activa una alarma.
	1	El número T programado no se comprueba hasta que se haya realizado la selección D. Si el CN no conoce el número T, se emite una alarma con la selección D. Este comportamiento es deseable, p. ej., si la programación T debe producir también un posicionamiento y no es necesario que existan para este fin los datos de herramienta (almacén revólver).

2.5 Correcciones de herramientas

2.5.1 Contorno y trayectoria de herramienta programados

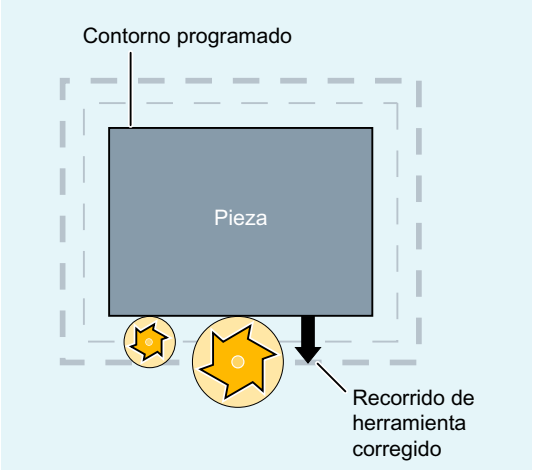
Las medidas de pieza se programan directamente (p. ej., según plano de fabricación). Por ello, al crear el programa de pieza no es necesario tener en cuenta datos de herramienta como el radio de la fresa, la posición del filo de la herramienta de tornear (herramienta izquierda o derecha), ni las longitudes de la herramienta.

El control realiza las correcciones de la trayectoria

Durante el mecanizado de la pieza, los desplazamientos de la herramienta deben de controlarse de acuerdo con la geometría de la herramienta, de forma que el contorno programado pueda mecanizarse con cualquiera de las herramientas utilizadas.

Para que el control pueda calcular los recorridos de herramienta, los datos de herramienta deben estar introducidos en la memoria de corrección de herramienta del control. A través del programa CN solo se llaman la herramienta (T...) y el bloque de datos de corrección (D...) necesarios.

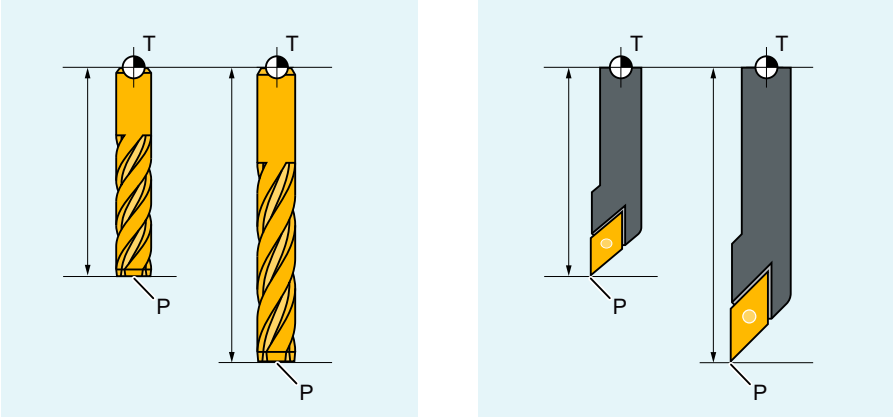
Durante la ejecución del programa, el control toma los datos de corrección de la memoria de corrección de herramienta y corrige la trayectoria atendiendo a la geometría de cada una de las herramientas utilizadas:



2.5.2 Corrección longitudinal de herramienta

Con la corrección de la longitud de herramienta se compensan las diferentes longitudes de las herramientas usadas.

La longitud de la herramienta es la distancia entre el punto de referencia del portaherramientas y la punta de la herramienta:



T Punto de referencia del portaherramientas

P Punta de la herramienta

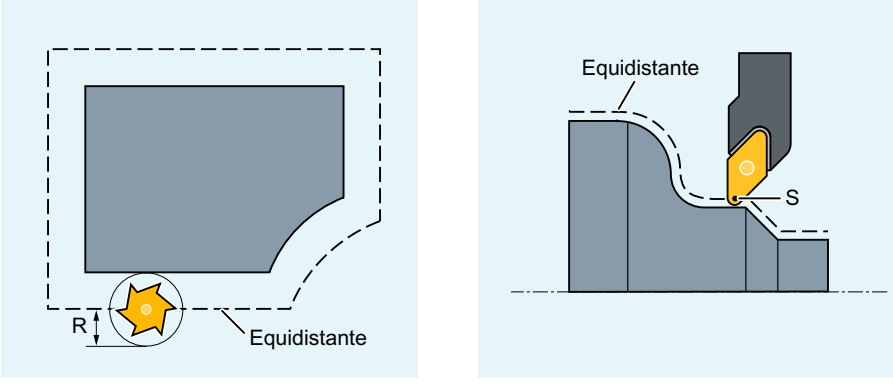
Esta longitud se mide y se introduce junto con los valores de desgaste especificables en la memoria de corrección de herramienta del control. A partir de estos valores el control recalcula los desplazamientos en la dirección de penetración.

Nota

El valor de corrección de la longitud de la herramienta depende de la orientación en el espacio de la herramienta.

2.5.3 Corrección del radio de herramienta

El contorno geométrico de la pieza y la trayectoria de la herramienta no coinciden. El centro de la fresa o el punto medio del corte deben realizar una trayectoria equidistante al contorno (trayectoria de centro de herramienta) en función del radio de la herramienta. Para ello, durante el procesamiento del programa, el control desplaza la trayectoria de centro de herramienta programada en función del radio de la herramienta activa (memoria de corrección de la herramienta), de manera que el filo de la herramienta se desplaza exactamente por el contorno programado.



R Radio de la herramienta

S Centro del filo

La corrección del radio de herramienta se describe con detalle en el capítulo "Correcciones del radio de herramienta (Página 257)".

Consulte también

Corrección de herramienta en 2 1/2 D (CUT2D, CUT2DD, CUT2DF, CUT2DFD) (Página 294)

2.5.4 Memoria de corrección de herramienta

En la memoria de corrección de herramienta del control deben estar presentes los siguientes datos para cada filo de herramienta:

- Tipo de herramienta
- Posición del filo
- Magnitudes geométricas de la herramienta (longitud, radio)

Estos datos se registran como parámetros de la herramienta (máx. 25). Los parámetros necesarios para cada herramienta dependen del tipo de herramienta. Los parámetros de herramienta no requeridos deben ocuparse con el valor "Cero" (que equivale a la preasignación del sistema).

Nota

Una vez introducidos los valores en la memoria de correcciones, serán considerados en el cálculo cada vez que se llame una herramienta.

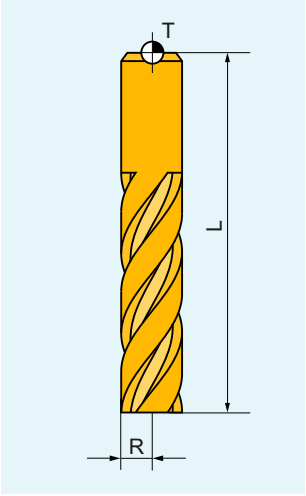
Tipo de herramienta

El tipo de herramienta (broca, fresa o herramientas de torneado) determina qué datos geométricos se necesitan y cómo se calculan.

Posición del filo

La posición del filo describe la posición de la punta de la herramienta en relación con el centro del filo. La posición del filo es necesaria, junto con el radio del filo, para calcular la corrección de radio de herramienta en herramientas de tornear (tipo de herramienta 5xx) (Página 77).

Magnitudes geométricas de la herramienta (longitud, radio)



T	Punto de referencia del portaherramientas
R	Radio de la herramienta
L	Longitud de herramienta

Las magnitudes geométricas de herramienta constan de varios componentes (geometría, desgaste). A partir de estos componentes, el control numérico calcula una dimensión resultante (p. ej., longitud total 1, radio total). Al activarse la corrección de herramienta, se aplica el resultado de la medida total correspondiente.

El tipo de herramienta y el plano de trabajo actual (G17/G18/G19) determinan la manera en que se calculan estos valores en los ejes.

2.5.5 Tipos de herramientas

2.5.5.1 Número de tipo de herramienta y grupos de herramientas

A cada tipo de herramienta se le asigna un número unívoco de 3 cifras. La primera cifra (centenas) sirve para asignar la herramienta a una de las siguientes tecnologías o grupos de herramientas:

Tipo de herramienta	Grupo de herramientas
1xy	Fresas (Página 72)
2xy	Brocas (Página 74)
3xy	reservado
4xy	Muelas rectificadoras (Página 75)
5xy	Herramientas de tornear (Página 77)
6xy	reservado
7xy	Herramientas especiales (Página 79)

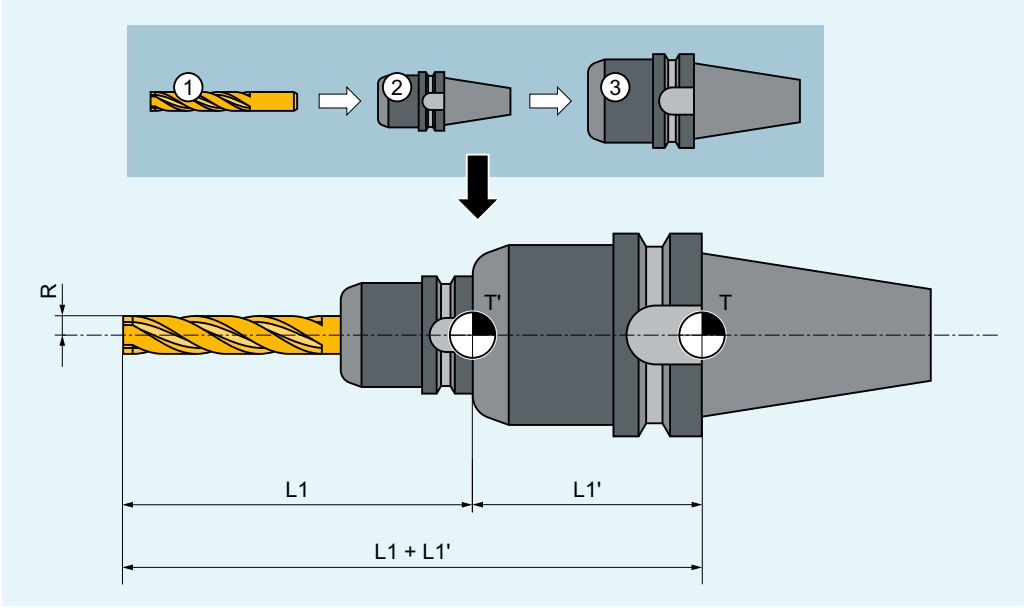
2.5.5.2 Fresas

Dentro del grupo "Fresas" existen los siguientes tipos de herramientas:

100	Fresa según CLDATA (Cutter Location Data)
110	Fresa esférica
111	Fresa cilíndrica para matricería
120	Fresa de mango sin redondeado de ángulos
121	Fresa de mango con redondeado de ángulos
130	Fresa con cabeza angular sin redondeado de ángulos
131	Fresa con cabeza angular con redondeado de ángulos
140	Fresa para planear
145	Fresa para roscar
150	Fresa de plaquitas
151	Sierra
155	Fresa troncocónica sin redondeado de ángulos
156	Fresa troncocónica con redondeado de ángulos
157	Fresa cónica para matricería
160	Fresa para roscar

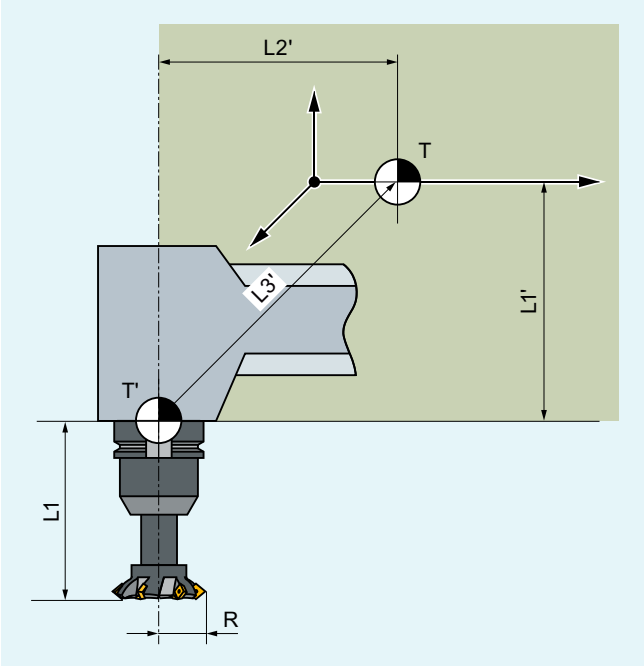
Parámetros de herramienta

Las siguientes figuras proporcionan una vista general de los parámetros de herramienta que se registran en la memoria de corrección para las fresas:



①	Herramienta
②	Portaherramientas
③	Adaptador de herramienta
T	Punto de referencia del adaptador (con la herramienta insertada = punto de referencia del portaherramientas)
T'	Punto de referencia del soporte de herramienta
L1	Geometría: Longitud 1
L1'	Cota del adaptador: Longitud 1
L1 + L1'	Longitud total L1
R	Radio

Parámetros de herramienta	Significado
$TC_DP1	Tipo de herramienta 1xy
$TC_DP3	Geometría: Longitud 1
$TC_DP6	Geometría: Radio
$TC_DP21	Cota del adaptador: Longitud 1
• Valores de desgaste según necesidades. • El resto de los valores deben ponerse a 0.	



T Punto de referencia del portaherramientas
T' Punto de referencia del soporte de herramienta
L1 Geometría: Longitud 1
R Radio de la herramienta
L1' Acotado básico: Longitud 1
L2' Acotado básico: Longitud 2
L3' Acotado básico: Longitud 3

Parámetros de herramienta	Significado
$TC_DP1	Tipo de herramienta
$TC_DP3	Geometría: Longitud 1
$TC_DP6	Geometría: Radio
$TC_DP21	Acotado básico: Longitud 1
$TC_DP22	Acotado básico: Longitud 2
$TC_DP23	Acotado básico: Longitud 3
• Valores de desgaste según necesidades. • El resto de los valores deben ponerse a 0.	

2.5.5.3 Broca

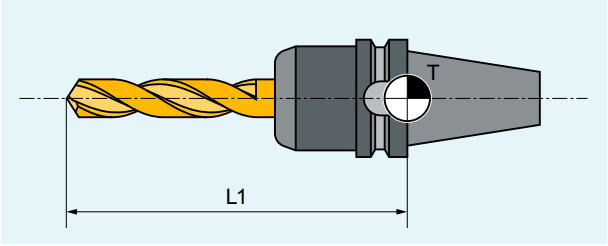
Dentro del grupo "Brocas" existen los siguientes tipos de herramientas:

N.°	Tipo de herramienta
200	Broca helicoidal
205	Broca maciza
210	Mandril

N.°	Tipo de herramienta
220	Broca de puntear
230	Avellanador cónico
231	Avellanador plano
240	Macho de roscar, rosca normal
241	Macho de roscar, rosca fina
242	Macho de roscar, rosca Withworth
250	Escariador

Parámetros de herramienta

La siguiente figura proporciona una vista general de los parámetros de herramienta que se registran en la memoria de corrección para las brocas:



T Punto de referencia del portaherramientas

L1 Longitud 1

Parámetros de herramienta	Significado
$TC_DP1	Tipo de herramienta
$TC_DP3	Geometría: Longitud 1
• Valores de desgaste según necesidades. • El resto de los valores deben ponerse a 0.	

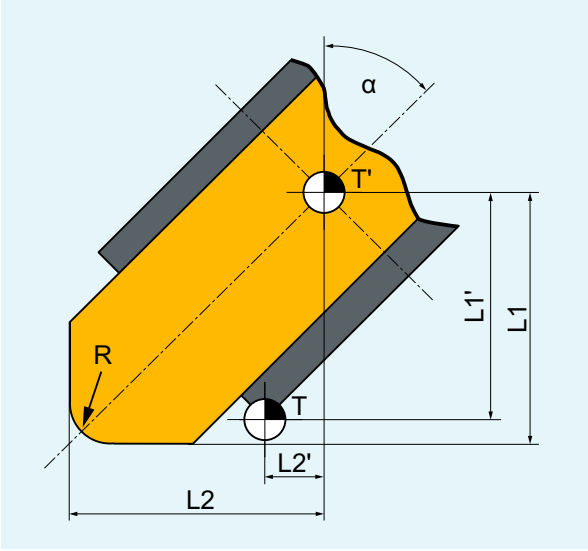
2.5.5.4 Muelas rectificadoras

Dentro del grupo "Muelas rectificadoras" existen los siguientes tipos de herramientas:

400	Muela para rectificado normal
401	Muela para rectificado normal con vigilancia
402	Muela para rectificado normal sin vigilancia ni acotado básico (WZV)
403	Muela para rectificado normal con vigilancia sin acotado básico para velocidad periférica de la muela SUG
410	Muela plana
411	Muela para rectificado plano (WZV) con vigilancia
412	Muela para rectificado plano (WZV) sin vigilancia
413	Muela para rectificado plano con vigilancia sin acotado básico para velocidad periférica de la muela SUG
490	Diamante

Parámetros de herramienta

La siguiente figura proporciona una vista general de los parámetros de herramienta que se registran en la memoria de corrección para las muelas rectificadoras:



T	Punto de referencia del portaherramientas
T'	Punto de referencia del soporte de herramienta
L1	Geometría: Longitud 1
L1'	Acotado básico: Longitud 1
L2	Geometría: Longitud 2
L2'	Acotado básico: Longitud 2
R	Radio
α	Ángulo para la muela inclinada

Parámetros específicos de filo	Descripción
$TC_DP1	Tipo de herramienta 4xy
$TC_DP2	Posición del filo
$TC_DP3	Geometría: Longitud 1
$TC_DP4	Geometría: Longitud 2
$TC_DP6	Radio
$TC_DP21	Acotado básico: Longitud 1
$TC_DP22	Acotado básico: Longitud 2
• Valores de desgaste según necesidades. • El resto de los valores deben ponerse a 0.	

Parámetros específicos de herramienta	Descripción
$TC_TPG1	N.º de cabezal
$TC_TPG2	Regla de concatenación[1)]
$TC_TPG3	Radio mínimo de la muela

Parámetros específicos de herramienta	Descripción
$TC_TPG4	Ancho mínimo de la muela
$TC_TPG5	Ancho actual de la muela
$TC_TPG6	Velocidad de rotación máxima
$TC_TPG7	Velocidad periférica máxima
$TC_TPG8	Ángulo para la muela inclinada
$TC_TPG9	Número de parámetro para el cálculo del radio
$TC_TPG_DRSPATH	Ruta de directorio al programa de diamantado
$TC_TPG_DRSPROG	Nombre del programa de diamantado

1) Las correcciones longitudinales de geometría, desgaste y acotado básico se pueden concatenar para las correcciones izquierda y derecha de la muela, respectivamente. Es decir: si se modifican las correcciones longitudinales para el filo izquierdo, también se introducen automáticamente para el filo derecho y viceversa.

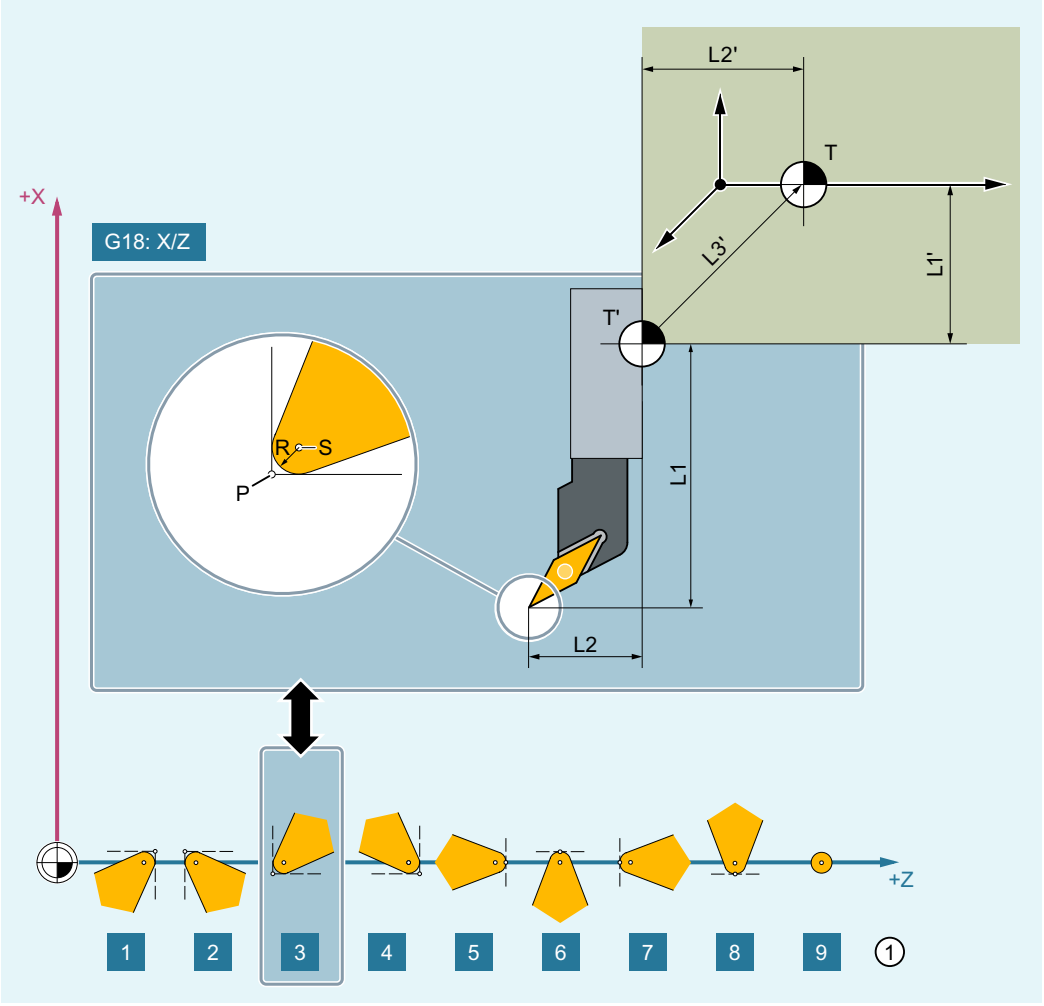
2.5.5.5 Cuchillas de tornear

Dentro del grupo "Herramientas de tornear" existen los siguientes tipos de herramientas:

500	Desbastadora
510	Cuchilla de acabado
520	Cuchilla de ranurar
530	Cuchilla de tronzar
540	Cuchilla de roscar
550	Cuchilla fungiforme/de perfilado (WZV)
560	Broca con plaquitas (ECOCUT)
580	Palpador con parámetro Posición del filo

Parámetros de herramienta

La siguiente figura proporciona una vista general de los parámetros de herramienta que se registran en la memoria de corrección para las herramientas de tornear:



①	Posición del filo (1 - 9) en mecanizado detrás del centro de giro
P	Punta de la herramienta
S	Centro del filo
R	Radio del filo
T	Punto de referencia del portaherramientas
T'	Punto de referencia del soporte de herramienta
L1	Geometría: Longitud 1
L2	Geometría: Longitud 2
L1'	Acotado básico: Longitud 1
L2'	Acotado básico: Longitud 2
L3'	Acotado básico: Longitud 3

Parámetros de herramienta	Descripción
$TC_DP1	Tipo de herramienta
$TC_DP2	Posición del filo
$TC_DP3	Geometría: Longitud 1
$TC_DP4	Geometría: Longitud 2
$TC_DP6	Geometría: Radio (radio del filo)
$TC_DP21	Acotado básico: Longitud 1
$TC_DP22	Acotado básico: Longitud 2
$TC_DP23	Acotado básico: Longitud 3
• Valores de desgaste según necesidades. • El resto de los valores deben ponerse a 0.	

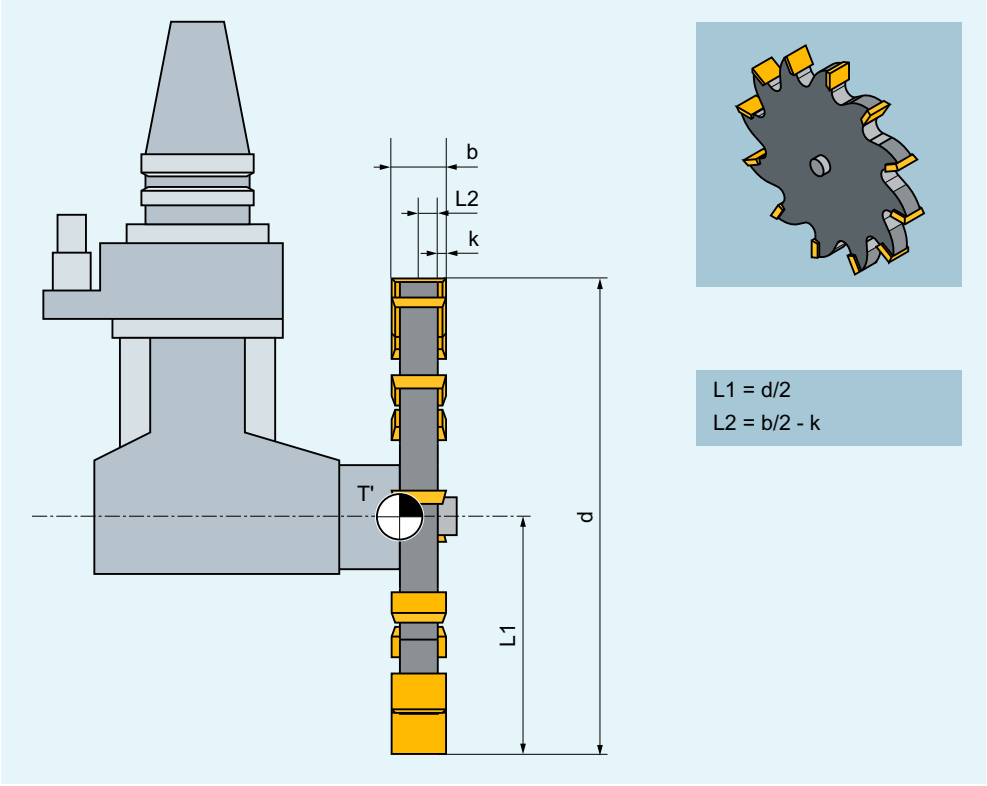
2.5.5.6 Herramientas especiales

Dentro del grupo "Herramientas especiales" existen los siguientes tipos de herramientas:

700	Sierra para ranurar
710	Palpador 3D
711	Palpador de bordes
712	Palpador monodireccional
713	Palpador en L
714	Palpador en estrella
725	Herramienta de calibrado
730	Tope
731	Pinola
732	Luna

Parámetros de herramienta

La siguiente figura proporciona una vista general de los parámetros de herramienta que se registran en la memoria de corrección para la herramienta especial "Sierra para ranurar":



T'	Punto de referencia del soporte de herramienta
L1	Geometría: Longitud 1
L2	Geometría: Longitud 2
d	Diámetro
b	Ancho de ranura
k	Saliente

Parámetros de herramienta	Significado
$TC_DP1	Tipo de herramienta
$TC_DP3	Geometría: Longitud 1
$TC_DP4	Geometría: Longitud 2
$TC_DP6	Diámetro
$TC_DP7	Ancho de ranura
$TC_DP8	Saliente
$TC_DP21	Acotado básico: Longitud 1
$TC_DP22	Acotado básico: Longitud 2
$TC_DP23	Acotado básico: Longitud 3
• Valores de desgaste según necesidades. • El resto de los valores deben ponerse a 0.	

2.5.6 Activación/desactivación de correcciones de herramientas (D, D0)

Es posible asignar diferentes bloques de datos de corrección de herramienta (p. ej., diferentes valores de corrección para los filos izquierdo y derecho de un útil de ranurar) a los filos 1 a 8 de una herramienta (con la gestión de herramientas activa 12).

Los datos de corrección (entre otros, los datos para la corrección de la longitud de herramienta) de un filo especial se activan llamando el número D. Al programar D0 se invalidan las correcciones para la herramienta.

Una corrección del radio de herramienta se tiene que activar adicionalmente con G41/G42.

Nota

Las correcciones de longitud de herramienta actúan cuando se ha programado el número D. Si no se programa ningún número D, al realizar un cambio de herramienta estará activo el ajuste estándar definido mediante el dato de máquina (→ ver las indicaciones del fabricante de la máquina).

Sintaxis

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

Descripción

D:	Dirección para activar un bloque de datos de corrección para la herramienta activa La corrección de la longitud de herramienta se retira con el primer desplazamiento programado del eje de corrección longitudinal correspondiente. **Atención:** Una corrección de la longitud de herramienta actúa también sin programación D si para el cambio de herramienta se ha configurado la activación automática de un filo de herramienta (→ ver las indicaciones del fabricante de la máquina).	
<Nr>:	El número D indica el bloque de datos de corrección de herramienta que se desea activar. El tipo de programación D depende de la configuración de la máquina (→ ver las indicaciones del fabricante de la máquina). Existen las siguientes posibilidades: • Número D = número de filo Para cada herramienta T<N.º> (sin gestión de herramientas) o T="Nombre" (con gestión de herramientas), existen números D del 1 al 12 (máx.). Estos números D están asignados directamente a los filos de las herramientas. Cada número D (= número de filo) tiene un bloque de datos de corrección ($TC_DPx[<t>,<d>]). • Libre elección de números D Los números D pueden asignarse libremente a los números de filo de una herramienta. El límite superior de los números D utilizables viene definido por un dato de máquina.	
	Rango de valores:	0 - 32000
D0:	Desactivación del bloque de datos de corrección para la herramienta activa	
G41:	Comando para activar la corrección de radio de herramienta con el sentido de mecanizado **a la izquierda** del contorno	
G42:	Comando para activar la corrección de radio de herramienta con el sentido de mecanizado **a la derecha** del contorno	
G40:	Comando para la desactivación de la corrección de radio de herramienta	

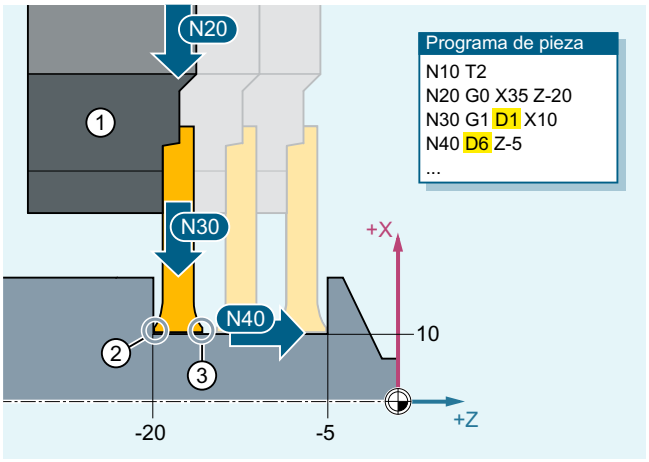
Encontrará **más información** sobre G40/G41/G42 en el capítulo "Correcciones del radio de herramienta (Página 257)".

Ejemplos

Ejemplo 1: cambio de herramienta con comando T (torneado)

Código de programa	Comentarios
N10 T1 D1	; Cargar herramienta T1 y activar el bloque de datos de corrección de herramienta D1 de T1.
N11 G0 X... Z...	; Las correcciones longitudinales se retiran.
N50 T4 D2	; Cambiar herramienta T4 y activar el bloque de datos de corrección de herramienta D2 de T4.
...	
N70 G0 Z... D1	; Activar otro filo D1 para la herramienta T4.

Ejemplo 2: diferentes valores de corrección para los filos izquierdo y derecho de un útil de ranurar



① Útil de torneado para ranurado (T2)
② Filo D1
③ Filo D6

Consulte también

Corrección del radio de herramienta (Página 69)

Información adicional

Modificación de los datos de corrección de herramienta

En el ajuste predeterminado, las modificaciones de los datos de corrección de herramienta solo serán válidas tras reprogramar T o D.

El siguiente dato de máquina establece la activación inmediata de los datos de corrección de herramienta introducidos.

MD9440 $MM_ACTIVATE_SEL_USER

⚠ ADVERTENCIA

Peligro de colisión

Si está activado el MD9440, las correcciones de herramienta derivadas de la modificación de datos de corrección de herramienta durante la parada del programa de pieza se aplicarán al reanudarse el programa de pieza.

2.5.7 Supresión de correcciones de herramienta (SUPD)

El comando SUPD permite suprimir las correcciones de herramienta por secuencias.

La ventaja frente a la deselección de corrección de herramienta con D0 es que con SUPD se conserva la información de corrección de herramienta y, por tanto, se suprime la reactivación de los datos de corrección de herramienta al programar el número D.

SUPD se utiliza exclusivamente para suprimir la corrección de la longitud de herramienta, p. ej., en secuencias del programa de pieza en las que deba calcularse el origen pero no la longitud de herramienta. No se recomienda el uso con G41/G42 activo para suprimir también las correcciones de radio de herramienta.

Sintaxis

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

Descripción

SUPD:	Comando G para desactivar el bloque de datos de corrección para la herramienta activa en la secuencia activa	
	Efecto:	Secuencia a secuencia

Condiciones

- SUPD debe utilizarse solo en secuencias lineales.
- SUPD no puede utilizarse en acciones síncronas.
- SUPD no puede utilizarse con las siguientes funciones:
  - Corrección de radio de herramienta 3D para fresado frontal 3D (CUT3DFxx)
  - Tablas de levas (CTAB)
- Si bien es posible utilizar SUPD con la corrección de radio de herramienta activa (G41/G42), no es recomendable.
  Si, a pesar de ello, debe utilizarse la función, el siguiente dato de operador debe ajustarse a 0:
  SD42480 $SC_STOP_CUTCOM_STOPRE = 0
  Con ello se impide que se interrumpa el programa en caso de G41/G42 activos.

Ejemplo

En el subprograma SUB_SUP debe suprimirse la longitud de herramienta durante el desplazamiento.

Programa de pieza

Código de programa	Comentarios
...	

Código de programa	Comentarios
N300 $P_UIFR[1]=CTRANS(X,1000,Y,400,Z,-120)	
N310 T="BALL_D3"	
N320 M6	
N330 TRAFOOF	
N340 G54 G0 Z49 D1	
N350 G0 X1100 Y500 C0 A0	
N360 **SUB_SUP**	; Llamada del subprograma.
N370 G0 X1200	
N380 M30	

Subprograma con D0

Código de programa	Comentarios
N10 PROC **SUB_SUP**	
N20 DEF INT NUMBER	
N30 NUMBER=$P_TOOL	
N40 G0 Z49 **D0**	; Deselección de las correcciones de herramienta.
N50 D=NUMBER	; Nueva selección de las correcciones de herramienta.
N60 RET	

Subprograma con SUPD

Código de programa	Comentarios
N10 PROC **SUB_SUP**	
N40 G0 Z49 **SUPD**	; Supresión de las correcciones de herramientas en la secuencia activa.
N60 RET	

2.5.8 Offset programable de corrección de herramienta (TOFFL, TOFF, TOFFR, TOFFLR)

A través de las direcciones TOFFx, el usuario tiene la posibilidad de modificar la longitud de herramienta efectiva y el radio de herramienta efectivo en el programa CN sin cambiar los datos de corrección de herramienta guardados en la memoria de corrección.

Al finalizar el programa se borran los offsets programados.

Sintaxis

Offset de longitud de herramienta

```
TOFFL=<valor>
TOFFL[1]=<valor> TOFFL[2]=<valor> TOFFL[3]=<valor>
TOFF[<EjeGeo>]=<valor>
```

La longitud de herramienta se puede modificar simultáneamente en las tres componentes. Sin embargo, no se pueden utilizar a la vez en una secuencia los comandos del grupo TOFFL/ TOFFL[1..3], por un lado, y los del grupo TOFF[<EjeGeo>], por el otro. En una secuencia tampoco se deben escribir a la vez TOFFL y TOFFL[1].

Si en una secuencia no se programan las tres componentes longitudinales de herramienta, las componentes no programadas no sufren modificaciones. De esta manera es posible estructurar correcciones para varias componentes por secuencia. Sin embargo, esto solamente es válido mientras las componentes de herramienta se modifiquen bien solo con TOFFL, bien solo con TOFF. Un cambio de la variante de programación de TOFFL a TOFF o viceversa, borra primero todos los offsets de longitud de herramienta que pudieran haberse programado antes (ver ejemplo 3).

Offset del radio de herramienta

```
TOFFR=<valor>
```

Offset simultáneo de longitud de herramienta y radio de herramienta

```
TOFFLR=<valor>
```

Descripción

TOFFL:	Corrección de la longitud de herramienta efectiva TOFFL se puede programar con o sin índice:	
	TOFFL=...	El valor de offset programado actúa en el mismo sentido en el que actúa la componente longitudinal de herramienta **L1** guardada en la memoria de corrección. Las instrucciones TOFFL y TOFFL[1] tienen idéntico efecto.
	TOFFL[1]=... TOFFL[2]=... TOFFL[3]=...	El valor de offset programado actúa en el mismo sentido en el que actúa la componente longitudinal de herramienta **L1**, **L2** o **L3** guardada en la memoria de corrección.
	Nota: El tipo de herramienta y el plano de trabajo actual (G17/G18/G19) determinan la manera en que se calculan los valores de corrección de la longitud de herramienta en los ejes.	
TOFF:	Corrección de la longitud de herramienta en la componente paralela al eje geométrico indicado TOFF actúa en el sentido de la componente longitudinal de herramienta que actúa paralelamente al eje geométrico indicado en el índice cuando la herramienta no está girada (portaherramientas orientable o bien transformada de orientación). **Nota:** Un frame no afecta a la asignación de los valores programados a las componentes longitudinales de herramienta. Por lo tanto, para la asignación de las componentes longitudinales de herramienta a los ejes geométricos no se toma como referencia el sistema de coordenadas de pieza (WKS), sino el sistema de coordenadas de herramienta en la posición básica de herramienta.	
<EjeGeo>:	Identificador del eje geométrico	
TOFFR:	Corrección del radio de herramienta efectivo TOFFR modifica el radio de herramienta efectivo al valor de offset programado **con la corrección de radio de herramienta activa**.	

TOFFLR:	Corrección de la longitud de herramienta efectiva en la componente L1 **y** del radio de herramienta efectivo **Nota:** En las herramientas con redondeo de esquinas (tipos 111, 121, 131 y 156), TOFFLR corrige también el radio de redondeo.	
<Value>:	Valor offset	
	Tipo:	REAL

Ejemplos

Ejemplo 1: offset de longitud de herramienta positivo

La herramienta activa es una broca con la longitud L1 = 100 mm.

El plano activo es G17. Por lo tanto, la broca señala en la dirección Z.

La longitud de broca efectiva debe alargarse 1 mm. Para la programación de este offset de longitud de herramienta se dispone de las siguientes variantes:

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

Ejemplo 2: offset de longitud de herramienta negativo

La herramienta activa es una broca con la longitud L1 = 100 mm.

El plano activo es G18. Por lo tanto, la broca señala en la dirección Y.

La longitud de broca efectiva debe acortarse 1 mm. Para la programación de este offset de longitud de herramienta se dispone de las siguientes variantes:

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

Ejemplo 3: Cambio del tipo de programación de TOFFL a TOFF

La herramienta activa es una fresa. El plano activo es G17.

```
Código de programa                  Comentarios
N10 TOFFL[1]=3 TOFFL[3]=5           ; Offsets activos: L1=3, L2=0, L3=5
N20 TOFFL[2]=4                      ; Offsets activos: L1=3, L2=4, L3=5
N30 TOFF[Z]=1.3                     ; Offsets activos: L1=0, L2=0, L3=1.3
```

Ejemplo 4: Asignación de valores de offset tras el cambio de plano

```
Código de programa          Comentarios
N10 $TC_DP1[1,1]=120
N20 $TC_DP3[1,1]=100        ; Longitud de herramienta L1=100 mm
N30 T1 D1 G17
N40 TOFF[Z]=1.0             ; Offset en dirección Z (corresponde a L1 en G17).
```

Código de programa	Comentarios
N50 G0 X0 Y0 Z0	; Posición de eje de la máquina X0 Y0 Z101
N60 G18 G0 X0 Y0 Z0	; Posición de eje de la máquina X0 Y100 Z1
N70 G17	
N80 **TOFFL=1.0**	; Offset en dirección L1 (corresponde a Z en G17).
N90 G0 X0 Y0 Z0	; Posición de eje de la máquina X0 Y0 Z101.
N100 G18 G0 X0 Y0 Z0	; Posición de eje de la máquina X0 Y101 Z0.

En este ejemplo se conserva el offset de 1 mm en el eje Z al cambiar a G18 en la secuencia N60; la longitud de herramienta efectiva en el eje Y es la longitud de herramienta sin modificar de 100 mm.

En la secuencia N100, por el contrario, el offset actúa en el eje Y al cambiar a G18, porque durante la programación ha sido asignado a la longitud de herramienta L1 y con G18 esta componente longitudinal actúa en el eje Y.

Ejemplo 5: Offset simultáneo de longitud de herramienta y radio de herramienta

a, fresa de mango **sin** redondeo de esquinas (tipo de herramienta 120):

Código de programa	Comentarios
...	
TOFFLR=0.1	; Offsets activos:
	; Offset longitud de herramienta (L1) = 5
	; Offset radio de herramienta = 5
...	

b, fresa de mango **con** redondeo de esquinas (tipo de herramienta 121):

Código de programa	Comentarios
...	
TOFFLR=0.1	; Offsets activos:
	; Offset longitud de herramienta (L1) = 0.1
	; Offset radio de herramienta = 0.1
	; Offset radio de redondeo = 0.1
...	

Información adicional

Offsets de longitud de herramienta

En función del tipo de programación, los offsets de longitud de herramienta programados se asignan bien a las componentes longitudinales de herramienta L1, L2 y L3 (TOFFL) guardadas en la memoria de corrección, bien a los ejes geométricos (TOFF). En caso de cambio de plano (G17/G18/G19 ↔ G17/G18/G19), los offsets programados se tratan de la forma correspondiente:

- Si los valores de offset están asignados a las componentes longitudinales de herramienta, los sentidos en los que actúan los offsets programados se cambian como corresponde.
- Si los valores de offset están asignados a los ejes geométricos, un cambio de plano no influye en la asignación relativa a los ejes de coordenadas.

Al asignar los valores de offset programados a las componentes longitudinales de herramienta se evalúan los siguientes datos de operador:

SD42940 $SC_TOOL_LENGTH_CONST (cambio de las componentes longitudinales de herramienta con cambio de plano)

SD42950 $SC_TOOL_LENGTH_TYPE (asignación de la compensación longitudinal de herramienta independientemente del tipo de herramienta)

Si estos datos de operador tienen valores válidos diferentes de 0, estos tienen prioridad frente al contenido del grupo G 6 (selección del plano G17/G18/G19) o bien frente al tipo de herramienta incluido en los datos de herramienta ($TC_DP1[<N.º T>, <N.º D>]). Por lo tanto, estos datos de operador influyen en la evaluación de los offsets de la misma manera que las componentes longitudinales de herramienta de L1 a L3.

Offset del radio de herramienta

La dirección TOFFR tiene casi el mismo efecto que la dirección OFFN (Página 257). Se obtiene una diferencia únicamente con la transformada de curva envolvente activa (TRACYL) y la corrección de pared de ranura activa. En este caso OFFN actúa con signo negativo sobre el radio de herramienta; TOFFR, por el contrario, lo hace con signo positivo.

OFFN y TOFFR pueden estar activos simultáneamente. Por lo general actúan de forma aditiva (salvo con la corrección de pared de ranura).

Cambio de herramienta

Todos los valores de offset se conservan al realizar un cambio de herramienta (cambio de filo). Por lo tanto, también pasan a estar activos con la nueva herramienta (el nuevo filo).

Variables de sistema para la lectura de los valores de offset actuales

Los offsets activos actualmente pueden leerse con las siguientes variables de sistema:

Variable de sistema	Descripción
$P_TOFFL [<n>] con 0 ≤ n ≤ 3	Lee el valor de offset actual de TOFFL (para n = 0) o TOFFL[1...3] (para n = 1, 2, 3) en el contexto de la decodificación previa.
$P_TOFF [<EjeGeo>]	Lee el valor de offset actual de TOFF[<EjeGeo>] en el contexto de la decodificación previa.
$P_TOFFR	Lee el valor de offset actual de TOFFR en el contexto de la decodificación previa.
$P_TOFFCR	Lee el valor de offset actual del radio de redondeo en el contexto de la decodificación previa.
$AC_TOFFL[<n>] con 0 ≤ n ≤ 3	Lee el valor de offset actual de TOFFL (para n = 0) o TOFFL[1...3] (para n = 1, 2, 3) en el contexto del proceso principal (acciones síncronas).
$AC_TOFF[<EjeGeo>]	Lee el valor de offset actual de TOFF[<EjeGeo>] en el contexto del proceso principal (acciones síncronas).
$AC_TOFFR	Lee el valor de offset actual de TOFFR en el contexto del proceso principal (acciones síncronas).
$AC_TOFFCR	Lee el valor de offset actual del radio de redondeo en el contexto del proceso principal (acciones síncronas).

Nota

Las variables de sistema $AC_TOFFL, $AC_TOFF, AC_TOFFR y AC_TOFFCR disparan una parada automática de decodificación previa al leer desde el contexto de la decodificación previa (programa CN).

Aplicaciones

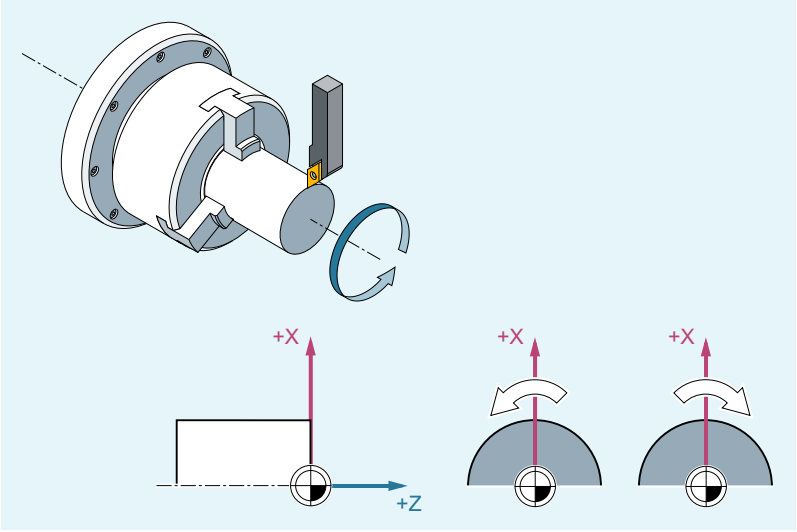
La función "Offset de corrección de herramienta programable" es especialmente interesante para fresas esféricas y fresas con radios de redondeo, ya que a menudo estas se calculan en el sistema CAM respecto al centro de la esfera en lugar de respecto a la punta de esta. Sin embargo, al efectuar la medición de la herramienta, normalmente se mide la punta de la herramienta y el resultado se guarda como longitud de la herramienta en la memoria de corrección.

Para la corrección de radio de herramienta 3D por medio de una fresa esférica es recomendable corregir al mismo tiempo con el mismo valor la longitud y el radio de la herramienta. Para ello puede usarse la dirección TOFFLR.

2.6 Movimiento de cabezales

2.6.1 Velocidad de giro del cabezal (S), sentido de giro del cabezal (M3/M4/M5)

Los datos de velocidad y sentido de giro del cabezal colocan el cabezal en un movimiento de giro y cumplen el requisito para el mecanizado con arranque de virutas.



Además del cabezal principal puede haber otros cabezales (p. ej., en el caso de los tornos, el contracabezal o una herramienta motorizada). Generalmente, el cabezal principal se declara como cabezal maestro mediante un dato de máquina. Esta asignación puede modificarse por medio de un comando CN.

Sintaxis

```
S.../S<n>=...
M3 / M<n>=3
M4 / M<n>=4
M5 / M<n>=5
```

SETMS(<n>)	
...	
SETMS	

Descripción

S...:	Velocidad de giro del cabezal maestro en revoluciones/min
S<n>=...:	Velocidad de giro del cabezal <n> en revoluciones/min

	Nota: La velocidad definida con S0=… se tiene en cuenta para el cabezal maestro.
M3:	Giro a la derecha para cabezal maestro
M<n>=3:	Giro a la derecha para cabezal <n>
M4:	Giro a la izquierda para cabezal maestro
M<n>=4:	Giro a la izquierda para cabezal <n>
M5:	Parada del cabezal para el cabezal maestro
M<n>=5:	Parada del cabezal para cabezal <n>
SETMS(<n>):	El cabezal <n> debe actuar como cabezal maestro
SETMS:	SETMS sin indicación de cabezal retorna al cabezal maestro configurado

Nota

Por cada secuencia CN se pueden programar como máximo 3 valores S, p. ej.:

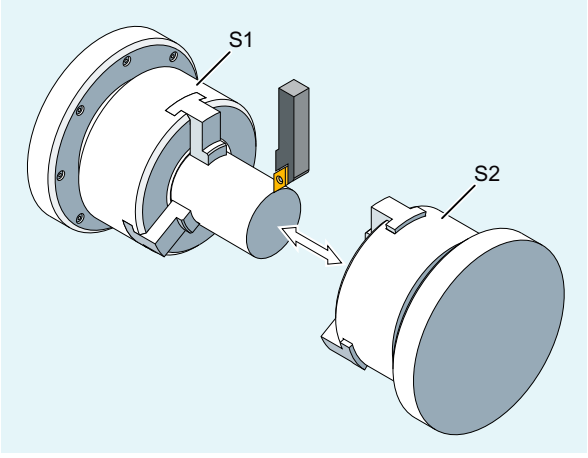
```
S... S2=... S3=...
```

Nota

SETMS debe encontrarse en una secuencia propia.

Ejemplo

S1 es el cabezal maestro, S2 es el segundo cabezal de trabajo. La pieza debe ser mecanizada en sus 2 extremos. Por eso, es preciso un reparto de los pasos de mecanizado. Tras el tronzado, el sincronizador (S2) toma la pieza para el mecanizado de la parte tronzada. Para ello se define el cabezal S2 como cabezal maestro, para el que será válido G95.



Código de programa	**Comentarios**
N10 S300 M3	; Velocidad y sentido de giro para el cabezal de accionamiento = cabezal maestro predefinido.
...	; Mecanizado de la parte derecha de la pieza.
N100 SETMS(2)	; S2 es ahora el cabezal maestro.
N110 S400 G95 F…	; Velocidad para el nuevo cabezal maestro.

Código de programa	Comentarios
...	; Mecanizado de la parte izquierda de la pieza.
N160 SETMS	; Retornar al cabezal maestro S1.

Información adicional

Interpretación del valor S con el cabezal maestro

Si en el grupo G 1 (comandos de desplazamiento modalmente activos) está activa la función G331 o G332, el valor S programado siempre se interpretará como velocidad de giro en rpm. De lo contrario, la interpretación del valor S dependerá del grupo G 15 (tipo de avance): Con G96, G961 o G962 activos, el valor S se interpreta como velocidad de corte constante en m/min; en todos los demás casos, como velocidad de giro en rpm.

Al cambiar de G96/G961/G962 a G331/G332, el valor de la velocidad de corte constante se pone a cero; al cambiar de G331/G332 a una función dentro del grupo G 1 diferente de G331/G332, el valor de la velocidad de giro se pone a cero. En caso necesario, deben reprogramarse los correspondientes valores S.

Comandos M predefinidos M3, M4, M5

En una secuencia con comandos de ejes se activan las funciones M3, M4, M5 **antes** de que se inicien los desplazamientos de los ejes (ajuste por defecto del control).

Ejemplo:

Código de programa	Comentarios
N10 G1 F500 X70 Y20 S270 M3	; El cabezal acelera a 270 r/min; entonces los ejes X e Y inician su desplazamiento.
N100 G0 Z150 M5	; Parada del cabezal antes del movimiento de retirada en Z.

Nota

Mediante datos de máquina se define si los desplazamientos de los ejes se deben realizar después de que el cabezal haya alcanzado su velocidad programada o se haya detenido, o si los desplazamientos se deben realizar inmediatamente después de las activaciones programadas.

Mecanizado con varios cabezales

Desde un mismo canal se pueden controlar simultáneamente hasta 5 cabezales (un cabezal maestro y 4 cabezales adicionales).

Mediante datos de máquina se define uno de los cabezales como **cabezal maestro**. Para este cabezal son válidas las funciones especiales como, p. ej., roscar, tallar, avance por vuelta, y tiempo de espera. Para el resto de cabezales (p. ej., un segundo cabezal de trabajo o una herramienta motorizada) se debe indicar la velocidad así como el sentido de giro/parada del cabezal especificando el número en cuestión.

Ejemplo:

Código de programa	Comentarios
`N10 S300 M3 S2=780 M2=4`	`; Cabezal maestro: 300 rpm, giro a derechas` `2.° cabezal: 780 rpm, giro a izquierdas`

Conmutación programable del cabezal maestro

Mediante el comando `SETMS(<n>)` se puede definir cualquier cabezal como cabezal maestro en el programa CN. `SETMS` debe encontrarse en una secuencia propia.

Ejemplo:

Código de programa	Comentarios
`N10 SETMS(2)`	`; El cabezal 2 es ahora el cabezal maestro.`

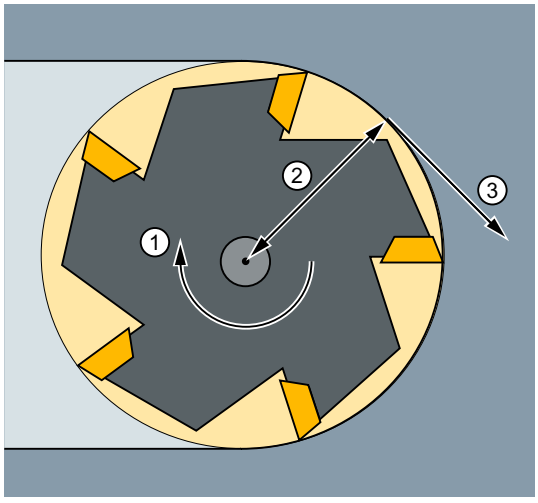
Nota

Para el cabezal maestro recién declarado se aplican ahora la velocidad de giro indicada con `S...` y las funciones programadas con `M3`, `M4`, `M5`.

Con `SETMS` sin indicación de cabezal se retorna al cabezal maestro definido mediante el dato de máquina.

2.6.2 Velocidad de corte de la herramienta (SVC)

Como alternativa a la velocidad de giro del cabezal, en los trabajos de fresado se puede programar también la velocidad de corte de la herramienta más habitual en la práctica.



① Velocidad de giro del cabezal
② Radio de la herramienta
③ Velocidad de corte de la herramienta

Mediante el radio de la herramienta activa, el control calcula la velocidad de giro activa del cabezal a partir de la velocidad de corte programada de la herramienta:

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

donde: S: Velocidad de giro del cabezal en rpm

SVC: Velocidad de corte de la herramienta en m/min o ft/min

R_{WKZ}: Radio de la herramienta activa en mm

El tipo ($TC_DP1) de la herramienta activa no se tiene en cuenta.

La velocidad de corte programada de la herramienta es independiente del avance de contorneado F y del grupo de funciones G 15 (tipo de avance). El sentido de giro y el arranque del cabezal se realizan mediante M3 o M4; la parada de cabezal, mediante M5.

Una modificación de los datos del radio de la herramienta en la memoria de corrección pasa a estar activa con la siguiente selección de corrección de herramienta o con la siguiente actualización de los datos de corrección activos.

El cambio de herramienta y la activación/desactivación de un bloque de datos de corrección de herramienta provocan que se vuelva a calcular la velocidad de giro activa del cabezal.

Requisitos

La programación de la velocidad de corte de la herramienta requiere:

- las condiciones geométricas de una herramienta rotativa (fresa o herramienta de taladrado)
- un bloque de datos de corrección de herramienta activo

Sintaxis

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

Descripción

SVC:	Palabra reservada para la programación de la velocidad de corte de la herramienta	
[<n>]:	Número del cabezal Con esta dirección extendida se indica para qué cabezal debe estar activa la velocidad de corte programada. Sin dirección extendida, la indicación se refiere siempre al cabezal maestro actual. **Nota:** Para cada cabezal se puede predefinir una velocidad de corte propia. **Nota:** La programación de SVC sin dirección extendida presupone que el cabezal maestro posee la herramienta activa. Al cambiar el cabezal maestro, el usuario debe seleccionar la herramienta correspondiente.	
<Value>:	Valor de la velocidad de corte de la herramienta	
	Unidad:	m/min (con G71/G710) o bien ft/min (con G70/G700)

`T... D...:`	En la secuencia con SVC debe conocerse el radio de la herramienta. Esto quiere decir que debe estar activa o seleccionada en la secuencia una herramienta con el bloque de datos de corrección de herramienta. Al realizar la programación en la misma secuencia, el orden de SVC y la selección T/D son indiferentes.
`S... M3/M4:`	La programación de la velocidad de giro del cabezal anula la selección de la velocidad de corte de la herramienta. **Nota:** Puede cambiarse libremente entre la programación SVC y la programación S, incluso con el cabezal girando. Se borra el valor que no está activo en cada caso.

Nota

La programación de SVC no se puede realizar si están activos los siguientes movimientos de avance del cabezal:

- Velocidad de corte constante: G96/G961/G962 S... (Página 101)
- Velocidad periférica de muela constante: SUG (Página 106)
- Posicionamiento del cabezal: SPOS/SPOSA/M19 (Página 121)
- Conmutación del cabezal maestro al modo de eje: M70 (Página 121)

A su vez, la programación de una de estas funciones provoca la cancelación de SVC (velocidad de corte de la herramienta).

Nota

Máxima velocidad de giro de la herramienta

Mediante la variable de sistema $TC_TP_MAX_VELO[<Número T>] se puede predefinir una velocidad de giro máxima de la herramienta (velocidad de giro del cabezal).
Si no está definido ningún límite de velocidad, no tiene lugar ninguna vigilancia.

Nota

Las trayectorias de herramienta generadas, p. ej., mediante sistemas CAD de "herramientas normalizadas" que ya tienen en cuenta el radio de la herramienta y que solo contienen la diferencia con respecto a la herramienta normalizada en el radio de corte no son compatibles con la programación de SVC.

Ejemplos

Para todos los ejemplos se debe aplicar: Portaherramientas = cabezal (para fresado estándar)

Ejemplo 1: fresa de radio 6 mm

Código de programa	**Comentarios**
`N10 G0 X10 T1 D1`	`; Selección de la fresa con, p. ej., $TC_DP6[1,1] = 6 (radio de herramienta = 6 mm)`

Código de programa	Comentarios
N20 SVC=100 M3	; Velocidad de corte = 100 m/min ⇒ Velocidad de giro del cabezal resultante: S = (100 m/min * 1000) / (6,0 mm * 2 * 3,14) = 2653,93 rpm
N30 G1 X50 G95 FZ=0.03	; SVC y avance por diente
...	

Ejemplo 2: selección de herramienta y SVC en la misma secuencia

Código de programa	Comentarios
N10 G0 X20	
N20 T1 D1 SVC=100	; Selección de herramienta y de bloque de datos de corrección junto con SVC en la secuencia (cualquier orden).
N30 X30 M3	; Arranque del cabezal con sentido de giro a la derecha, velocidad de corte 100 m/min
N40 G1 X20 F0.3 G95	; SVC y avance por vuelta

Ejemplo 3: especificación de velocidades de corte para dos cabezales

Código de programa	Comentarios
N10 SVC[3]=100 M6 T1 D1	
N20 SVC[5]=200	; El radio de herramienta de la corrección de herramienta activa es el mismo para los dos cabezales; la velocidad de giro efectiva es distinta para el cabezal 3 y el 5.

Ejemplo 4:

Supuestos:

El maestro referido al cambio de herramienta se determina mediante los portaherramientas:

DM20124 $MC_TOOL_MANAGEMENT_TOOLHOLDER > 1

Al cambiar la herramienta se conserva la corrección de herramienta antigua y la corrección de una herramienta nueva no se activará hasta la programación de D:

DM20270 $MC_CUTTING_EDGE_DEFAULT = - 2

Código de programa	Comentarios
N10 $TC_MPP1[9998,1]=2	; El puesto de almacén es el portaherramientas
N11 $TC_MPP5[9998,1]=1	; El puesto de almacén es el portaherramientas 1
N12 $TC_MPP_SP[9998,1]=3	; El portaherramientas 1 está asignado al cabezal 3
N20 $TC_MPP1[9998,2]=2	; El puesto de almacén es el portaherramientas
N21 $TC_MPP5[9998,2]=4	; El puesto de almacén es el portaherramientas 4
N22 $TC_MPP_SP[9998,2]=6	; El portaherramientas 4 está asignado al cabezal 6
N30 $TC_TP2[2]="WZ2"	
N31 $TC_DP6[2,1]=5.0	; Radio = 5,0 mm de T2, corrección D1
N40 $TC_TP2[8]="WZ8"	
N41 $TC_DP6[8,1]=9.0	; Radio = 9,0 mm de T8, corrección D1

Código de programa	Comentarios
N42 $TC_DP6[8,4]=7.0	; Radio = 7,0 mm de T8, corrección D4
...	
N100 SETMTH(1)	; Definir número de portaherramientas maestro
N110 T="WZ2" M6 D1	; La herramienta T2 se carga y la corrección D1 se activa.
N120 G1 G94 F1000 M3=3 SVC=100	; S3 = (100 m/min * 1000) / (5,0 mm * 2 * 3,14) = 3184,71 rpm
N130 SETMTH(4)	; Definir número de portaherramientas maestro
N140 T="WZ8"	; Corresponde a T8="WZ8"
N150 M6	; Corresponde a M4=6
	La herramienta "WZ8" se para en el portaherramientas maestro, pero debido a DM20270=-2 se conserva la corrección de herramienta antigua.
N160 SVC=50	; S3 = (50 m/min * 1000) / (5,0 mm * 2 * 3,14) = 1592,36 rpm
	La corrección del portaherramientas 1 aún está activa y está asignada al cabezal 3.
N170 D4	; La corrección D4 de la nueva herramienta "WZ8" pasa a estar activa (en el portaherramientas 4).
N180 SVC=300	; S6 = (300 m/min * 1000) / (7,0 mm * 2 * 3,14) = 6824,39 rpm
	El cabezal 6 está asignado al portaherramientas 4.

Ejemplo 5:

Supuestos:

Los cabezales son, a la vez, portaherramientas:

DM20124 $MC_TOOL_MANAGEMENT_TOOLHOLDER = 0

Al cambiar la herramienta se selecciona automáticamente el bloque de datos de corrección de herramienta D4:

DM20270 $MC_CUTTING_EDGE_DEFAULT = 4

Código de programa	Comentarios
N10 $TC_MPP1[9998,1]=2	; El puesto de almacén es el portaherramientas
N11 $TC_MPP5[9998,1]=1	; El puesto de almacén es el portaherramientas 1 = cabezal 1
N20 $TC_MPP1[9998,2]=2	; El puesto de almacén es el portaherramientas
N21 $TC_MPP5[9998,2]=3	; El puesto de almacén es el portaherramientas 3 = cabezal 3
N30 $TC_TP2[2]="WZ2"	
N31 $TC_DP6[2,1]=5.0	; Radio = 5,0 mm de T2, corrección D1
N40 $TC_TP2[8]="WZ8"	
N41 $TC_DP6[8,1]=9.0	; Radio = 9,0 mm de T8, corrección D1
N42 $TC_DP6[8,4]=7.0	; Radio = 7,0 mm de T8, corrección D4
...	
N100 SETMS(1)	; Cabezal 1 = cabezal maestro
N110 T="WZ2" M6 D1	; La herramienta T2 se carga y la corrección D1 se activa.
N120 G1 G94 F1000 M3 SVC=100	; S1 = (100 m/min * 1000) / (5,0 mm * 2 * 3,14) = 3184,71 rpm
N200 SETMS(3)	; Cabezal 3 = cabezal maestro

Código de programa	Comentarios
N210 M4 SVC=150	; S3 = (150 m/min * 1000) / (5,0 mm * 2 * 3,14) = 4777,07 rpm
	Se refiere a la corrección de herramienta D1 de T="WZ2", S1 sigue girando con la velocidad de giro antigua.
N220 T="WZ8"	; Corresponde a T8="WZ8"
N230 M4 SVC=200	; S3 = (200 m/min * 1000) / (5,0 mm * 2 * 3,14) = 6369,43 rpm
	Se refiere a la corrección de herramienta D1 de T="WZ2".
N240 M6	; Corresponde a M3=6
	La herramienta "WZ8" se para en el cabezal maestro; la corrección de herramienta D4 de la herramienta nueva pasa a estar activa.
N250 SVC=50	; S3 = (50 m/min * 1000) / (7,0 mm * 2 * 3,14) = 1137,40 rpm
	La corrección D4 sobre el cabezal maestro está activa.
N260 D1	; Corrección D1 de la herramienta nueva "WZ8" activa.
N270 SVC[1]=300	; S1 = (300 m/min * 1000) / (9,0 mm * 2 * 3,14) = 5307,86 rpm
	S3 = (50 m/min * 1000) / (9,0 mm * 2 * 3,14) = 884,64 rpm
...	

Información adicional

Radio de la herramienta

Los siguientes datos de corrección de herramienta (de la herramienta activa) contribuyen al radio de herramienta:

- $TC_DP6 (radio - geometría)
- $TC_DP15 (radio - desgaste)
- $TC_SCPx6 (corrección a $TC_DP6)
- $TC_ECPx6 (corrección a $TC_DP6)

No se tienen en cuenta:

- Correcciones de radio online
- Creces para el contorno programado (OFFN)

Corrección del radio de herramienta (G41/G42)

Tanto la corrección del radio de herramienta (G41/G42) como SVC se refieren al radio de herramienta, pero están desacoplados funcionalmente y son independientes entre sí.

Roscado sin mandril de compensación (G331, G332)

La programación de SVC también es posible junto con G331 o G332.

Acciones síncronas

No se puede especificar SVC a partir de acciones síncronas.

Lectura de la velocidad de corte y la variante de programación de la velocidad de giro del cabezal

La velocidad de corte de un cabezal y la variante de programación de la velocidad de giro (velocidad de giro del cabezal S o velocidad de corte de la herramienta SVC) se pueden leer mediante variables de sistema:

- Con la parada de decodificación previa en el programa de pieza mediante variables de sistema:

$AC_SVC[<n>]	Velocidad de corte que estaba activa durante la preparación de la secuencia de proceso principal actual para el cabezal con el número <n>.
$AC_S_TYPE[<n>]	Variante de programación de la velocidad de giro del cabezal que estaba activa durante la preparación de la secuencia de proceso principal actual para el cabezal con el número <n>.

Valor:	Significado:
1	Velocidad de giro del cabezal S en r/min
2	Velocidad de corte de la herramienta SVC en m/min o ft/min

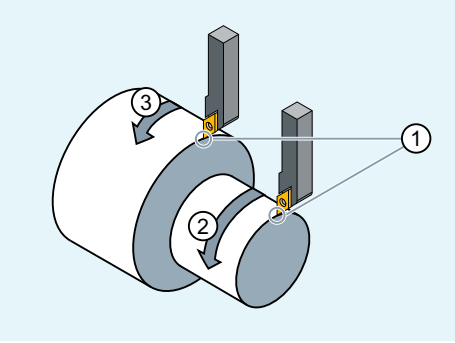
- Sin la parada de decodificación previa en el programa de pieza mediante variables de sistema:

$P_SVC[<n>]	Velocidad de corte programada para el cabezal <n>
$P_S_TYPE[<n>]	Variante de programación de la velocidad de giro del cabezal programada para el cabezal <n>

Valor:	Significado:
1	Velocidad de giro del cabezal S en r/min
2	Velocidad de corte de la herramienta SVC en m/min o ft/min

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

Si está activa la función "Velocidad de corte constante", la velocidad de giro del cabezal varía en función del diámetro de la pieza de manera que la velocidad de corte S en m/min o ft/min se mantenga constante en el filo de la herramienta.



① Velocidad de corte constante
② Velocidad de giro del cabezal aumentada
③ Velocidad de giro del cabezal reducida

De ello resultan las ventajas siguientes:

- Esquemas de torneado uniformes y, en consecuencia, mayor calidad de superficie
- Mecanizado que protege la herramienta

Sintaxis

Activar/desactivar la velocidad de corte constante para el cabezal maestro:

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

Limitación de la velocidad de giro para el cabezal maestro:

```
LIMS=<Valor>
LIMS[<Cabezal>]=<Valor>
```

Otro eje de referencia para G96/G961/G962:

```
SCC[<Eje>]
```

Nota

SCC[<Eje>] se puede programar por separado o juntamente con G96/G961/G962.

Descripción

G96:	Avance por vuelta (como con G95 (Página 108)) y velocidad de corte constante Con G96 se activa G95 automáticamente. En caso de que G95 no se hubiera conectado antes, debe indicarse un nuevo valor de avance F... en la llamada de G96.	
G961:	Avance lineal (como con G94 (Página 108)) y velocidad de corte constante	
G962:	Avance lineal o avance por vuelta y velocidad de corte constante	
S...:	Junto con G96, G961 o G962, S... no se interpreta como velocidad de giro del cabezal, sino como velocidad de corte. La velocidad de corte actúa siempre sobre el cabezal maestro.	
	Unidad:	m/min (con G71/G710) o bien ft/min (con G70/G700)
	Rango de valores:	0,1 m/min ... 9999 9999,9 m/min
G97:	Avance por vuelta y velocidad de giro del cabezal constante (velocidad de corte constante DES)	
G971:	Avance lineal y velocidad de giro del cabezal constante (velocidad de corte constante DES)	
G972:	Avance lineal o avance por vuelta y velocidad de giro del cabezal constante (velocidad de corte constante DES)	
G973:	Avance por vuelta sin limitación de la velocidad de giro del cabezal y velocidad de giro del cabezal constante (G97 sin LIMS para modo ISO)	
	Nota: Después de G97 (o G971 ... G973), S... vuelve a interpretarse como velocidad del cabezal en revoluciones/min. En caso de que no esté indicada ninguna velocidad de giro de cabezal nueva, se conserva la última velocidad ajustada mediante G96 (o G961 o G962).	
LIMS:	Limitación de la velocidad de giro para el cabezal maestro (sólo activa con G96/G961/G97 activo) En máquinas con cabezales maestros conmutables, en una secuencia se pueden programar limitaciones con distintos valores para hasta 4 cabezales.	
	<Cabezal>:	Número del cabezal
	<Valor>:	Límite superior de la velocidad de giro del cabezal en revoluciones/min
SCC:	Si la función G96/G961/G962 está activa, con SCC[<Eje>] se puede asignar cualquier eje geométrico como eje de referencia.	

Nota

Para la primera selección de G96/G961/G962 debe introducirse una velocidad de corte constante S...; para cada nueva selección de G96/G961/G962 la introducción es opcional.

Nota

La limitación de la velocidad de giro programada con LIMS no debe superar la velocidad de giro límite programada con G26 o definida mediante datos del operador.

Nota

El eje de referencia para G96/G961/G962 en el momento de la programación de SCC[<Eje>] tiene que ser un eje geométrico conocido en el canal. La programación de SCC[<Eje>] también es posible con G96/G961/G962 activo.

Ejemplos

Ejemplo 1: Activar la velocidad de corte constante con limitación de la velocidad de giro

```
Código de programa                          Comentarios
N10 SETMS(3)
N20 G96 S100 LIMS=2500                      ; Velocidad de corte constante = 100 m/
                                            min, máx. velocidad de giro = 2500 rpm
...
N60 G96 G90 X0 Z10 F8 S100 LIMS=444         ; Máx. velocidad de giro = 444 rpm
```

Ejemplo 2: Especificar la limitación de la velocidad de giro para 4 cabezales

Las limitaciones de la velocidad de giro se definen para el cabezal 1 (cabezal maestro) y los cabezales 2, 3 y 4:

```
Código de programa
N10 LIMS=300 LIMS[2]=450 LIMS[3]=800
LIMS[4]=1500
...
```

Ejemplo 3: Asignación de un eje Y en un refrentado con eje X

```
Código de programa          Comentarios
N10 G18 LIMS=3000 T1 D1     ; Limitación de la velocidad de giro a 3000 rpm
N20 G0 X100 Z200
N30 Z100
N40 G96 S20 M3              ; Velocidad de corte constante = 20 m/min, dependien-
                            te del eje X.
N50 G0 X80
N60 G1 F1.2 X34             ; Refrentado en X a 1,2 mm/vuelta.
N70 G0 G94 X100
N80 Z80
N100 T2 D1
N110 G96 S40 SCC[Y]         ; El eje Y se asigna a G96 y G96 se activa (es posible
                            en una secuencia). Velocidad de corte constante =
                            40 m/min, dependiente del eje Y.
...
N140 Y30
N150 G01 F1.2 Y=27          ; Ranurado en Y, avance F = 1,2 mm/vuelta.
N160 G97                    ; Velocidad de corte constante desactivada.
N170 G0 Y100
```

Información adicional

Cálculo de la velocidad de giro del cabezal

Para calcular la velocidad de giro del cabezal a partir de la velocidad de corte programada se toma como base la posición ENS del eje transversal (radio).

Nota

Los frames entre WKS y ENS (como los frames programables SCALE, TRANS o ROT) se consideran en el cálculo de la velocidad de giro del cabezal y pueden provocar una variación de la velocidad de giro (p. ej. si varía el diámetro efectivo con SCALE).

Limitación de la velocidad de giro LIMS

Si debe mecanizarse una pieza con diámetros muy diferentes, es aconsejable definir una limitación para la velocidad de giro del cabezal con LIMS (velocidad de giro máxima del cabezal). De este modo, se evitan velocidades de giro del cabezal muy elevadas para pequeños diámetros. LIMS solo actúa cuando G96, G961 y G97 están activos. LIMS no tiene efecto en G971. Cuando la secuencia pasa a la marcha principal se adoptan todos los valores programados a los datos del operador.

Nota

Los límites de velocidad modificados en el programa de pieza con LIMS se incluyen en los datos del operador y quedan memorizados una vez finalizado el programa.

En cambio, si no se desea que los nuevos límites de velocidad modificados con LIMS sigan siendo válidos una vez finalizado el programa, debe insertarse la siguiente definición en el bloque GUD del fabricante de la máquina:

```
REDEF $SA_SPIND_MAX_VELO_LIMS PRLOC
```

Desactivar la velocidad de corte constante (G97/G971/G972/G973)

Después de G97 (o G971 ... G973), S... vuelve a interpretarse como velocidad de giro del cabezal en revoluciones/min. En caso de que no esté indicada ninguna velocidad de giro de cabezal nueva, se conserva la última velocidad ajustada mediante G96 (o G961 o G962).

La función G96/G961 también puede desactivarse con G94 o G95. En este caso, para continuar el proceso de mecanizado se toma la última velocidad de giro S... programada.

G97 se puede programar sin G96 previo. La función actúa entonces como G95; también puede programarse LIMS adicionalmente.

Con G973 se puede desactivar la velocidad de corte constante sin activar una limitación de revoluciones del cabezal.

Nota

El eje de refrentado se debe definir como tal mediante datos de máquina.

Desplazamiento en rápido G0

Al realizar desplazamientos en rápido (G0) no se producen variaciones en la velocidad de giro del cabezal.

Excepción:

Cuando se realiza un desplazamiento en rápido hasta el contorno y la siguiente secuencia CN contiene un desplazamiento G1/G2/G3/..., la velocidad de giro para el siguiente desplazamiento se fija ya en la secuencia de desplazamiento G0.

Otro eje de referencia para G96/G961/G962

Si la función G96/G961/G962 está activa, con SCC[<Eje>] se puede asignar cualquier eje geométrico como eje de referencia. Si cambia el eje de referencia y, por tanto, la posición de referencia de la punta de la herramienta (TCP, Tool Center Point) para la velocidad de corte constante, la velocidad de giro del cabezal resultante se alcanza a través de la rampa de frenado o de aceleración ajustada.

Cambio de eje de canal asignado

La propiedad Eje de referencia para G96/G961/G962 siempre está asignada a un eje geométrico. Si se intercambia el eje de canal asignado, la propiedad Eje de referencia para G96/G961/G962 permanece en el canal antiguo.

Los intercambios de eje geométrico no influyen en la asignación del eje geométrico a la velocidad de corte constante. Si un intercambio de eje geométrico modifica la posición de referencia TCP para G96/G961/G962, el cabezal adopta la nueva velocidad de giro a través de una rampa.

Si con el intercambio de eje geométrico no se asigna ningún eje de canal nuevo (p. ej., GEOAX(0,X)), la velocidad de giro del cabezal se congela según G97.

Ejemplos de intercambio de eje geométrico con asignaciones del eje de referencia:

Código de programa	**Comentarios**
N05 G95 F0.1	
N10 GEOAX(1,X1)	; El eje de canal X1 se convierte en el primer eje geométrico.
N20 SCC[X]	; El primer eje geométrico (X) se convierte en el eje de referencia
	; para G96/G961/G962.
N30 GEOAX(1,X2)	; El eje de canal X2 se convierte en el primer eje geométrico.
N40 G96 M3 S20	; El eje de referencia para G96 es el eje de canal X2.

Código de programa	**Comentarios**
N05 G95 F0.1	
N10 GEOAX(1,X1)	; El eje de canal X1 se convierte en el primer eje geométrico.
N20 SCC[X1]	; X1 e implícitamente el primer eje geométrico (X) se convierte
	;en eje de referencia para G96/G961/G962.
N30 GEOAX(1,X2)	; El eje de canal X2 se convierte en el primer eje geométrico.
N40 G96 M3 S20	; El eje de referencia para G96 es X2 o X, sin alarma.

Código de programa	**Comentarios**
N05 G95 F0.1	
N10 GEOAX(1,X2)	; El eje de canal X2 se convierte en el primer eje geométrico.

Código de programa	Comentarios
N20 SCC[X1]	; X1 no es un eje geométrico, alarma.

Código de programa	Comentarios
N05 G0 Z50	
N10 X35 Y30	
N15 SCC[X]	; El eje de referencia para G96/G961/G962 es X.
N20 G96 M3 S20	; Velocidad de corte constante a 10 mm/min CON.
N25 G1 F1.5 X20	; Refrentado en X a 1,5 mm/vuelta.
N30 G0 Z51	
N35 SCC[Y]	; El eje de referencia para G96 es Y,
	; Reducción de la velocidad de giro del cabezal (Y30).
N40 G1 F1.2 Y25	; Refrentado en Y a 1,2 mm/vuelta.

2.6.4 Conexión/desconexión de la velocidad periférica de muela constante (GWPSON, GWPSOF)

Con los procedimientos predefinidos GWPSON(...) y GWPSOF(...) se conecta o desconecta la velocidad periférica de muela constante (SUG) para muelas rectificadoras (tipo de herramienta: 400 a 499).

Sintaxis

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

Descripción

GWPSON(...):	Conectar la velocidad periférica de muela constante
GWPSOF(...):	Desconectar la velocidad periférica de muela constante
<TNr>:	Número T **Nota:** Solo es necesario si la velocidad periférica de muela constante debe conectarse o desconectarse para una muela no activa en vez de para la que se encuentra activa.
S<n>=…:	Velocidad periférica de muela en m/s o pies/s para el cabezal <n>
S0=... o S...:	Velocidad periférica de muela para el cabezal maestro

Consulta del estado

Mediante la siguiente variable del sistema se puede consultar desde el programa de pieza si la velocidad periférica de muela constante se encuentra activada para un determinado cabezal:

$P_GWPS[<n>] ; con <n> = número de cabezal

Valor	Descripción
0 (= FALSE)	SUG está **desactivada**.
1 (= TRUE)	SUG está **activada**.

2.6.5 Límite programable de la velocidad del cabezal (G25, G26)

Las velocidades de giro mínimas y máximas del cabezal definidas en los datos de máquina y del operador pueden modificarse mediante un comando de programa de pieza.

La programación de límites de velocidad para cabezal se puede realizar para todos los cabezales de un mismo canal.

Sintaxis

```
G25 S… S1=… S2=…
G26 S… S1=… S2=…
```

Descripción

`G25`:	Límite **inferior** de la velocidad de giro del cabezal
`G26`:	Límite **superior** de la velocidad de giro del cabezal
`S... S1=… S2=…` :	Velocidades de giro máximas o mínimas del cabezal **Nota:** por cada secuencia pueden programarse como máximo tres limitaciones de la velocidad de giro del cabezal. Rango de valores: 0.1 ... 9999 9999.9 rpm

Nota

La limitación de velocidad de giro del cabezal programada mediante `G25` o `G26` sobrescribe las velocidades límite en los datos del operador y, por tanto, permanece memorizada tras el final del programa.

En cambio, si no se desea que los nuevos límites de velocidad fijados con `G25` o `G26` sigan siendo válidos una vez finalizado el programa, deben insertarse en el bloque GUD del fabricante de la máquina las siguientes definiciones:

REDEF $SA_SPIND_MIN_VELO_G25 PRLOC

REDEF $SA_SPIND_MAX_VELO_G26 PRLOC

Ejemplo

Código de programa	**Comentarios**
`N10 G26 S1400 S2=350 S3=600`	`; Límite superior de velocidad de giro para el cabezal maestro, cabezales 2 y 3.`

2.7 Ajuste de avance

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

Estos comandos permiten ajustar en el programa CN las velocidades de avance para todos los ejes que intervienen en la secuencia de mecanizado.

Sintaxis

```
G93
G94
G95
F<Valor>
FGROUP(<Eje_1>,<Eje_2>,...)
FGREF[<Eje giratorio>]=<Radio de referencia>
FL[<Eje>]=<Valor>
```

Descripción

G93:	Tipo de avance de contorneado: avance inverso al tiempo en [rpm]
G94:	Tipo de avance de contorneado: avance lineal [mm/min], [pulgadas/min] o [grados/min]
G95:	Tipo de avance de contorneado: avance por vuelta [mm/vuelta] o [pulgadas/vuelta] El avance por vuelta puede calcularse a partir de un cabezal maestro, cualquier otro cabezal o un eje giratorio, a elegir.
F<Valor>	Avance de contorneado para todos los ejes de contorneado o para los seleccionados con FGROUP.
FGROUP:	Define los ejes de contorneado a los que se refiere el avance de contorneado programado en F.
FGREF:	FGREF permite programar el radio efectivo (<Radio de referencia>) para cada uno de los ejes giratorios indicados en FGROUP
FL:	Velocidad límite para ejes síncronos/de contorneado Se adopta la unidad definida con G94. Para cada eje (eje de canal, geométrico o de orientación) puede programarse un valor FL.
<Eje>:	Nombre de un eje de canal, tipo: AXIS

Ejemplos

Ejemplo 1: efecto de FGROUP

El siguiente ejemplo ilustra el efecto de FGROUP en la trayectoria y el avance de contorneado. La variable $AC_TIME contiene el tiempo en segundos desde el principio de la secuencia. Solamente es posible durante acciones síncronas.

Código de programa	Comentarios
N100 G0 X0 A0	
N110 FGROUP(X,A)	

Código de programa	Comentarios
N120 G91 G1 G710 F100	; Avance = 100 mm/min o 100 grados/min
N130 DO $R1=$AC_TIME	
N140 X10	; Avance = 100 mm/min, trayectoria = 10 mm, R1 = aprox. 6 s
N150 DO $R2=$AC_TIME	
N160 X10 A10	; Avance = 100 mm/min, trayectoria = 14,14 mm, R2 = aprox. 8 s
N170 DO $R3=$AC_TIME	
N180 A10	; Avance = 100 grados/min, trayectoria = 10 grados, R3 = aprox. 6 s
N190 DO $R4=$AC_TIME	
N200 X0.001 A10	; Avance = 100 mm/min, trayectoria = 10 mm, R4 = aprox. 6 s
N210 G700 F100	; Avance = 2540 mm/min o 100 grados/min
N220 DO $R5=$AC_TIME	
N230 X10	; Avance = 2540 mm/min, trayectoria = 254 mm, R5 = aprox. 6 s
N240 DO $R6=$AC_TIME	
N250 X10 A10	; Avance = 2540 mm/min, trayectoria = 254,2 mm, R6 = aprox. 6 s
N260 DO $R7=$AC_TIME	
N270 A10	; Avance = 100 grados/min, trayectoria = 10 grados, R7 = aprox. 6 s
N280 DO $R8=$AC_TIME	
N290 X0.001 A10	; Avance = 2540 mm/min, trayectoria = 10 mm, R8 = aprox. 0,288 s
N300 FGREF[A]=360/(2*$PI)	; Ajustar 1 grado = 1 pulgada sobre el radio efectivo.
N310 DO $R9=$AC_TIME	
N320 X0.001 A10	; Avance = 2540 mm/min, trayectoria = 254 mm, R9 = aprox. 6 s
N330 M30	

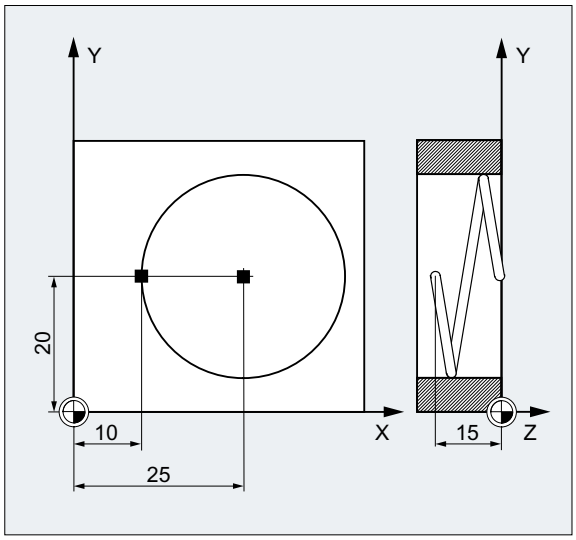
Ejemplo 2: desplazamiento de ejes síncronos con velocidad límite FL

La velocidad de los ejes de contorneado se reduce si el eje síncrono Z alcanza la velocidad límite.

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

Ejemplo 3: interpolación helicoidal

Los ejes de contorneado X e Y se desplazan con el avance programado, el avance para el eje Z se toma como avance para eje síncrono.

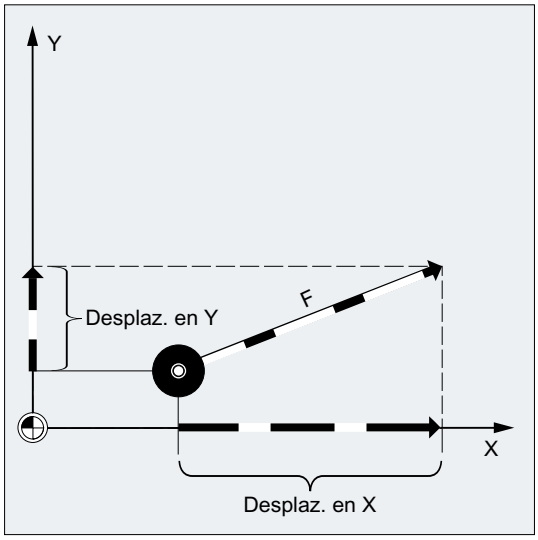


Código de programa	Comentarios
N10 G17 G94 G1 Z0 F500	; Cota de penetración de la herramienta.
N20 X10 Y20	; Desplazamiento a la posición inicial.
N25 FGROUP(X,Y)	; Los ejes X/Y son de contorneado, el eje Z es síncrono.
N30 G2 X10 Y20 Z-15 I15 J0 F1000 FL[Z]=200	; A la trayectoria circular se le aplica el avance de 1000 mm/min; en la dirección Z se realiza un desplazamiento síncrono.
...	
N100 FL[Z]=$MA_AX_VELO_LIMIT[0,Z]	; Al leer la velocidad del DM se anula la velocidad límite y se lee el valor del DM.
N110 M30	; Fin del programa.

Información adicional

Velocidad de avance para ejes de contorneado (F)

Por regla general, el avance en la trayectoria se calcula mediante la suma vectorial de las velocidades para cada uno de los ejes geométricos que participan en el desplazamiento. Está referido bien al centro de la fresa o bien a la punta de la herramienta cuando se trata de un torno.



La velocidad de avance se programa en la dirección `F`. Según el ajuste previo en los datos de máquina, las unidades de medida definidas con los comandos G se aplican en mm o en pulgadas.

Para cada secuencia CN puede programarse un valor `F`. La unidad de la velocidad de avance se especifica con uno de los comandos G `G93`/`G94`/`G95`. El avance `F` solamente actúa en ejes de contorneado y es válido hasta que se programa un nuevo valor de avance. Se admiten caracteres separadores tras la dirección `F`.

Ejemplos:

`F100` o `F 100`

`F.5`

`F=2*FEED`

Tipo de avance (G93/G94/G95)

Los comandos G `G93`, `G94` y `G95` son modalmente activos. Si se cambia entre `G93`, `G94` y `G95`, debe volver a programarse el valor de avance de contorneado. Cuando se mecaniza con ejes giratorios también puede indicarse el avance en grados/min.

Avance inverso al tiempo (G93)

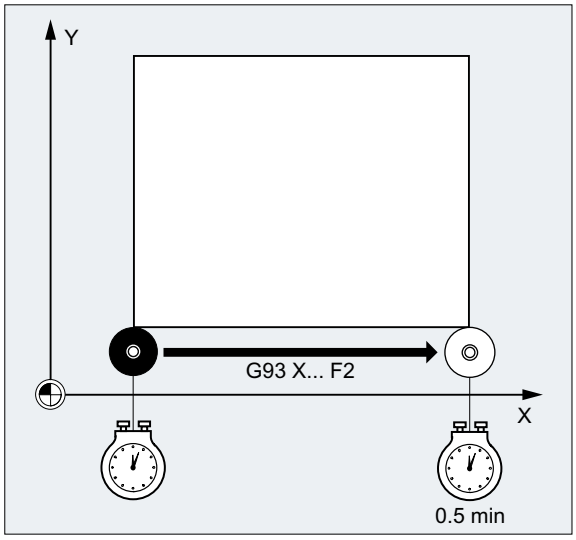
El avance inverso al tiempo indica la duración de la retirada de una secuencia.

Unidad: rpm

Ejemplo:

```
N10 G93 G01 X100 F2
```

Significado: la trayectoria programada se recorre en 0,5 min.



Nota

Si las longitudes de trayectoria varían sustancialmente de una secuencia a otra, deberá introducirse en `G93` un nuevo valor `F` en cada secuencia. Cuando se mecaniza con ejes giratorios también puede indicarse el avance en grados/min.

Avance para ejes síncronos

El avance programado en la dirección `F` se aplica a todos los ejes de contorneado programados en la secuencia, pero no es válido para los ejes síncronos. Los ejes síncronos se tratan como si necesitasen el mismo tiempo para recorrer su desplazamiento que los ejes de contorneado y el resto de ejes necesitan para alcanzar el punto final programado.

Velocidad límite para ejes síncronos (FL)

El comando `FL` permite programar una velocidad límite para ejes síncronos. Si no se programa ningún valor `FL`, se adopta la velocidad en rápido. `FL` se desactiva mediante la asignación al DM (DM36200 $MA_AX_VELO_LIMIT).

Desplazamiento de un eje de contorneado como eje síncrono (FGROUP)

Mediante `FGROUP` se define si un eje de contorneado debe desplazarse con avance de contorneado o como un eje síncrono. En la interpolación helicoidal puede establecerse, p. ej., que solamente dos ejes geométricos X e Y se desplacen con el avance programado. El eje Z se considera como si fuese un eje síncrono.

Ejemplo: `FGROUP(X,Y)`

Modificación de FGROUP

Es posible modificar el ajuste realizado con FGROUP:

1. Programando nuevamente FGROUP: p. ej., FGROUP(X,Y,Z)
2. Programando FGROUP sin indicar el eje: FGROUP()
   Tras FGROUP() es válido el estado inicial configurado en el dato de máquina. Los ejes geométricos se desplazan otra vez como ejes de contorneado.

Nota

Los identificadores de eje en FGROUP deben ser nombres de ejes de canal.

Sistema de unidades para el avance F

Con los comandos G G700 y G710 se define, además de los parámetros geométricos, el sistema de medida para los avances F, es decir:

- Con G700: [pulgadas/min]
- Con G710: [mm/min]

Nota

G70/G71 **no** influyen en los parámetros de avance.

Sistema de unidades para ejes síncronos con velocidad límite FL

La unidad de medida definida para F por medio del comando G G700/G710 también se aplica a FL.

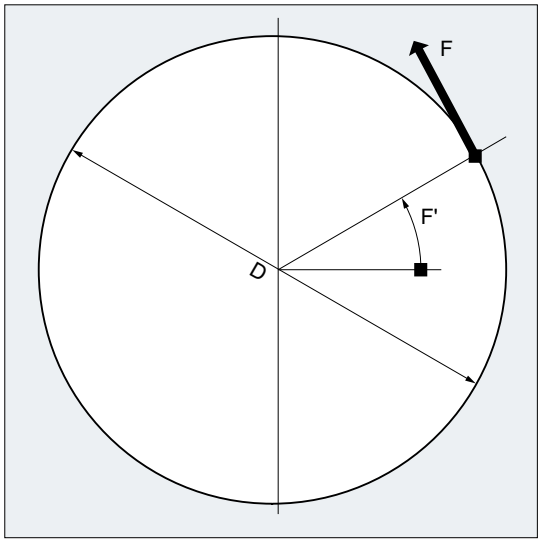
Sistema de unidades para ejes lineales y giratorios

En el caso de los ejes lineales y giratorios agrupados mediante FGROUP y que intervienen en una misma trayectoria, el avance adopta la unidad de medida de los ejes lineales. Dependiendo del ajuste previo efectuado con G94/G95, las unidades son mm/min o pulgadas/min, o bien mm/vuelta o pulgadas/vuelta.

La velocidad tangencial para el eje giratorio se calcula en mm/min o pulgadas/min teniendo en cuenta la siguiente fórmula:

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

donde: F: velocidad tangencial
F': velocidad angular
π: constante circular
D: diámetro



Desplazamiento de ejes giratorios con velocidad de contorneado F (FGREF)

Para los procesos de mecanizado en los que bien la herramienta, la pieza o las dos se deban desplazar como ejes giratorios, se puede programar el avance para el mecanizado de manera convencional mediante el valor F. Para ello debe indicarse un radio efectivo (radio de referencia) para cada uno de los ejes giratorios que intervienen.

La unidad del radio de referencia depende del ajuste `G70`/`G71`/`G700`/`G710`.

Para que se calcule el avance de contorneado deben incluirse en el comando `FGROUP` todos los ejes que intervienen.

A fin de mantener la compatibilidad con el comportamiento sin programación `FGREF`, tras el arranque del sistema y en caso de un RESET se activa la evaluación 1 grado = 1 mm. Esto equivale a un radio de referencia de FGREF = 360 mm/(2π) = 57.296 mm.

Nota

Este ajuste previo es independiente del sistema base activo (DM10240 $MN_SCALING_SYSTEM_IS_METRIC) y del ajuste `G70`/`G71`/`G700`/`G710` activo en ese momento.

Particularidades:

Código de programa

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

En esta programación, el valor F programado en N110 como avance de eje giratorio se calcula en grados/min, mientras que el cálculo del avance en N120 depende del ajuste G70/G71/G700/G710 activo en ese momento (100 pulgadas/min o 100 mm/min).

ATENCIÓN

Diferencia de avance

La evaluación FGREF también se aplica cuando en la secuencia sólo se han programado ejes giratorios. En este caso, la interpretación acostumbrada del valor F en grados/min sólo se aplica cuando el radio de referencia es, de acuerdo con el ajuste previo FGREF:

- Con G71/G710: FGREF[A]=57.296
- Con G70/G700: FGREF[A]=57.296/25.4

Lectura del radio de referencia

El valor del radio de referencia de un eje giratorio puede leerse mediante variables del sistema:

- En acciones síncronas o con parada de decodificación previa en el programa de pieza, a través de la variable del sistema:

  $AA_FGREF[<Eje>] Valor actual de proceso principal

- Sin parada de decodificación previa en el programa de pieza, a través de la variable del sistema:

  $PA_FGREF[<Eje>] Valor programado

Si no se ha programado ningún valor, en ambas variables se lee para los ejes giratorios el ajuste previo 360 mm/(2π) = 57.296 mm (equivale a 1 mm por grado).

Para los ejes lineales se lee siempre en ambas variables el valor 1 mm.

Lectura de los ejes de contorneado que determinan la velocidad

Los ejes que intervienen en la interpolación de la trayectoria pueden leerse mediante variables del sistema:

- En acciones síncronas o con parada de decodificación previa en el programa de pieza, a través de las variables del sistema:

  $AA_FGROUP[<Eje>] Devuelve el valor "1" cuando el eje indicado por medio del ajuste por defecto o mediante la programación FGROUP influye en la velocidad de contorneado de la secuencia de proceso principal actual. En caso contrario, la variable devuelve el valor "0".

  $AC_FGROUP_MASK Proporciona una clave de bits de los ejes de canal programados con FGROUP que contribuyen a la velocidad de contorneado.

- Sin la parada de decodificación previa en el programa de pieza mediante variantes de sistema:

$PA_FGROUP[<Eje>]	Devuelve el valor "1" cuando el eje indicado por medio del ajuste por defecto o mediante la programación `FGROUP` influye en la velocidad de contorneado. En caso contrario, la variable devuelve el valor "0".
$P_FGROUP_MASK	Proporciona una clave de bits de los ejes de canal programados con `FGROUP` que contribuyen a la velocidad de contorneado.

Factores de referencia de trayectoria para ejes de orientación con FGREF

En los ejes de orientación, el efecto de los factores `FGREF[]` depende de si la modificación de la orientación de la herramienta se realiza por interpolación vectorial o de ejes giratorios.

En la **interpolación de ejes giratorios**, los correspondientes factores `FGREF` de los ejes de orientación se incluyen individualmente como radio de referencia para los trayectos de los ejes, al igual que en el caso de los ejes giratorios.

En la **interpolación vectorial** actúa un factor `FGREF` efectivo que se calcula como media geométrica de los distintos factores `FGREF`:

FGREF[efectivo] = n.ª raíz de [(FGREF[A] * FGREF[B]...)]

donde: A: Identificador de eje del 1er eje de orientación
B: Identificador de eje del 2º eje de orientación
C: Identificador del 3er eje de orientación
n: Número de ejes de orientación

Ejemplo:

Para una transformada estándar de 5 ejes existen dos ejes de orientación y, en consecuencia, el factor efectivo se calcula como la raíz del producto de los dos factores axiales:

FGREF[efectivo] = raíz cuadrada de [(FGREF[A] * FGREF[B])]

Nota

Por lo tanto, el factor efectivo para ejes de orientación `FGREF` permite establecer un punto en la herramienta que el avance de contorneado programado toma como referencia.

2.7.2 Desplazar ejes de posicionado (POS, POSA, POSP, FA, WAITP, WAITMC)

Los ejes de posicionado se desplazan, independientemente de los ejes de contorneado, con avance de eje propio. Los comandos de interpolación no tienen validez alguna. Los comandos POS/POSA/POSP permiten desplazar los ejes de posicionado y coordinar simultáneamente los desplazamientos.

Ejemplos típicos de ejes de posicionado:

- Dispositivos de alimentación de palets
- Estaciones de medida

WAITP permite definir en el programa CN el punto en el cual se debe esperar hasta que un eje programado con POSA en una secuencia CN anterior haya alcanzado su punto final.

Con WAITMC se cambia inmediatamente a la siguiente secuencia CN al llegar a la meta de espera indicada.

Sintaxis

```
POS[<Eje>]=<Posición>
POSA[<Eje>]=<Posición>
POSP[<Eje>]=(<Posición final>,<Longitud parcial>,<Modo>)
FA[<Eje>]=<Valor>
WAITP(<Eje>) ; ¡Programación en la secuencia CN propia!
WAITMC(<Meta de espera>)
```

Descripción

POS/POSA:	Desplazamiento del eje de posicionado a la posición indicada POS y POSA tienen las mismas funciones, pero su comportamiento es distinto a la hora de cambiar de secuencia: • Con POS no se pasa a la siguiente secuencia CN hasta que se ha alcanzado la posición a la que debía llegarse. • Con POSA se pasa a la siguiente secuencia CN aunque no se haya alcanzado la posición a la que debía llegarse.		
	<Eje>:	Nombre del eje que se va a desplazar (identificador de eje geométrico o de canal)	
	<Posición>:	Posición de eje que debe alcanzarse	
		Tipo:	REAL
POSP:	Desplazamiento por tramos del eje de posicionado a la posición final indicada		
	<Posición final>:	Posición final de eje que debe alcanzarse	
	<Longitud parcial>:	Longitud de un tramo	
	<Modo>:	Modo de aproximación	
		= 0:	En los dos últimos tramos, la distancia restante hasta la posición final se divide en dos partes iguales (predeterminado).
		= 1:	De esta manera, la longitud parcial se adapta de forma que la suma de todas las longitudes parciales calculadas dé como resultado la distancia exacta hasta la posición final.
	Nota: POSP se utiliza especialmente para la programación de movimientos de vaivén (Página 984).		

FA:	Avance para el eje de posicionado indicado		
	<Eje>:	Nombre del eje que se va a desplazar (identificador de eje geométrico o de canal)	
	<Valor>:	Velocidad de avance	
		Unidad:	mm/min, pulgadas/min o grados/min
	Nota: Por cada secuencia CN pueden programarse como máximo 5 valores FA.		
WAITP:	Espera hasta el fin del desplazamiento de un eje de posicionado Al ejecutar las siguientes secuencias se espera hasta que el eje de posicionado indicado y programado con POSA en una secuencia CN anterior haya alcanzado su posición final (con parada precisa fina).		
	<Eje>:	Nombre del eje (identificador de eje geométrico o de canal) al que debe aplicarse el comando WAITP	
	Nota: WAITP permite habilitar (por PLC) un eje como eje de vaivén o para el desplazamiento como eje de posicionado concurrente.		
WAITMC:	Espera hasta que se alcance la meta de espera indicada Una vez alcanzada la meta de espera se cambia enseguida a la siguiente secuencia CN.		
	<Meta de espera>:	Número de la meta de espera	

⚠ PRECAUCIÓN

Desplazamiento con POSA

Si en la secuencia siguiente se lee un comando que implícitamente provoca la decodificación interna, entonces, ésta solo se ejecuta cuando se hayan ejecutado completamente todas las secuencias previamente decodificadas y memorizadas. La secuencia anterior se detiene en la parada precisa (como con G9).

Ejemplos

Ejemplo 1: desplazamiento con POSA y acceso a los datos de estado de la máquina

Cuando se accede a los datos de estado de la máquina ($A...), el control numérico genera una parada interna de decodificación previa. El mecanizado se detiene hasta que se ejecutan por completo todas las secuencias preparadas y memorizadas previamente.

```
Código de programa                                  Comentarios
N40 POSA[X]=100
N50 IF $AA_IM[X]==R100 GOTOF LABEL1                 ; Acceso a los datos de estado de
                                                      la máquina.
N60 G0 Y100
N70 WAITP(X)
N80 MARCA1:
N...
```

Ejemplo 2: espera hasta el fin del desplazamiento con WAITP

Dispositivo de alimentación de palets

Eje U: Memoria de palets
Transporte del palet de piezas a la zona de trabajo

Eje V: Sistema de transferencia a una estación de medida en el que se llevan a cabo controles aleatorios simultáneos al proceso

Código de programa	Comentarios
`N10 FA[U]=100 FA[V]=100`	`; Parámetros de avance específicos para los diferentes ejes de posicionado U y V.`
`N20 POSA[V]=90 POSA[U]=100 G0 X50 Y70`	`; Desplazamiento de ejes de posicionado y de contorneado.`
`N50 WAITP(U)`	`; La ejecución del programa continúa una vez que el eje U ha alcanzado la posición programada en N20.`
…	

Información adicional

Desplazamiento con POSA

`POSA` no afecta al cambio de secuencia ni a la ejecución del programa. El desplazamiento al punto final puede tener lugar paralelamente a la ejecución de las secuencias CN subsiguientes.

Desplazamiento con POS

El cambio de secuencia sólo se ejecuta cuando todos los ejes programados en `POS` han alcanzado su posición final.

Espera hasta el fin del desplazamiento con WAITP

Tras un comando `WAITP`, el eje se considerará no ocupado por el programa CN hasta que sea programado nuevamente. Este eje se puede activar luego como eje de posicionado a través del PLC o como eje de vaivén a través del programa CN/PLC o HMI.

Cambio de secuencia en la rampa de frenado con IPOBRKA y WAITMC

Un eje sólo se frena si todavía no se ha alcanzado la meta de espera o si otro criterio de fin de secuencia impide el cambio de secuencia. Después de un comando `WAITMC`, el eje arranca inmediatamente si no existe otro criterio de fin de secuencia que impida el cambio de secuencia.

2.7.3 Servicio de cabezal regulado en posición (SPCON, SPCOF)

Mediante los comandos SPCON y SPCOF se activa y desactiva explícitamente el modo de regulación de posición del cabezal.

Nota

La activación del modo de regulación de posición con SPCON requiere como máximo tres ciclos del regulador de posición.

Sintaxis

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

Descripción

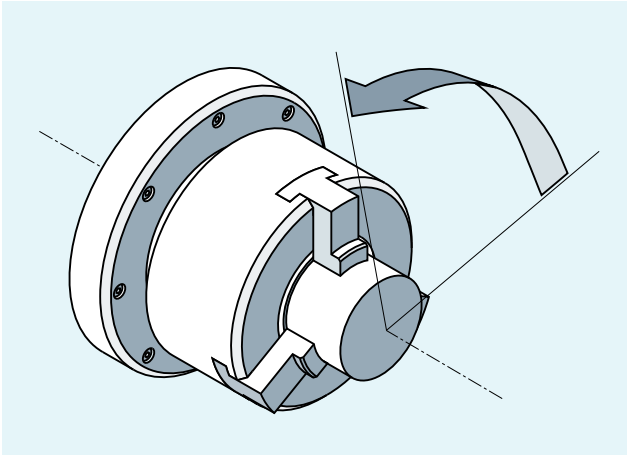
`SPCON`:	Activación del modo de regulación de posición El cabezal indicado pasa de la regulación de velocidad a la regulación de posición. `SPCON` actúa de forma modal y se mantiene hasta `SPCOF`.
`SPCOF`:	Desactivación del modo de regulación de posición El cabezal indicado pasa de la regulación de posición a la regulación de velocidad.
`<n>,<m>...`:	N.º de cabezal Si no se indica ningún número de cabezal: cabezal maestro del canal

Nota

En un cabezal síncrono con acoplamiento de valores de consigna, el cabezal maestro no debe conmutarse al modo con regulación de velocidad mediante SPCOF.

2.7.4 Posicionamiento de cabezales (SPOS, SPOSA, M19, M70, WAITS)

SPOS, SPOSA y M19 permiten colocar cabezales en determinadas posiciones angulares, p. ej., en el cambio de herramienta.



SPOS, SPOSA y M19 provocan un cambio temporal al modo de regulación de posición hasta el siguiente M3/M4/M5/M41…M45.

Posicionamiento en el modo Eje

El cabezal también se puede desplazar haciendo referencia a la dirección indicada en su dato de máquina como eje de contorneado, eje síncrono o eje de posicionado. Cuando se especifica el identificador del eje, automáticamente se utiliza el cabezal como si fuese un eje. Con M70 se conmuta el cabezal directamente al modo Eje.

Fin del posicionado

El criterio de fin del movimiento durante el posicionamiento del cabezal puede programarse mediante FINEA, CORSEA, IPOENDA o IPOBRKA.

Cuando se cumplen los criterios de fin de movimiento para todos los cabezales o ejes procesados en la secuencia y, además, el criterio de cambio de secuencia para la interpolación de la trayectoria, se produce el cambio de secuencia.

Sincronización

Para sincronizar movimientos de cabezal, se puede esperar con WAITS hasta alcanzar la posición del cabezal.

Requisitos

El cabezal que se desee posicionar debe poder funcionar en el modo de regulación de posición.

Sintaxis

Posicionamiento del cabezal:

```
SPOS=<Valor>/SPOS[<n>]=<Valor>
SPOSA=<Valor>/SPOSA[<n>]=<Valor>
```

`M19/M<n>=19`

Conmutación del cabezal al modo Eje:

`M70/M<n>=70`

Definición del criterio de fin del movimiento:

`FINEA/FINEA[S<n>]`

`COARSEA/COARSEA[S<n>]`

`IPOENDA/IPOENDA[S<n>]`

`IPOBRKA/IPOBRKA(<Eje>[,<Instante>])`; ¡Programación en la secuencia CN propia!

Sincronización de movimientos del cabezal:

`WAITS/WAITS(<n>,<m>)` ; ¡Programación en la secuencia CN propia!

Descripción

`SPOS/SPOSA`:	Posicionamiento del cabezal en la posición angular indicada SPOS y SPOSA tienen las mismas funciones, pero su comportamiento es distinto a la hora de cambiar de secuencia: • Con SPOS no se pasa a la siguiente secuencia CN hasta que se ha alcanzado la posición. • Con SPOSA se pasa a la siguiente secuencia CN aunque no se haya alcanzado la posición.			
	`<n>`:	Número del cabezal que debe posicionarse. Si no se indica ningún número de cabezal o este es "0", SPOS y SPOSA se refieren al cabezal maestro.		
	`<Valor>`:	Posición angular en la que se desea colocar el cabezal		
		Unidad:	Grados	
		Tipo:	REAL	
		La programación del modo de aproximación a la posición ofrece las siguientes posibilidades:		
		`=AC(<Valor>)`:	Acotado absoluto	
			Rango de valores:	0 … 359,9999
		`=IC(<Valor>)`:	Acotado incremental	
			Rango de valores:	0 … ±99 999,999
		`=DC(<Valor>)`:	Aproximación por la vía directa al valor absoluto	
		`=ACN(<Valor>)`:	Acotado absoluto, aproximación en sentido negativo	
		`=ACP(<Valor>)`:	Acotado absoluto, aproximación en sentido positivo	
		`=<Wert>`:	Como DC(<Valor>)	
`M<n>=19`:	Posicionar el cabezal maestro (M19 o M0=19) o el cabezal con el número <n> (M<n>=19) en la posición angular definida con SD43240 $SA_M19_SPOS utilizando el modo de aproximación a la posición configurado en SD43250 $SA_M19_SPOSMODE La secuencia CN no cambia hasta que se ha alcanzado la posición.			

M<n>=70:	Conmutar al modo Eje el cabezal maestro (M70 o M0=70) o el cabezal con el número <n> (M<n>=70) No hay aproximación a ninguna posición definida. La secuencia CN se cambia cuando se ha ejecutado la conmutación			
FINEA:	Fin del movimiento al alcanzar la "Parada precisa fina"			
COARSEA:	Fin del movimiento al alcanzar la "Parada precisa basta"			
IPOENDA:	Fin del movimiento al alcanzar la "Parada interpolador"			
S<n>:	Cabezal al que debe aplicarse el criterio de fin del movimiento programado			
	<n>:	N.º de cabezal		
	Si no se indica ningún cabezal [S<n>] o el número de cabezal es "0", el criterio de fin del movimiento programado se refiere al cabezal maestro.			
IPOBRKA:	Es posible el cambio de secuencia en la rampa de frenado			
	<Eje>:	Identificador de eje de canal		
	<Instante>:	Momento del cambio de secuencia, referido a la rampa de frenado		
		Unidad:	Porcentaje	
		Rango de valores:	100 (momento de inicio de la rampa de frenado) … 0 (fin de la rampa de frenado)	
		Si no se define el parámetro <Instante>, se aplica el valor actual del dato de operador: SD43600 $SA_IPOBRAKE_BLOCK_EXCHANGE **Nota:** IBOBRKA con instante "0" es idéntico a IPOENDA.		
WAITS:	Comando de sincronización para los cabezales indicados Al ejecutar las siguientes secuencias se espera hasta que los cabezales indicados y programados con SPOSA en una secuencia CN anterior hayan alcanzado su posición (con parada precisa fina).			
	WAITS tras M5:	Espera hasta que los cabezales indicados se paren.		
	WAITS tras M3/M4:	Espera hasta que los cabezales indicados hayan alcanzado su velocidad de consigna.		
	<n>,<m>:	Números de los cabezales a los que debe aplicarse el comando de sincronización Si no se indica ningún número de cabezal o este es "0", WAITS se refiere al cabezal maestro.		

Nota

Por cada secuencia CN se pueden posicionar hasta 3 cabezales.

Nota

En el acotado incremental IC(<Valor>), el posicionamiento del cabezal puede ser de varias vueltas.

Nota

Si antes de programar SPOS se ha activado la regulación de posición con SPCON, esta se mantiene hasta que se programe SPCOF.

Nota

El control reconoce automáticamente el paso al modo Eje basándose en la secuencia de programación. Por este motivo, en principio ya no es necesaria la programación explícita de M70 en el programa de pieza. No obstante, M70 puede seguir programándose para, p. ej., mejorar la legibilidad del programa de pieza.

Información adicional

Posicionamiento con SPOSA

SPOSA no afecta al cambio de secuencia ni al desarrollo del programa de pieza. El cabezal se puede posicionar en paralelo con la ejecución de las secuencias de programa de CN siguientes. El cambio de secuencia tiene lugar cuando todas las funciones programadas en la secuencia (con excepción del cabezal) hayan alcanzado su criterio de fin de secuencia. El posicionamiento del cabezal se puede extender en varias secuencias (ver WAITS).

Nota

Si en una secuencia subsiguiente se lee un comando que genere implícitamente una parada de decodificación, la ejecución se detendrá en esta secuencia hasta que todos los cabezales a posicionar estén parados.

Posicionamiento con SPOS/M19

La conmutación de secuencia sólo se ejecuta cuando todas las funciones programadas en la secuencia han alcanzado su criterio de fin de secuencia (p. ej., todas las funciones auxiliares han sido confirmadas por el PLC, todos los ejes han alcanzado su punto final) y cuando el cabezal ha alcanzado la posición programada.

Velocidad de los movimientos:

La velocidad y el comportamiento de deceleración para el posicionamiento están guardados en los datos de máquina. Los valores configurados pueden modificarse mediante programación o acciones síncronas; ver:

- Avance para ejes de posicionado/cabezales (FA, FPR, FPRAON, FPRAOF) (Página 126)
- Corrección programable de la aceleración (ACC) (Página 130)

Indicación de las posiciones del cabezal:

Dado que los comandos `G90`/`G91` no actúan aquí, se aplican explícitamente los respectivos acotados, como p. ej. `AC`, `IC`, `DC`, `ACN`, `ACP`. Si no se indica nada, se toma automáticamente el desplazamiento `DC`.

Sincronización de movimientos del cabezal con WAITS

`WAITS` permite definir en el programa CN un punto en el cual debe esperarse hasta que uno o varios cabezales programados con `SPOSA` en una secuencia CN anterior hayan alcanzado su posición.

Ejemplo:

Código de programa	Comentarios
`N10 SPOSA[2]=180 SPOSA[3]=0`	
`...`	
`N40 WAITS(2,3)`	`; En la secuencia se espera hasta que los cabezales 2 y 3 hayan alcanzado las posiciones programadas para ellos en la secuencia N10.`

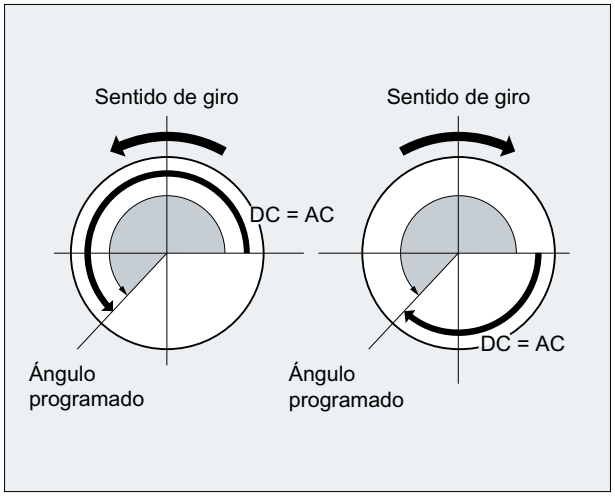
Después de `M5` puede esperarse con `WAITS` a que se paren los cabezales. Después de `M3`/`M4` puede esperarse con `WAITS` a que los cabezales alcancen la velocidad o el sentido de giro especificados.

Nota

En el caso de que el cabezal aún no haya sido sincronizado, se toma como sentido de giro positivo el definido mediante datos de máquina (estado de suministro).

Posicionar el cabezal cuando está girando (M3/M4)

Si se activa `M3` o `M4`, el cabezal se detiene en el valor programado.



No hay ninguna diferencia entre los parámetros `DC` y `AC`. En ambos casos el cabezal gira en el sentido indicado mediante `M3`/`M4` hasta que alcanza la posición final absoluta. Con `ACN` y `ACP` se frena y se mantiene, en caso necesario, la dirección de aproximación correspondiente. Si se configura `IC`, el cabezal gira adicionalmente hasta el valor indicado partiendo de la posición en la que se encuentra actualmente.

Posicionar el cabezal a partir de la posición de parada (M5)

Se recorre exactamente la distancia programada partiendo de la posición de parada (`M5`).

2.7.5 Avance para ejes de posicionado/cabezales (FA, FPR, FPRAON, FPRAOF)

Además, existe la posibilidad de calcular el avance por vuelta para ejes síncronos y de contorneado o para los diferentes ejes de posicionado/cabezales tomando como referencia otro eje giratorio o cabezal.

Los ejes de posicionado, tales como los sistemas de transporte de piezas, las torretas o las lunas, se desplazan de forma independiente a los ejes síncronos o de contorneado. Por eso se define un avance propio para cada eje de posicionado.

También puede programarse un avance por eje propio para los cabezales.

Sintaxis

Avance para eje de posicionado:
`FA[<Eje>]=…`

Avance axial para cabezal:
`FA[SPI(<n>)]=…`
`FA[S<n>]=…`

Cálculo del avance por vuelta para ejes de contorneado/síncronos:

```
FPR(<Eje giratorio>)
FPR(SPI(<n>))
FPR(S<n>)
```

Cálculo del avance por vuelta para ejes de posicionado/cabezales:

```
FPRAON(<Eje>,<Eje giratorio>)
FPRAON(<Eje>,SPI(<n>))
FPRAON(<Eje>,S<n>)
FPRAON(SPI(<n>),<Eje giratorio>)
FPRAON(S<n>,<Eje giratorio>)
FPRAON(SPI(<n>),SPI(<n>))
FPRAON(S<n>,S<n>)
FPRAOF(<Eje>,SPI(<n>),…)
FPRAOF(<Eje>,S<n>,…)
```

Descripción

`FA[...]=...`:	Avance para el eje de posicionado indicado o velocidad de posicionado (avance axial) para el cabezal indicado	
	Unidad:	mm/min, pulgadas/min o grados/min
	Rango de valores:	… 999 999,999 mm/min, grados/min … 39 999,9999 pulgadas/min
`FPR(...)`:	`FPR` identifica el eje giratorio (`<Eje giratorio>`) o el cabezal (`SPI(<n>)`/`S<n>`) que sirve de referencia para calcular el avance por vuelta programado en `G95` para los ejes de contorneado y síncronos.	

FPRAON(...):	Cálculo del avance por vuelta para ejes de posicionado y cabezales El primer parámetro (<Eje>/SPI(<n>)/S<n>) identifica el eje de posicionado/cabezal que debe desplazarse con el avance por vuelta. El segundo parámetro (<Eje giratorio>/SPI(<n>)/S<n>) identifica el eje giratorio/cabezal que sirve como referencia para calcular el avance por vuelta. **Nota:** El segundo parámetro también puede omitirse; en tal caso, para calcular el avance se tomará como referencia el cabezal maestro.
FPRAOF(...):	FPRAOF permite desactivar el avance por vuelta calculado para los ejes o cabezales indicados.
<Eje>:	Identificador de eje (eje geométrico o de posicionado)
SPI(<n>)/S<n>:	Identificador de cabezal SPI(<n>) y S<n> son funcionalmente idénticos.
	<n>: Número de cabezal
	Nota: SPI convierte el número de cabezal en un identificador de eje. El parámetro de transferencia (<n>) debe contener un número de cabezal válido.

Nota

El valor programado para el avance FA[...] es modal.

Por cada secuencia se pueden programar como máximo 5 avances para ejes de posicionado/cabezales.

Nota

El valor de avance se calcula atendiendo a la siguiente fórmula:

Avance calculado =
avance programado * valor avance de referencia

Ejemplos

Ejemplo 1: acoplamiento de cabezales síncronos

En el acoplamiento de cabezales síncronos es posible programar la velocidad de posicionado del cabezal esclavo independientemente del cabezal maestro, p. ej. para el posicionamiento.

```
Código de programa   Comentarios
...
FA[S2]=100           ; Velocidad de posicionado del cabezal esclavo (cabezal 2)
                     = 100 grados/min
...
```

Ejemplo 2: avance por vuelta calculado para ejes de contorneado

Los ejes de contorneado X, Y deben desplazarse con avance por vuelta; como referencia se tomará la velocidad del eje giratorio A:

Código de programa
...
N40 FPR(A)
N50 G95 X50 Y50 F500
...

Ejemplo 3: cálculo del avance por vuelta para el cabezal maestro

Código de programa	**Comentarios**
N30 FPRAON(S1,S2)	; El avance por vuelta para el cabezal maestro (S1) debe calcularse tomando como referencia el cabezal 2.
N40 SPOS=150	; Posicionar el cabezal maestro.
N50 FPRAOF(S1)	; Deseleccionar el avance por vuelta calculado para el cabezal maestro.

Ejemplo 4: cálculo del avance por vuelta para el eje de posicionado

Código de programa	**Comentarios**
N30 FPRAON(X)	; El avance por vuelta para el eje de posicionado X debe calcularse tomando como referencia el cabezal maestro.
N40 POS[X]=50 FA[X]=500	; El eje de posicionado se desplaza a 500 mm/vuelta del cabezal maestro.
N50 FPRAOF(X)	

Información adicional

FA[...]

Siempre se aplica el tipo de avance `G94`. Si `G70`/`G71` están activos, la unidad de medida expresada en el sistema métrico/en pulgadas depende del ajuste previo en el dato de máquina. Con `G700`/`G710` puede modificarse la unidad de medida en el programa.

Nota

Si no se programa ningún valor `FA`, se aplica el valor ajustado en el dato de máquina.

FPR(...)

`FPR`, como ampliación del comando `G95` (avance por vuelta referido al cabezal maestro), permite calcular también el avance por vuelta tomando como referencia cualquier cabezal o eje giratorio. `G95 FPR(…)` es válido para ejes síncronos y de contorneado.

En el caso de que el eje giratorio/cabezal indicado en FPR, esté trabajando en regulación de posición, se realiza el acoplamiento en valor de consigna; en caso contrario, el acoplamiento se realiza con el valor real de velocidad.

FPRAON(...)

Con `FPRAON` se puede calcular axialmente el avance por vuelta para ejes de posicionado y cabezales a partir del avance momentáneo de otro eje giratorio o cabezal.

FPRAOF(...)

Con `FPRAOF` se desactiva el avance por vuelta para uno o varios ejes/cabezales simultáneamente.

2.7.6 Corrección programable del avance (OVR, OVRRAP, OVRA)

La velocidad de los ejes de contorneado/posicionado y de los cabezales puede modificarse en el programa CN.

Sintaxis

```
OVR=<Valor>
OVRRAP=<Valor>
OVRA[<Eje>]=<Valor>
OVRA[SPI(<n>)]=<Valor>
OVRA[S<n>]=<Valor>
```

Descripción

`OVR`:	Modificación del avance para el avance de contorneado F	
`OVRRAP`:	Modificación del avance para la velocidad en rápido	
`OVRA`:	Modificación del avance para el avance de posicionado `FA` o bien para la velocidad de giro del cabezal `S`	
`<Eje>`:	Identificador de eje (eje geométrico o de posicionado)	
`SPI(<n>)/S<n>`:	Identificador de cabezal `SPI(<n>)` y `S<n>` son funcionalmente idénticos.	
	`<n>`:	Número de cabezal
	Nota: `SPI` convierte el número de cabezal en un identificador de eje. El parámetro de transferencia `(<n>)` debe contener un número de cabezal válido.	
`<Valor>`:	Modificación porcentual del avance El valor se refiere o se superpone a la corrección del avance ajustada en el panel de mando de la máquina.	
	Rango de valores:	0…200%, valor entero
	Nota: En correcciones de trayectoria y de desplazamientos en rápido no se sobrepasan las velocidades máximas definidas en los datos de máquina.	

2.7.7 Corrección programable de la aceleración (ACC)

En secciones críticas del programa puede llegar a ser necesario limitar la aceleración por debajo de los valores máximos posibles para evitar, p. ej., vibraciones mecánicas.

La corrección programable de la aceleración permite cambiar la aceleración de cada eje de contorneado o cada cabezal mediante comandos en el programa CN. La limitación actúa en todos los tipos de interpolación. Como valor 100% de aceleración se toma el indicado en los datos de máquina correspondientes.

Sintaxis

```
ACC[<Eje>]=<Valor>
ACC[SPI(<n>)]=<Valor>
ACC(S<n>)=<Valor>
```

Desactivar:
```
ACC[...]=100
```

Sintaxis

ACC:	Modificación de la aceleración para el eje de contorneado indicado o variación de la velocidad de giro para el cabezal indicado	
<Eje>:	Nombre de eje de canal del eje de contorneado	
SPI(<n>)/S<n>:	Identificador de cabezal SPI(<n>) y S<n> son funcionalmente idénticos.	
	<n>:	Número de cabezal
	Nota: SPI convierte el número de cabezal en un identificador de eje. El parámetro de transferencia (<n>) debe contener un número de cabezal válido.	
<Valor>:	Variación porcentual de la aceleración El valor se refiere o se superpone a la corrección del avance ajustada en el panel de mando de la máquina.	
	Rango de valores:	1…200%, valor entero

Nota

Si la aceleración es mayor, podrían sobrepasarse los valores máximos permitidos por el fabricante de la máquina.

Ejemplo

Código de programa	Comentarios
N50 ACC[X]=80	; El desplazamiento del carro en la dirección X debe estar limitado a una aceleración del 80%.
N60 ACC[SPI(1)]=50	; El cabezal 1 debe acelerarse o frenarse sólo con el 50% de la capacidad de aceleración.

Información adicional

Corrección de la aceleración programada con ACC

La corrección de la aceleración programada con ACC[...] se tiene en cuenta siempre en la salida, al igual que en la variable de sistema $AA_ACC. La lectura en el programa de pieza y en acciones síncronas tiene lugar en distintos momentos del mecanizado de CN.

En el programa de pieza

El valor escrito en el programa de pieza solo se tiene en cuenta en la variable de sistema $AA_ACC según lo escrito en dicho programa si, entre tanto, ACC no ha sido modificado por una acción síncrona.

En acciones síncronas

Por analogía se aplica: El valor escrito en una acción síncrona solo se tiene en cuenta en la variable de sistema $AA_ACC según lo escrito en dicha acción si, entre tanto, ACC no ha sido modificado por un programa de pieza.

La aceleración especificada puede modificarse también a través de acciones síncronas.

Ejemplo:

Código de programa

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

El valor de aceleración actual se puede averiguar con la variable de sistema $AA_ACC[<Eje>]. En un dato de máquina se puede ajustar si, con RESET/fin del programa de pieza, se debe aplicar el último valor ACC ajustado o el 100%.

2.7.8 Avance con corrección del volante (FD, FDA)

Los comandos FD y FDA permiten desplazar ejes con volantes durante la ejecución del programa de pieza. En este caso, los desplazamientos programados de los ejes se solapan con los impulsos de volante evaluados como especificación de recorrido o de velocidad.

Ejes de contorneado

En los ejes de contorneado puede superponerse el avance de contorneado programado. En tal caso, se evalúa el volante del 1.er eje geométrico del canal. Los impulsos de volante evaluados en función del sentido de giro por cada ciclo IPO coinciden con la velocidad de contorneado que se va a superponer. Los límites de la velocidad de contorneado que pueden alcanzarse con la corrección por volante son:

- Mínimo: 0
- Máximo: límites de los datos de máquina de los ejes de contorneado que intervienen en el desplazamiento

Nota

Avance de contorneado

El avance de contorneado F y el avance por volante FD no deben programarse en una misma secuencia CN.

Ejes de posicionado
En los ejes de posicionado puede superponerse por eje el recorrido o la velocidad. En este caso se evalúa el volante asignado al eje.

- Corrección de la trayectoria
  Los impulsos de volante evaluados en función del sentido de giro coinciden con el recorrido que debe realizar el eje. Sólo se tienen en cuenta los impulsos de volante en dirección a la posición programada.
- Corrección de la velocidad
  Los impulsos de volante evaluados en función del sentido de giro por cada ciclo IPO coinciden con la velocidad por eje que se va a superponer. Los límites de la velocidad de contorneado que pueden alcanzarse con la corrección por volante son:
  - Mínimo: 0
  - Máximo: límites de los datos de máquina del eje de posicionado

Sintaxis

```
FD=<Velocidad>
FDA[<Eje>]=<Velocidad>
```

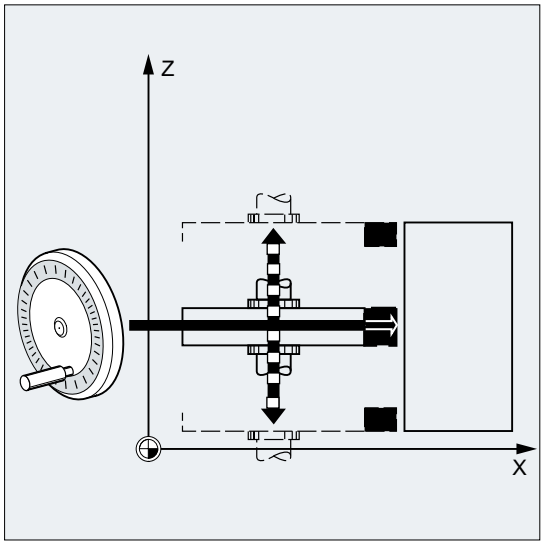
Descripción

`FD=<Velocidad>`:	Avance de contorneado y habilitación de la corrección de la velocidad por volante. <Velocidad>: • Valor = 0: ¡No permitido! • Valor ≠ 0: Velocidad de contorneado
`FDA[<Eje>]=<Velocidad>`:	Avance por eje <Velocidad>: • Valor = 0: Especificación de recorrido por volante • Valor ≠ 0: Velocidad por eje
`<Eje>`:	Identificador del eje de posicionado

Nota

`FD` y `FDA` actúan por secuencia.

Ejemplo



incremental: El eje de vaivén Z para la rectificadora se desplaza mediante el volante del eje X a una posición

en la que la emisión de chispas generadas por el contacto entre la muela y la pieza sea constante. Al activar el borrado de trayecto residual se cambia a la siguiente secuencia CN y se sigue trabajando en el modo AUTOMÁTICO.

Información adicional

Desplazamiento de ejes de contorneado con corrección de la velocidad (FD=<Velocidad>)

La secuencia del programa de pieza en la cual se ha programado la corrección de la velocidad de contorneado debe cumplir las siguientes condiciones:

- Comando de desplazamiento `G1`, `G2` o `G3` activo
- Parada precisa `G60` activa
- Avance lineal `G94` activo

Corrección del avance

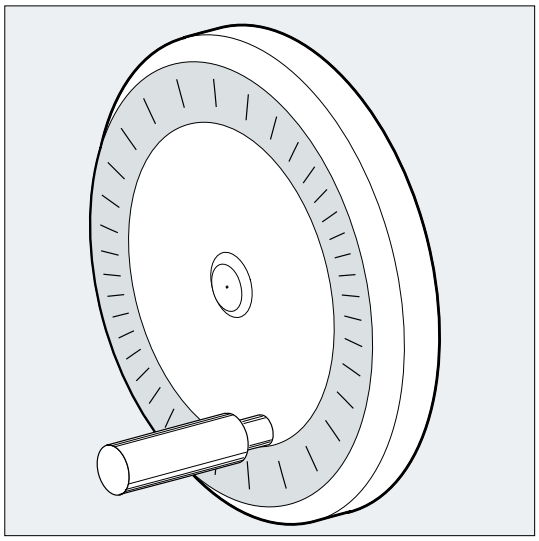
La corrección del avance solamente es válida para la velocidad de contorneado programada, no para la componente de la velocidad creada por volante (excepción: corrección del avance = 0).

Ejemplo:

Código de programa	Descripción
`N10 X… Y… F500`	`; Avance de contorneado = 500 mm/min`
`N20 X… Y… FD=700`	`; Avance de contorneado = 700 mm/min y corrección de la velocidad por volante.`
	`; En N20 se acelera de 500 a 700 mm/min. El volante permite modificar la velocidad de contorneado en función del sentido de giro entre 0 y el valor máximo (datos de máquina).`

Desplazamiento de ejes de posicionado con especificación de recorrido (FDA[<Eje>]=0)

La programación de `FDA[<Eje>]=0` en una secuencia CN provoca que el avance se ajuste a cero, de manera que no se realice ningún desplazamiento desde el programa. A partir de este momento, el desplazamiento programado a la posición de destino es controlado exclusivamente por el usuario girando el volante.



Ejemplo:

Código de programa	Descripción
...	
N20 POS[V]=90 FDA[V]=0	; Posición de destino = 90 mm, avance axial = 0 mm/min y corrección de la trayectoria por volante.
	; Velocidad del eje V al principio de la secuencia = 0 mm/min.
	; La especificación de recorrido y de velocidad se realiza mediante impulsos de volante

Sentido de desplazamiento, velocidad de desplazamiento:
Los ejes se desplazan en el sentido definido mediante impulsos de volante con el signo adecuado. En función del sentido de giro, el desplazamiento puede ser hacia delante o hacia atrás. Cuanto más rápido se gire el volante, mayor será la velocidad de desplazamiento.

Zona de desplazamiento:
La zona de desplazamiento queda delimitada por la posición inicial y el punto final programado.

Desplazamiento de ejes de posicionado con corrección de la velocidad (FDA[<Eje>]=<Velocidad>)

La programación de `FDA[…]=…` en una secuencia CN provoca que el avance se acelere o se frene desde el último valor `FA` programado hasta el valor programado en `FDA`. Partiendo del avance `FDA` actual, el movimiento programado hasta la posición de destino puede acelerarse o decelerarse hasta cero girando el volante. Como velocidad máxima se adoptan los valores parametrizados en los datos de máquina.

Ejemplo:

Código de programa	Descripción
N10 POS[V]=… FA[V]=100	; Avance axial = 100 mm/min
N20 POS[V]=100 FAD[V]=200	; Posición de destino axial = 100, avance axial = 200 mm/min ; y corrección de la velocidad por volante. ; En N20 se acelera de 100 a 200 mm/min. Con el ; volante puede modificarse la velocidad en función del ; sentido de giro entre 0 y el valor máximo (datos de máquina).

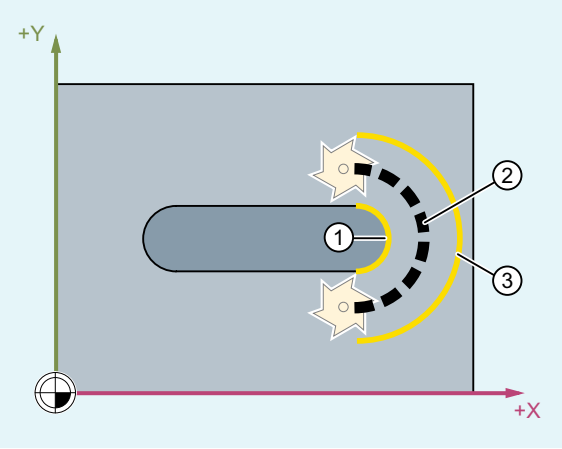
Zona de desplazamiento:
La zona de desplazamiento queda delimitada por la posición inicial y el punto final programado.

2.7.9 Optimización del avance para contornos de curvatura pronunciada (CFTCP, CFC, CFIN)

Cuando está activado el modo de corrección G41/G42 para el radio de la fresa, el avance programado se refiere primero a la trayectoria del centro de ésta (ver el capítulo "Transformaciones de coordenadas (frames) (Página 314)").

Cuando se está fresando un círculo (o al utilizar una interpolación polinómica o spline), en ciertas circunstancias el avance respecto al borde de la fresa puede experimentar variaciones tan sustanciales que éstas lleguen a afectar al resultado del mecanizado.

Ejemplo: Fresado de un radio externo pequeño con una herramienta más grande. La distancia que tiene que recorrer el punto exterior de la herramienta es mucho mayor de la que tiene que recorrer el contorno.



De esta forma se trabaja con un avance muy pequeño respecto al contorno. Para evitar este efecto, en contornos muy angulosos debería ajustarse el avance adecuadamente.

Sintaxis

```
CFTCP
```

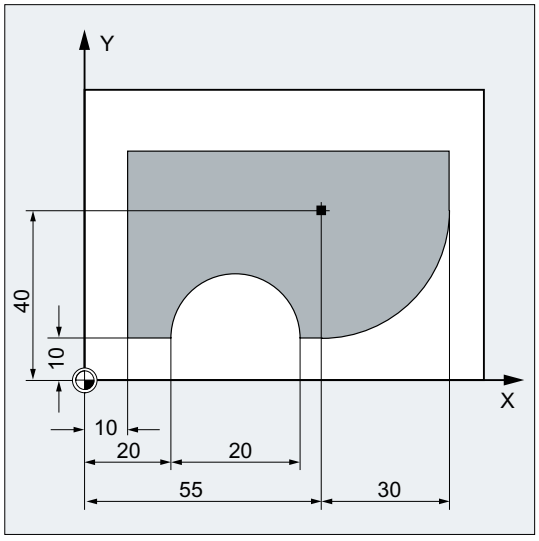
```
CFC
CFIN
```

Descripción

CFTCP:	Avance constante respecto a la trayectoria del centro de la fresa El controlador mantiene constante la velocidad de avance; las correcciones del avance se desactivan.
CFC:	Avance constante respecto al contorno (filo de la herramienta) Esta función está activa por defecto.
CFIN:	Avance constante respecto al filo de la herramienta sólo en contornos con curvatura interior; en caso contrario, el avance se refiere a la trayectoria del centro de la fresa La velocidad de avance se reduce para radios interiores.

Ejemplo

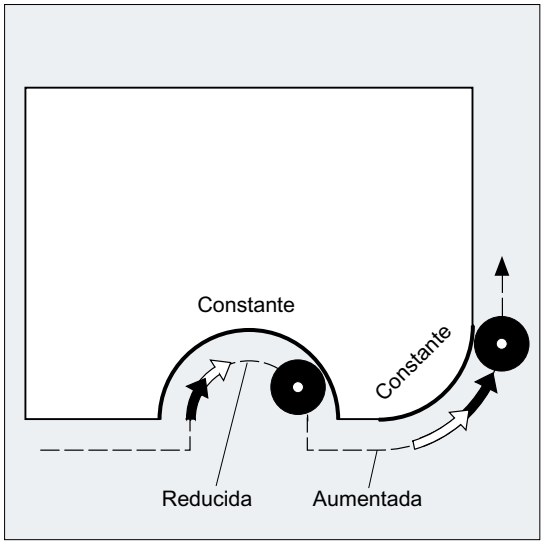
En este ejemplo, primero se mecaniza un contorno con corrección de la velocidad de avance (CFC). En el acabado se mecaniza adicionalmente el área ya mecanizada con CFIN. De este modo, se evita estropear la pieza al mecanizar los radios exteriores debido a una velocidad de avance demasiado elevada.



Código de programa	**Comentarios**
N10 G17 G54 G64 T1 M6	
N20 S3000 M3 CFC F500 G41	
N30 G0 X-10	
N40 Y0 Z-10	; Penetración hasta primera profundidad de trabajo
N50 KONTUR1	; Llamada de subprograma
N40 CFIN Z-25	; Penetración hasta segunda profundidad de trabajo
N50 KONTUR1	; Llamada de subprograma
N60 Y120	
N70 X200 M30	

Información adicional

Avance constante en el contorno con CFC



La velocidad de avance se reduce para radios interiores, en radios exteriores ésta se incrementa. De este modo, se mantiene una velocidad constante entre la punta de la herramienta y el contorno de la pieza.

2.7.10 Varios valores de avance en una secuencia (F, ST, SR, FMA, STA, SRA)

La función "Varios valores del avance en una secuencia" permite activar en sincronización con el desplazamiento diversos valores de avance de una secuencia CN en función de las entradas externas digitales o analógicas, así como el tiempo de espera y la retirada

Sintaxis

Movimiento interpolado:
```
F=... F7=... F6=... F5=... F4=... F3=... F2=... ST=... SR=...
```

Movimiento por eje:
```
FA[<Ax>]=... FMA[7,<Ax>]=... FMA[6,<Ax>]=... FMA[5,<Ax>]=...
FMA[4,<Ax>]=... FMA[3,<Ax>]=... FMA[2,<Ax>]=... STA[<Ax>]=...
SRA[<Ax>]=...
```

Descripción

`F=...`:	En la dirección `F` se programa el avance de contorneado que será válido mientras no haya ninguna señal de entrada.	
	Eficacia:	modal

F2=... a F7=... :	Además del avance de contorneado pueden programarse hasta 6 avances más en la secuencia. La extensión numérica indica el número de bit de la entrada con cuya modificación se activa el avance.	
	Eficacia:	secuencia a secuencia
ST=...:	Tiempo de espera en s (en la tecnología Rectificado: tiempo de afinado)	
	Bit de entrada:	1
	Eficacia:	secuencia a secuencia
SR=...:	Distancia de retirada La unidad para la distancia de retirada se refiere a la unidad de medida actualmente válida (mm o pulgadas).	
	Bit de entrada:	0
	Eficacia:	secuencia a secuencia
FA[<Ax>]=... :	En la dirección FA se programa el avance axial que será válido mientras no haya ninguna señal de entrada.	
	Eficacia:	modal
FMA[2,<Ax>]=... a FMA[7,<Ax>]=... :	Además del avance axial FA, con FMA pueden programarse hasta 6 avances más por eje en la secuencia. El primer parámetro indica el número de bit de la entrada; el segundo, el eje para el cual será válido el avance.	
	Eficacia:	secuencia a secuencia
STA[<Ax>]=...:	Tiempo de espera por eje en s (en la tecnología Rectificado: tiempo de afinado)	
	Bit de entrada:	1
	Eficacia:	secuencia a secuencia
SRA[<Ax>]=...:	Distancia de retirada por eje	
	Bit de entrada:	0
	Eficacia:	secuencia a secuencia
<Ax>:	Eje para el que debe aplicarse el avance	

Nota

Prioridad de las señales

El orden de consulta de las señales es a partir del bit de entrada 0 (E0) en sentido ascendente. De esta forma, el movimiento de retroceso tiene la prioridad más alta, y el avance F7, la más baja. El tiempo de espera y el movimiento de retroceso finalizan los desplazamientos de avance que se activaron con F2 a F7.

La señal con la prioridad más alta determina el avance actual.

Nota

Borrado de trayecto residual

Si se activa la entrada bit 1 para el tiempo de espera o la distancia de retirada bit 0, se borra el trayecto residual para ejes de contorneado o los correspondientes ejes individuales y se inicia el tiempo de espera o la retirada.

Nota

Distancia de retirada

La unidad para la distancia de retirada se refiere a la unidad de medida actualmente válida (mm o pulgadas).

La retirada se produce siempre en el sentido contrario al movimiento actual. Con `SR`/`SRA` se programa siempre el valor de la retirada. Se programa siempre sin signo.

Nota

POS en lugar de POSA

Si, para un eje, se han programado avances, tiempo de espera o distancia de retirada en base a una entrada externa, este eje no se debe programar en esta secuencia como eje POSA (eje de posicionamiento más allá de límites de secuencia).

Nota

Consulta de estado

El estado de una entrada también puede consultarse para comandos síncronos de distintos ejes.

Nota

LookAhead

LookAhead también está activo con varios avances en una secuencia. De este modo, el avance actual puede ser limitado por LookAhead.

Ejemplos

Ejemplo 1: Movimiento interpolado

Código de programa	**Comentarios**

`G1 X48 F1000 F7=200 F6=50 F5=25 F4=5 ST=1,5 SR=0,5`	`; Avance de contorneado = 1000`
	`; Valores adicionales de avance de contorneado:`
	`; 200 (bit de entrada 7)`
	`; 50 (bit de entrada 6)`
	`; 25 (bit de entrada 5)`
	`; 5 (bit de entrada 4)`
	`; Tiempo de espera 1,5 s`
	`; Retirada 0,5 mm`

Ejemplo 2: Movimiento por eje

Código de programa	Comentarios
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 para eje A = 800 ; Valores adicionales de avance para el eje A: 720 (bit de entrada 7) ; 640 (bit de entrada 6) ; 560 (bit de entrada 5) ; Tiempo de espera por eje: 1,5 s ; Retirada por eje: 0,5 mm

Ejemplo 3: varias operaciones en una secuencia

Código de programa	Comentarios
N20 T1 D1 F500 G0 X100	Posición inicial
N25 G1 X105 F=20 F7=5 F3=2.5 F2=0.5 ST=1.5 SR=0.5	; Avance normal con F, ; desbaste con F7, ; acabado con F3, ; acabado fino con F2, ; tiempo de espera 1,5 s, ; distancia de retirada 0,5 mm
...	

2.7.11 Avance por secuencia (FB)

La función "Avance por secuencia" permite especificar un avance independiente para una determinada secuencia. Después de esta secuencia se activa nuevamente el avance modal que se encontraba anteriormente activo.

Sintaxis

```
FB=<Valor>
```

Descripción

FB:	Avance sólo para la secuencia actual
<VALOR>:	El valor programado debe ser superior a cero. La interpolación se realiza conforme al tipo de avance activo: • G94: avance en mm/min o grados/min • G95: avance en mm/vuelta o pulgada/vuelta • G96: velocidad de corte constante

Nota

Si en la secuencia no hay programado ningún desplazamiento (p. ej., secuencia de cálculo), `FB` no tiene efecto alguno.

Si no se ha programado ningún avance explícito para chaflán/redondeo, el valor de `FB` se aplicará también a un elemento de contorno chaflán/redondeo existente en esa secuencia.

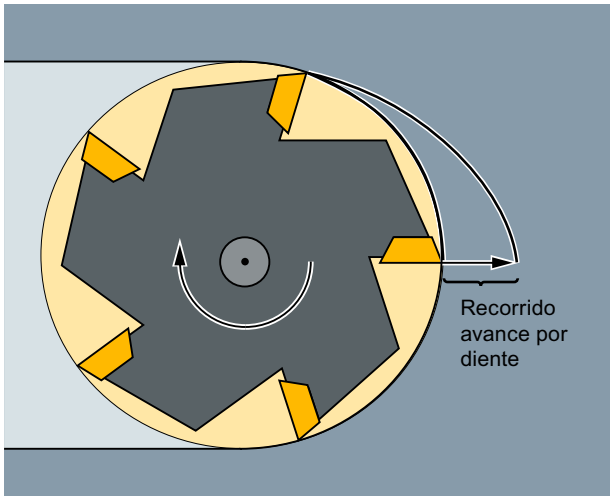
Las interpolaciones de avance `FLIN`, `FCUB`... no tienen restricciones.

La programación simultánea de `FB` y `FD` (desplazamiento manual con volante con corrección del avance) o `F` (avance de contorneado modal) **no** es posible.

Ejemplo

Código de programa	Comentarios
`N10 G0 X0 Y0 G17 F100 G94`	`; Posición inicial`
`N20 G1 X10`	`; Avance 100 mm/min`
`N30 X20 FB=80`	`; Avance 80 mm/min`
`N40 X30`	`; El avance vuelve a ser de 100 mm/min`
`...`	

2.7.12 Avance por diente (G95 FZ)



Mediante el parámetro de herramienta $TC_DPNT (número de dientes) del bloque de datos de corrección de herramienta activo, el control calcula el avance por vuelta aplicable a partir del avance por diente programado para cada secuencia de desplazamiento:

F = FZ * $TC_DPNT

donde: F: Avance por vuelta en mm/vuelta o pulgadas/vuelta
FZ: Avance por diente en mm/diente o pulgadas/diente
$TC_DPNT: Variable de sistema Parámetros de herramienta: número de dientes/vuelta

El tipo ($TC_DP1) de la herramienta activa no se tiene en cuenta.

El avance por diente programado es independiente del cambio de herramienta y de la activación/desactivación de un bloque de datos de corrección de herramienta, y se mantiene modalmente.

Una modificación del parámetro de herramienta $TC_DPNT del filo activo pasa a estar activa con la siguiente activación de la corrección de herramienta o con la siguiente actualización de los datos de corrección activos.

El cambio de herramienta y la activación/desactivación de un bloque de datos de corrección de herramienta provocan que se vuelva a calcular el avance por vuelta activo.

Nota

El avance por diente sólo se refiere a la trayectoria; no es posible una programación específica del eje.

Sintaxis

```
G95 FZ...
```

Descripción

G95:	Tipo de avance: avance por vuelta en mm/vuelta o pulgadas/vuelta (dependiendo de G700/G710) Respecto a G95, ver "Avance (G93, G94, G95, F, FGROUP, FL, FGREF) (Página 108)"	
FZ:	Velocidad de avance por diente	
	Activación:	con G95
	Efecto:	modal
	Unidad de medida:	mm/diente o pulgadas/diente (dependiendo de G700/G710)

ATENCIÓN

Cambio de herramienta/cambio del cabezal maestro

El usuario debe tener en cuenta mediante la programación correspondiente (p. ej., reprogramación de FZ) el subsiguiente cambio de herramienta o del cabezal maestro.

ATENCIÓN

Intervenciones de la herramienta no definidas

Los aspectos tecnológicos, tales como el fresado en concordancia o en oposición, el fresado frontal o el planeado, no se tienen en cuenta automáticamente, como tampoco la geometría de la trayectoria (recta, círculo...). Por ello debe prestarse atención a estos factores a la hora de programar el avance por diente.

Nota

Conmutación entre G95 F... y G95 FZ...

Con conmutación entre `G95 F...` (avance por vuelta) y `G95 FZ...` (avance por diente) y se borra el valor de avance no activo en cada caso.

Nota

Cálculo del avance con FPR

`FPR` permite calcular también el avance por diente tomando como referencia cualquier eje giratorio o cualquier cabezal, de la misma forma que para el avance por vuelta (ver "Avance para ejes de posicionado/cabezales (FA, FPR, FPRAON, FPRAOF) (Página 126)").

Ejemplos

Ejemplo 1: Fresa de 5 dientes ($TC_DPNT = 5)

Código de programa	Comentarios
N10 G0 X100 Y50	
N20 G1 G95 FZ=0.02	; Avance por diente: 0,02 mm/diente
N30 T3 D1	; Cargar herramienta y activar bloque de datos de corrección de herramienta.
M40 M3 S200	; Velocidad de giro del cabezal: 200 r/min
N50 X20	; Fresado con:
	FZ = 0,02 mm/diente
	avance por vuelta activo:
	F = 0,02 mm/diente * 5 dientes/vuelta = 0,1 mm/vuelta
	o bien
	F = 0,1 mm/vuelta * 200 rpm = 20 mm/min
...	

Ejemplo 2: Conmutación entre G95 F... y G95 FZ...

Código de programa	Comentarios
N10 G0 X100 Y50	
N20 G1 G95 F0.1	; Avance por vuelta 0,1 mm/vuelta
N30 T1 M6	
N35 M3 S100 D1	
N40 X20	

Código de programa	Comentarios
N50 G0 X100 M5	
N60 M6 T3 D1	; Cargar herramienta con, p. ej., 5 dientes ($TC_DPNT = 5).
N70 X22 M3 S300	
N80 G1 X3 G95 FZ=0.02	; Cambio de G95 F… a G95 FZ…, avance por diente a 0,02 mm/diente, activo.
...	

Ejemplo 3: cálculo del avance por diente tomando como referencia un cabezal (FBR)

Código de programa	Comentarios
…	
N41 FPR(S4)	; Herramienta en cabezal 4 (no cabezal maestro).
N51 G95 X51 FZ=0.5	; Avance por diente: 0,5 mm/diente en función del cabezal S4.
...	

Ejemplo 4: cambio subsiguiente de herramienta

Código de programa	Comentarios
N10 G0 X50 Y5	
N20 G1 G95 FZ=0.03	; Avance por diente: 0,03 mm/diente
N30 M6 T11 D1	; Cargar herramienta con, p. ej., 7 dientes ($TC_DPNT = 7).
N30 M3 S100	
N40 X30	; Avance por vuelta activo: 0,21 mm/vuelta
N50 G0 X100 M5	
N60 M6 T33 D1	; Cargar herramienta con, p. ej., 5 dientes ($TC_DPNT = 5).
N70 X22 M3 S300	
N80 G1 X3	; Avance por diente modal 0,03 mm/diente, avance efectivo por vuelta 0,15 mm/vuelta
...	

Ejemplo 5: cambio del cabezal maestro

Código de programa	Comentarios
N10 SETMS(1)	; El cabezal 1 es el cabezal maestro.
N20 T3 D3 M6	; La herramienta 3 se cambia al cabezal 1.
N30 S400 M3	; Velocidad de giro S400 del cabezal 1 (y, por tanto, T3).
N40 G95 G1 FZ0.03	; Avance por diente: 0,03 mm/diente
N50 X50	; Desplazamiento de trayectoria; el avance activo depende de: - El avance por diente FZ - La velocidad de giro del cabezal 1 - El número de dientes de la herramienta activa T3
N60 G0 X60	

Código de programa	Comentarios
...	
N100 SETMS(2)	; El cabezal 2 pasa a ser el cabezal maestro.
N110 T1 D1 M6	; La herramienta 1 se cambia al cabezal 2.
N120 S500 M3	; Velocidad de giro S500 del cabezal 2 (y, por tanto, T1).
N130 G95 G1 FZ0.03 X20	; Desplazamiento de trayectoria; el avance activo depende de: - El avance por diente FZ - La velocidad de giro del cabezal 2 - El número de dientes de la herramienta activa T1

Nota

Tras el cambio del cabezal maestro (N100) debe cargarse una herramienta (N110) accionada por el cabezal 2.

Información adicional

Cambio entre G93, G94 y G95

FZ puede programarse también sin estar activo G95, aunque no tiene ningún efecto, y se borra seleccionando G95; es decir, cambiando entre G93, G94 y G95 se borra, de la misma forma que el valor F, también el valor FZ.

Nueva selección de G95

Una nueva selección de G95 no tiene ningún efecto si G95 ya está activo (en caso de que no se haya programado ningún cambio entre F y FZ).

Avance activo por secuencia (FB)

Un avance activo por secuencia FB... se interpreta como avance por diente cuando G95 FZ... está activo (modalmente).

Mecanismo SAVE

En subprogramas con el atributo SAVE, FZ se escribe con el valor anterior al inicio del subprograma de la misma forma que sucede con F.

Varios valores del avance en una secuencia

En el avance por diente no es posible la función "Varios valores del avance en una secuencia".

Acciones síncronas

No se puede especificar FZ a partir de acciones síncronas.

Lectura de la velocidad de avance por diente y del tipo de avance de contorneado

La velocidad de avance por diente y el tipo de avance de contorneado pueden leerse mediante variables de sistema:

- Con la parada de decodificación previa en el programa de pieza mediante variantes de sistema:

	$AC_FZ	Velocidad de avance por diente que estaba activa en el momento de preparar la secuencia del proceso principal actual.	
	$AC_F_TYPE	Tipo de avance de contorneado que estaba activo en el momento de preparar la secuencia del proceso principal actual.	
		Valor:	**Significado:**
		0	mm/min
		1	mm/vuelta
		2	pulgadas/min
		3	pulgadas/vuelta
		11	mm/diente
		33	pulgadas/diente

- Sin la parada de decodificación previa en el programa de pieza mediante variantes de sistema:

	$P_FZ	Velocidad de avance por diente programada	
	$P_F_TYPE	Tipo de avance de contorneado programado	
		Valor:	**Significado:**
		0	mm/min
		1	mm/vuelta
		2	pulgadas/min
		3	pulgadas/vuelta
		11	mm/diente
		33	pulgadas/diente

Nota

Si `G95` no está activo, las variables $P_FZ y $AC_FZ devuelven siempre el valor cero.

2.8 Ajustes de geometría

2.8.1 Decalaje de origen ajustable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153)

Mediante los comandos G54 a G57 y G505 a G599 se activan los decalajes de origen ajustables a través de la interfaz de usuario para el decalaje del sistema de coordenadas de pieza respecto al sistema de coordenadas básico.

Sintaxis

Activar:

```
G54
...
G57
G505
...
G599
```

Desactivar o suprimir:

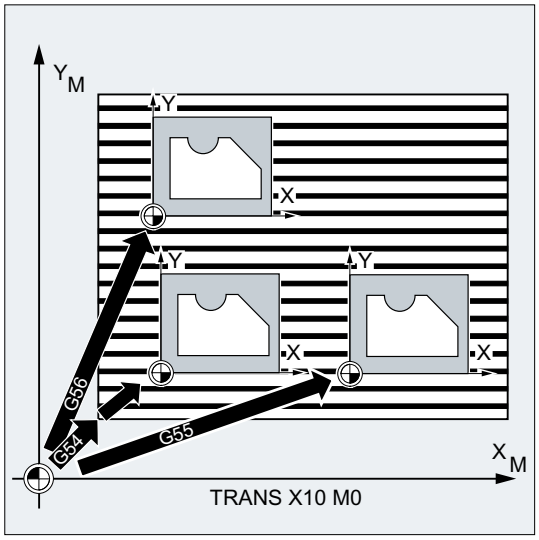
```
G500 / G53 / G153 / SUPA
```

Descripción

G54 ... G57:	Llamada del 1.º al 4.º decalaje de origen (DO) ajustable	
G505 ... G599:	Llamada del 5.º al 99.º DO ajustable	
G500:	Desactivación del DO ajustable actual	
	G500=Frame cero: (Ajuste por defecto; no contiene ningún decalaje, giro, simetría especular o escala)	Desactivación del DO ajustable hasta la siguiente llamada, activación del frame básico total ($P_ACTBFRAME).
	G500 distinto de 0:	Activación del primer decalaje de origen ajustable ($P_UIFR[0]) y activación del frame básico total ($P_ACTBFRAME), o bien activación de un frame básico eventualmente modificado.
G53:	G53 suprime en cada secuencia el DO ajustable y el DO programable.	
G153:	G153 actúa como G53 y suprime además el frame básico total.	
SUPA:	SUPA actúa como G153 y suprime además: • Decalajes con volante (DRF) • Movimientos superpuestos • DO externo • Decalaje de PRESET	

Ejemplo

Deben mecanizarse sucesivamente tres piezas dispuestas en un palet según los decalajes de origen `G54` a `G56`. El proceso de mecanizado está escrito en el subprograma L47.



Código de programa	**Comentarios**
`N10 G0 G90 X10 Y10 F500 T1`	`; Aproximación`
`N20 G54 S1000 M3`	`; Activación del primer DO, cabezal a derechas`
`N30 L47`	`; Ejecución del programa como subprograma`
`N40 G55 G0 Z200`	`; Activación del segundo DO, Z salvando obstáculo`
`N50 L47`	`; Ejecución del programa como subprograma`
`N60 G56`	`; Activación del tercer DO`
`N70 L47`	`; Ejecución del programa como subprograma`
`N80 G53 X200 Y300 M30`	`; Suprimir decalaje de origen, fin del programa`

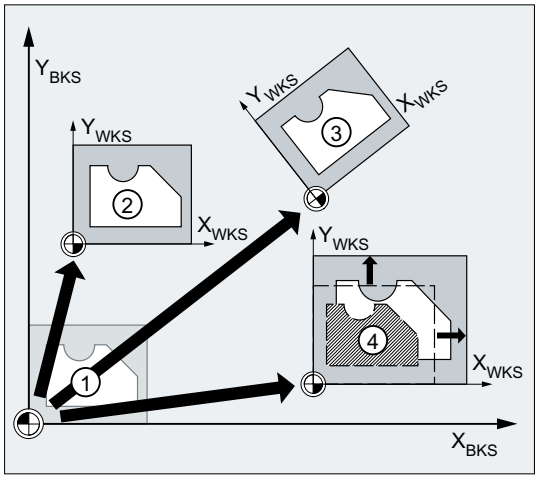
2.8.2 Decalaje de origen ajustable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153): Información adicional

Información adicional

Decalaje de origen ajustable

Un decalaje de origen ajustable es en principio un frame (Página 314) ajustable. Por ello, en caso de decalaje de origen ajustable también están disponibles los siguientes componentes o valores de frame:

- Decalaje
- Rotación
- Escalado
- Escala



① Posición de partida en el sistema de coordenadas básico
② Decalaje
③ Decalaje + rotación
④ Decalaje + escalado

Los valores de frame para los decalajes de origen ajustables se introducen a través de la interfaz de usuario:

SINUMERIK Operate: campo de manejo "Parámetros" > "Decalajes de origen" > "Detalles"

Número parametrizable de frames ajustables (G505 - G599)

El número de decalajes de origen ajustables específicos de usuario (G505 - G599) puede ajustarse para cada canal mediante:

MD28080 $MC_MM_NUM_USER_FRAMES = <Número>

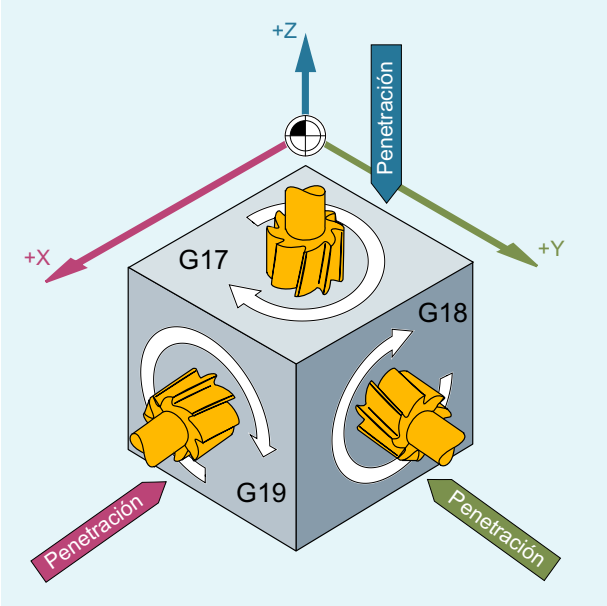
Consulte también

Decalaje de origen programable (G58, G59) (Página 323)

2.8.3 Selección del plano de trabajo (G17/G18/G19)

Mediante la indicación del plano de trabajo en el que se desea mecanizar se determinan simultáneamente las siguientes funciones:

- El plano para la corrección del radio de la herramienta.
- Dirección de penetración para la corrección longitudinal de herramienta dependiente del tipo de la misma.
- El plano para la interpolación circular.



Sintaxis

```
G17/G18/G19 ...
```

Descripción

G17:	Plano de trabajo X/Y Dirección de penetración Z selección del plano definido por los ejes geométricos 1 y 2
G18:	Plano de trabajo Z/X Dirección de penetración Y selección del plano definido por los ejes geométricos 3 y 1
G19:	Plano de trabajo Y/Z Dirección de penetración X selección del plano definido por los ejes geométricos 2 y 3

Nota

Por defecto se han ajustado el plano G17 (plano X/Y) para el fresado y G18 (plano Z/X) para el torneado.

Al utilizar la corrección de la trayectoria de la herramienta G41/G42 (ver capítulo "Correcciones del radio de herramienta (Página 257)") se debe indicar el plano de trabajo de manera que el control pueda corregir la longitud y el radio de la herramienta.

Ejemplo

El método "tradicional" en el fresado es:

1. Definir el plano de trabajo (G17 ajuste por defecto para fresado).
2. Llamar al tipo de herramienta (T) y a los valores de corrección de herramienta (D).

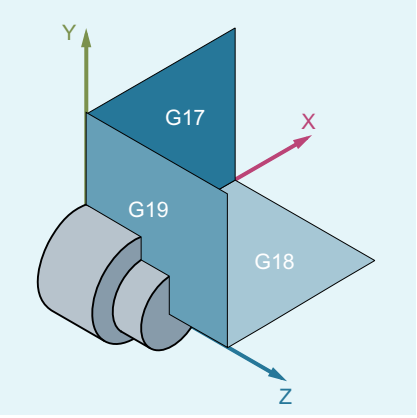
3. Activar la corrección de trayectoria (`G41`).
4. Programar los desplazamientos.

Código de programa	**Comentarios**
`N10 G17 T5 D8`	`; Llamada del plano de trabajo X/Y, llamada de herramienta. La corrección longitudinal tiene lugar en dirección Z.`
`N20 G1 G41 X10 Y30 Z-5 F500`	`; La corrección del radio tiene lugar en el plano X/Y.`
`N30 G2 X22.5 Y40 I50 J40`	`; Interpolación circular/corrección del radio de herramienta en el plano X/Y.`

Información adicional

Información general

Se recomienda definir el plano de trabajo G17 a G19 en la cabecera del programa. En el ajuste básico está preajustado G18, el plano Z/X, para el torneado:

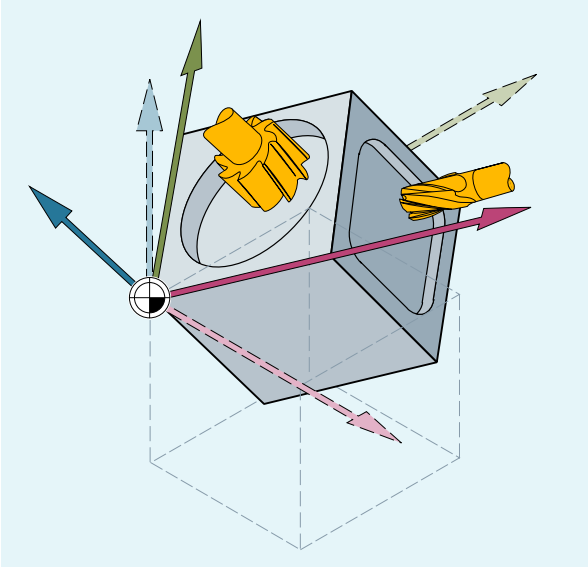


Para calcular el sentido de giro, el control necesita que se indique el plano de trabajo. Encontrará más información al respecto en el capítulo "Interpolación circular (Página 194)".

Mecanizado en planos inclinados

Mediante el giro del sistema de coordenadas con ROT (Página 325) se puede asociar dicho sistema de coordenadas a la superficie inclinada. Los planos de trabajo rotan de la misma forma que lo hacen los ejes que los definen.

Corrección de la longitud de la herramienta en planos inclinados



La corrección longitudinal de la herramienta se realiza generalmente en el plano de trabajo fijo, no rotado.

Nota

Las funcionalidades de "Corrección longitudinal de herramienta para herramientas orientables" permiten calcular los componentes de longitud de herramienta de acuerdo con los planos de trabajo a ser girados.

La selección del plano de corrección se realiza mediante CUT2D, CUT2DF. Encontrará más información al respecto y sobre la descripción de esta posibilidad de cálculo en el capítulo "Correcciones del radio de herramienta (Página 257)".

Para la definición en el espacio del sistema de coordenadas el control numérico dispone de los comandos necesarios. Encontrará más información al respecto en el capítulo "Transformaciones de coordenadas (frames) (Página 314)".

2.8.4 Acotado

La base de la mayoría de los programas CN es un plano de pieza con un acotado concreto.

Este acotado puede ser:

- acotado absoluto o incremental
- en milímetros o pulgadas
- en radio o diámetro (en el torneado)

Para poder adoptar directamente (sin conversión) los datos de un plano acotado en el programa CN, el usuario dispone de comandos de programación específicos para las diferentes posibilidades de acotado.

2.8.4.1 Acotado absoluto (G90, AC)

En el acotado absoluto, las indicaciones de posición se refieren siempre al origen del sistema de coordenadas actualmente válido, es decir, se programa la posición absoluta a la que debe desplazarse la herramienta.

Acotado absoluto modalmente activo

El acotado absoluto modalmente activo se activa con el comando `G90`. Actúa para todos los ejes que se programan en las secuencias CN posteriores.

Acotado absoluto válido secuencia a secuencia

Con el acotado incremental preajustado (`G91`) es posible ajustar por secuencias un acotado absoluto para determinados ejes con la ayuda del comando `AC`.

Nota

El acotado absoluto válido secuencia a secuencia (`AC`) también es posible para posicionados de cabezal (`SPOS`, `SPOSA`) y parámetros de interpolación (`I`, `J`, `K`).

Sintaxis

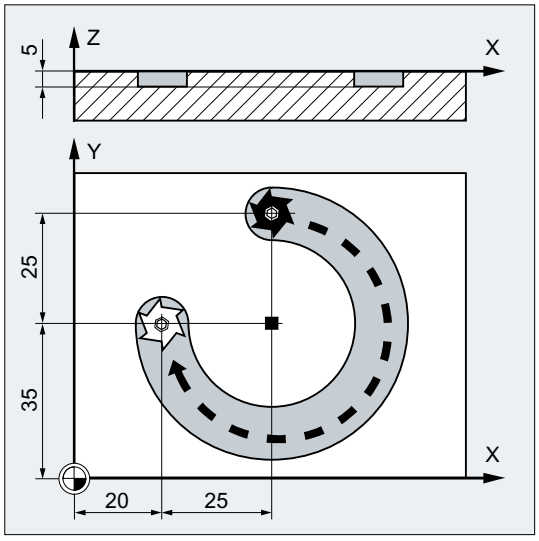
```
G90
<Eje>=AC(<Valor>)
```

Descripción

`G90`:	Comando para la activación del acotado absoluto modalmente activo
`AC`:	Comando para la activación del acotado absoluto válido secuencia a secuencia
`<Eje>`:	Identificador del eje que se va a desplazar
`<Valor>`:	Posición nominal del eje que se va a desplazar en cotas absolutas

Ejemplos

Ejemplo 1: fresado

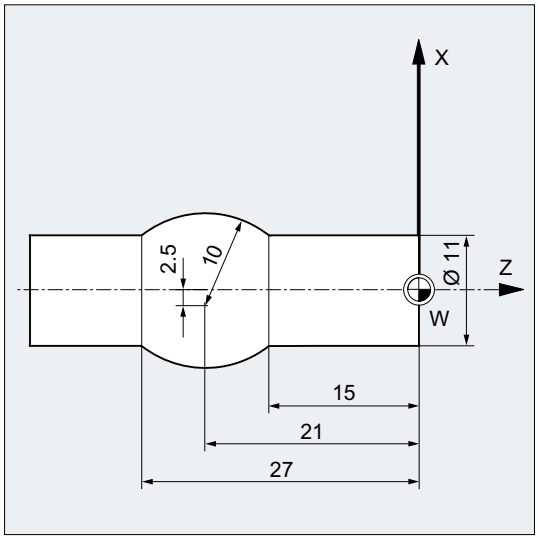


Código de programa	Comentarios
N10 G90 G0 X45 Y60 Z2 T1 S2000 M3	; Acotado absoluto, desplazamiento en rápido a la posición XYZ, selección de herramienta, cabezal activado con sentido de giro a la derecha.
N20 G1 Z-5 F500	; Interpolación lineal, posicionado de la herramienta.
N30 G2 X20 Y35 I=AC(45) J=AC(35)	; Interpolación circular en sentido horario, punto final del arco y centro del círculo en cotas absolutas.
N40 G0 Z2	; Retirada.
N50 M30	; Fin de secuencia.

Nota

Sobre la introducción de las coordenadas del centro del círculo I y J, ver capítulo "Interpolación circular".

Ejemplo 2: Torneado



Código de programa	**Comentarios**
`N5 T1 D1 S2000 M3`	`; Carga de la herramienta T1, cabezal activado con sentido de giro a la derecha.`
`N10 G0 G90 X11 Z1`	`; Acotado absoluto, desplazamiento en rápido a la posición XZ.`
`N20 G1 Z-15 F0,2`	`; Interpolación lineal, posicionado de la herramienta.`
`N30 G3 X11 Z-27 I=AC(-5) K=AC(-21)`	`; Interpolación circular en sentido antihorario, punto final del arco y centro del círculo en cotas absolutas.`
`N40 G1 Z-40`	`; Retirada.`
`N50 M30`	`; Fin de secuencia.`

Nota

Sobre la introducción de las coordenadas del centro del círculo I y J, ver capítulo "Interpolación circular".

Consulte también

Acotado absoluto e incremental en el torneado y el fresado (G90/G91) (Página 159)

2.8.4.2 Acotado incremental (G91, IC)

En el acotado incremental, una indicación de posición se refiere al último punto alcanzado, es decir, la programación en cotas incrementales describe cuánto debe desplazarse la herramienta.

Acotado incremental modalmente activo

El acotado incremental modalmente activo se activa con el comando `G91`. Actúa para todos los ejes que se programan en las secuencias CN posteriores.

Acotado incremental válido secuencia a secuencia

Con el acotado absoluto preajustado (G90) es posible ajustar por secuencias un acotado incremental para determinados ejes con la ayuda del comando IC.

Nota

El acotado incremental válido secuencia a secuencia (IC) también es posible para posicionados de cabezal (SPOS, SPOSA) y parámetros de interpolación (I, J, K).

Sintaxis

```
G91
<Eje>=IC(<Valor>)
```

Descripción

G91:	Comando para la activación del acotado incremental modalmente activo
IC:	Comando para la activación del acotado incremental válido secuencia a secuencia
<Eje>:	Identificador del eje que se va a desplazar
<Valor>:	Posición nominal del eje que se va a desplazar en cotas incrementales

Ampliación G91

Para determinadas aplicaciones como, p. ej., la aproximación con contacto, es necesario recorrer sólo la trayectoria programada en cotas incrementales. Para el desplazamiento no se tienen en cuenta ni la corrección longitudinal de herramienta ni el decalaje de origen activos.

Este comportamiento puede ajustarse por separado para la corrección longitudinal de herramienta y el decalaje de origen activos por medio de los siguientes datos de operador:

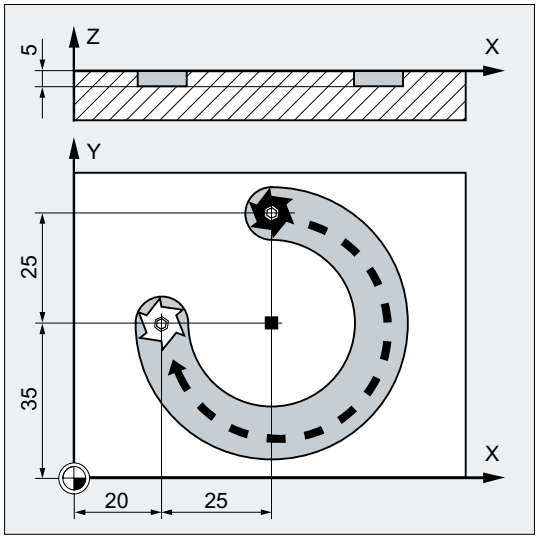
SD42440 $SC_FRAME_OFFSET_INCR_PROG (decalajes de origen en frames)

SD42442 $SC_TOOL_OFFSET_INCR_PROG (correcciones longitudinales de herramienta)

Valor	Descripción
0	En la programación incremental (acotado incremental) de un eje **no** se retiran el decalaje de origen ni la corrección longitudinal de herramienta activos.
1	En la programación incremental (acotado incremental) de un eje se retiran el decalaje de origen o la corrección longitudinal de herramienta activos.

Ejemplos

Ejemplo 1: fresado

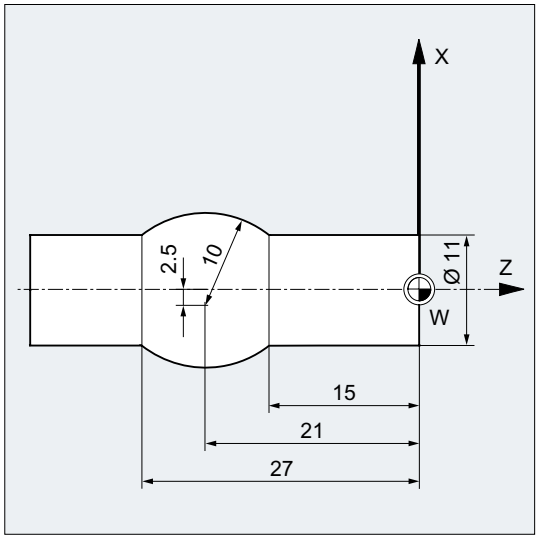


Código de programa	Comentarios
N10 G90 G0 X45 Y60 Z2 T1 S2000 M3	; Acotado absoluto, desplazamiento en rápido a la posición XYZ, selección de herramienta, cabezal activado con sentido de giro a la derecha.
N20 G1 Z-5 F500	; Interpolación lineal, posicionado de la herramienta.
N30 G2 X20 Y35 I0 J-25	; Interpolación circular en sentido horario, punto final del arco en cotas absolutas, centro del círculo en cotas incrementales.
N40 G0 Z2	; Retirada.
N50 M30	; Fin de secuencia.

Nota

Sobre la introducción de las coordenadas del centro del círculo I y J, ver capítulo "Interpolación circular".

Ejemplo 2: torneado



Código de programa	Comentarios
N5 T1 D1 S2000 M3	; Carga de la herramienta T1, cabezal activado con sentido de giro a la derecha.
N10 G0 G90 X11 Z1	; Acotado absoluto, desplazamiento en rápido a la posición XZ.
N20 G1 Z-15 F0,2	; Interpolación lineal, posicionado de la herramienta.
N30 G3 X11 Z-27 I-8 K-6	; Interpolación circular en sentido antihorario, punto final del arco en cotas absolutas, centro del círculo en cotas incrementales.
N40 G1 Z-40	; Retirada.
N50 M30	; Fin de secuencia.

Nota

Sobre la introducción de las coordenadas del centro del círculo I y J, ver capítulo "Interpolación circular".

Ejemplo 3: acotado incremental sin retirar el decalaje de origen activo

Ajustes:

- G54 contiene un decalaje en X de 25
- SD42440 $SC_FRAME_OFFSET_INCR_PROG = 0

Código de programa	Comentarios
N10 G90 G0 G54 X100	
N20 G1 G91 X10	; Acotado incremental activo, desplazamiento en X de 10 mm (en el desplazamiento no se tiene en cuenta el decalaje de origen).

Código de programa	Comentarios
N30 G90 X50	; Acotado absoluto activo, desplazamiento a la posición X75 (en el desplazamiento se tiene en cuenta el decalaje de origen).

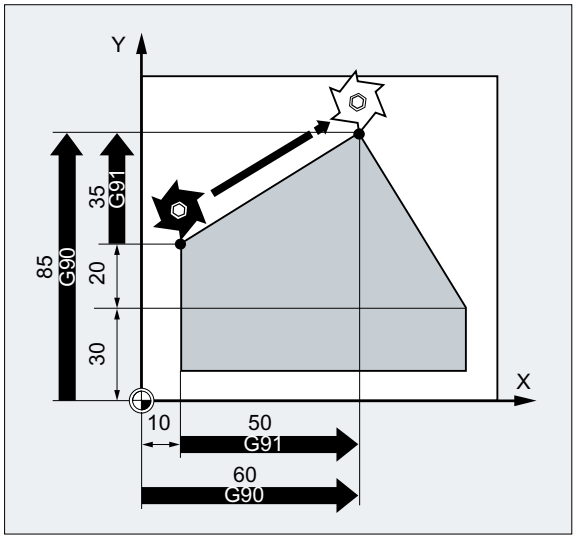
Consulte también

Acotado absoluto e incremental en el torneado y el fresado (G90/G91) (Página 159)

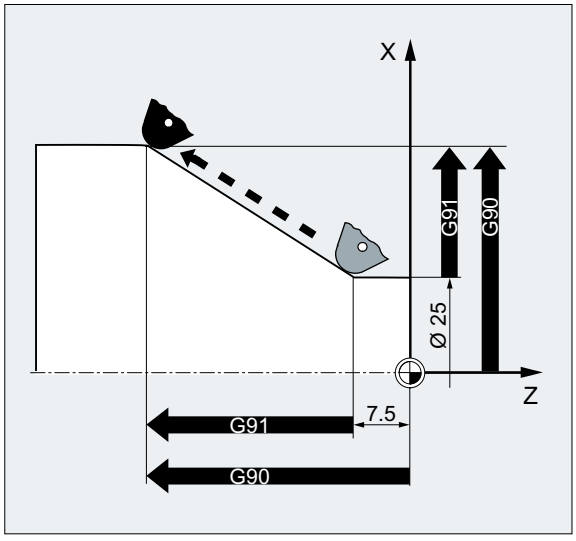
2.8.4.3 Acotado absoluto e incremental en el torneado y el fresado (G90/G91)

Las dos figuras siguientes ilustran la programación con acotado absoluto (G90) o acotado incremental (G91) tomando como ejemplo las tecnologías de torneado y fresado.

Fresado:



Torneado:



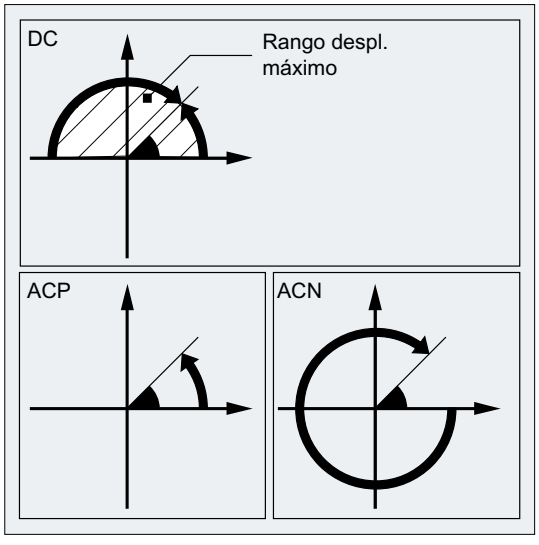
Nota

En tornos convencionales es normal utilizar secuencias de desplazamiento en cotas incrementales programadas en radios para el eje de refrentado, mientras que el acotado de referencia se introduce en diámetros. La conmutación para `G90` se realiza mediante los comandos `DIAMON`, `DIAMOF` y `DIAM90`.

2.8.4.4 Acotado absoluto para ejes giratorios (DC, ACP, ACN)

Para el posicionado de ejes giratorios en cotas absolutas se dispone de los comandos DC, ACP y ACN, activos por secuencias e independientes de G90/G91.

DC, ACP y ACN se diferencian en la estrategia de aproximación subyacente:



Sintaxis

```
<Eje giratorio>=DC(<Valor>)
<Eje giratorio>=ACP(<Valor>)
<Eje giratorio>=ACN(<Valor>)
```

Descripción

<Eje giratorio>:	Identificador del eje giratorio que debe desplazarse (p. ej., A, B o C)	
DC:	Comando de aproximación **directa** a la posición El eje giratorio se desplaza a la posición programada por el camino más corto y directo. El eje giratorio se desplaza como máximo 180°.	
ACP:	Comando para la aproximación a la posición en dirección **positiva** El eje giratorio se desplaza a la posición programada en sentido de giro positivo (sentido antihorario).	
ACN:	Comando para la aproximación a la posición en dirección **negativa** El eje giratorio se desplaza a la posición programada en sentido de giro negativo (sentido horario).	
<Valor>:	Posición de eje giratorio que debe alcanzarse en cotas absolutas	
	Rango de valores:	0-360 grados

Nota

El sentido positivo de giro (horario o antihorario) se define mediante datos de máquina.

Nota

Para el posicionado con definición del sentido (ACP, ACN) debe haberse ajustado en el dato de máquina la zona de desplazamiento entre 0° y 360° (módulo). Para poder desplazar más de 360° un eje giratorio (módulo) en una sola secuencia, debe programarse G91 o bien IC.

Nota

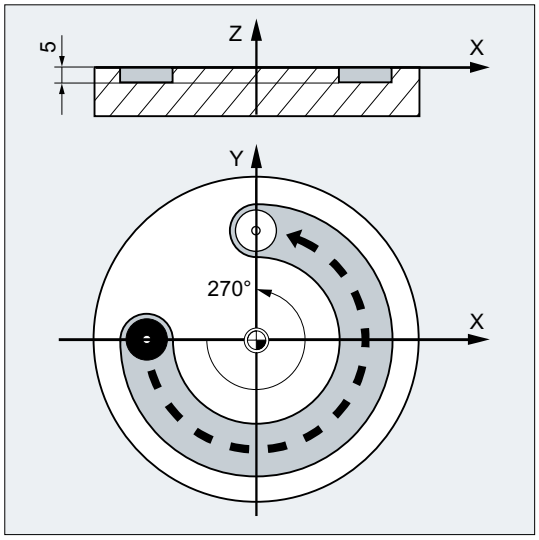
Los comandos DC, ACP y ACN pueden utilizarse también para el posicionado de cabezales (SPOS, SPOSA) desde el estado de parada.

Ejemplo: SPOS=DC(45)

Ejemplo

fresado en una mesa giratoria

La herramienta está parada y la mesa gira hasta los 270º **en sentido horario**. Se pretende fresar una ranura circular.



Código de programa	**Comentarios**
N10 SPOS=0	; Cabezal en regulación de posición.
N20 G90 G0 X-20 Y0 Z2 T1	; Acotado absoluto, posicionado de herramienta T1 en rápido.
N30 G1 Z-5 F500	; Descenso de la herramienta al avanzar.
N40 C=ACP(270)	; La mesa gira hasta los 270 grados en sentido horario (positivo), la herramienta fresa una ranura circular.
N50 G0 Z2 M30	; Levantamiento, fin de programa.

2.8.4.5 Sistema de medida métrico/imperial (G70/G71, G700/G710)

Con los comandos del Grupo G 13 (sistema de medida imperial/métrico) dentro del programa de pieza se puede conmutar entre el sistema de medida métrico y el sistema de medida en pulgadas.

Activación

Para que estén disponibles los comandos G700 y G710, debe estar activa la funcionalidad ampliada del sistema de medida (MD10260 $MN_CONVERT_SCALING_SYSTEM = 1).

Sintaxis

```
G70
G71
G700
G710
```

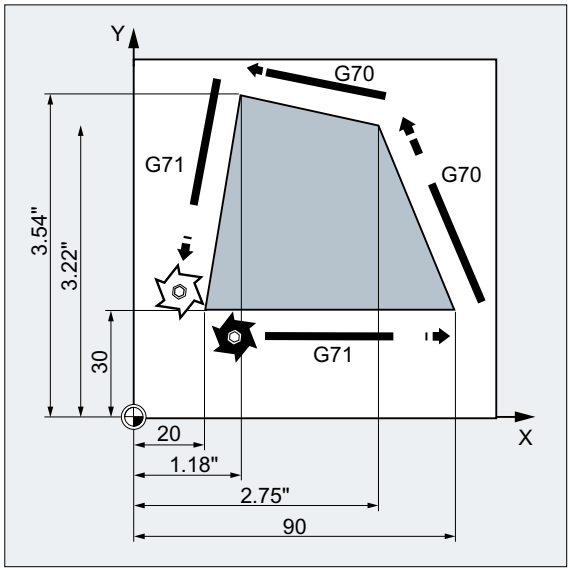
Significado

G70:	Activación del sistema de medida en pulgadas Los datos **geométricos con indicación de longitud** se leen y se escriben en el **sistema de medida en pulgadas**. Los datos **tecnológicos con indicación de longitud** (p. ej., avances, correcciones de herramienta, decalajes de origen ajustables, datos de máquina y variables de sistema) se leen y se escriben en el **sistema básico parametrizado**.	
	Grupo G:	13
	Posición básica:	Ajustable a través de MD20150 $MC_GCODE_RESET_VALUES
	Eficacia:	Modal
G71:	Activación del sistema de medida métrico Los datos **geométricos con indicación de longitud** se leen y se escriben en el **sistema de medida métrico**. Los datos **tecnológicos con indicación de longitud** (p. ej., avances, correcciones de herramienta, decalajes de origen ajustables, datos de máquina y variables de sistema) se leen y se escriben en el **sistema básico parametrizado**.	
	Grupo G:	13
	Posición básica:	Ajustable a través de MD20150 $MC_GCODE_RESET_VALUES
	Eficacia:	Modal
G700:	Activación del sistema de medida en pulgadas Todos los datos geométricos y tecnológicos con indicación de longitud se leen y se escriben en el sistema de medida en pulgadas.	
	Grupo G:	13
	Posición básica:	Ajustable a través de MD20150 $MC_GCODE_RESET_VALUES
	Eficacia:	Modal
G710:	Activación del sistema de medida métrico Todos los datos geométricos y tecnológicos con indicación de longitud se leen y se escriben en el sistema de medida métrico.	
	Grupo G:	13
	Posición básica:	Ajustable a través de MD20150 $MC_GCODE_RESET_VALUES
	Eficacia:	Modal

ATENCIÓN
Datos específicos de eje de los ejes giratorios
Los datos específicos de eje de los ejes giratorios se leen y se escriben siempre en el sistema básico parametrizado.

Ejemplo

El sistema básico es métrico (MD10240 $MN_SCALING_SYSTEM_IS_METRIC = 1). No obstante, la denominación de la pieza contiene dimensiones en pulgadas. Por ello, dentro del programa de pieza se conmuta al sistema de medida en pulgadas. Tras las indicaciones de medidas en pulgadas se cambia de nuevo al sistema de medida métrico.



Código de programa	Comentarios
N10 G0 G90 X20 Y30 Z2 S2000 M3 T1	; X = 20 mm, Y = 30 mm, Z = 2 mm, F = desplazamiento en rápido 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	; Sistema de medida programado: pulgadas
	; X=2,75 pulgadas, Y=3,22 pulgadas, F=500 mm/min
N50 X1.18 Y3.54	; X = 1,18 pulgadas, Y = 3,54 pulgadas, F = 500 mm/min
N60 **G71** X20 Y30	; Sistema de medida programado: métrico
	; X=20 mm, Y=30 mm, F=500 mm/min
N70 G0 Z2	; Z = 2 mm, F = desplazamiento en rápido mm/min
N80 M30	; Fin del programa

Información adicional

Lectura y escritura de datos con G70/G71 y G700/G710

Área de datos	G70/G71		G700/G710	
	Leer	Escribir	Leer	Escribir
Visualización, decimales (WKS)	P	P	P	P
Visualización, decimales (MKS)	G	G	G	G
Avances	G	G	P	P

Área de datos	G70/G71		G700/G710	
	Leer	Escribir	Leer	Escribir
Datos de posición X, Y, Z	P	P	P	P
Parámetros de interpolación I, J, K	P	P	P	P
Radio del círculo (CR)	P	P	P	P
Radio polar (RP)	P	P	P	P
Paso de rosca	P	P	P	P
FRAME programable	P	P	P	P
FRAMES ajustables	G	G	P	P
Frames básicos	G	G	P	P
Decalajes de origen externos	G	G	P	P
Decalaje de Preset axial	G	G	P	P
Límites del campo de trabajo (G25/G26)	G	G	P	P
Zonas protegidas	P	P	P	P
Correcciones de herramientas	G	G	P	P
Datos de máquina relativos a la longitud	G	G	P	P
Datos de operador relativos a la longitud	G	G	P	P
Variables de sistema relativas a la longitud	G	G	P	P
GUD	G	G	G	G
LUD	G	G	G	G
PUD	G	G	G	G
Parámetros R	G	G	G	G
Ciclos de Siemens	P	P	P	P
Valoración de incremento JOG/volante	G	G	G	G
P: la lectura/escritura se realiza en el sistema de medida programado G: la lectura/escritura se realiza en el sistema básico parametrizado				

Acciones síncronas

Nota

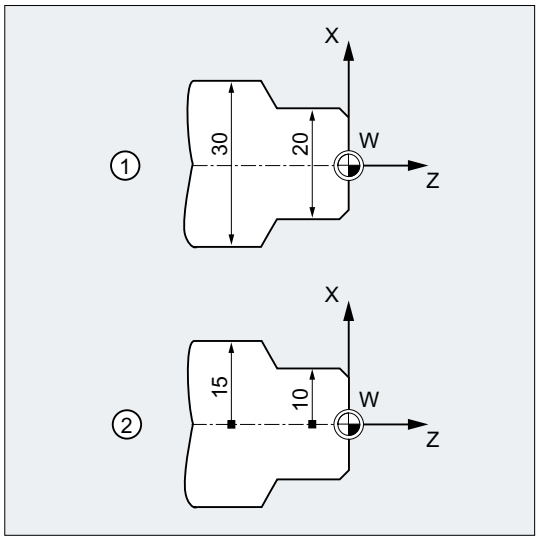
Lectura de datos de posición en acciones síncronas

Si no se programa explícitamente el sistema de medida en la acción síncrona (parte de condición y/o de acción, o bien función tecnológica), los **datos de posición relativos a la longitud** en la acción síncrona se leen siempre en el **sistema básico** parametrizado.

Más información en: Manual de funciones Acciones síncronas

2.8.4.6 Programación por diámetro/radio específica del canal (DIAMON, DIAM90, DIAMOF, DIAMCYCOF)

En el torneado, las cotas **del eje de refrentado** pueden indicarse en el diámetro (①) o en el radio (②):



Para poder adoptar directamente, sin conversión, el acotado del plano técnico en el programa CN, se activa la programación por diámetro o radio específica del canal por medio de los comandos modalmente activos `DIAMON`, `DIAM90`, `DIAMOF` y `DIAMCYCOF`.

Nota

La programación por diámetro/radio específica del canal se refiere al eje geométrico definido como eje de refrentado mediante DM 20100 $MC_DIAMETER_AX_DEF (→ ver las indicaciones del fabricante de la máquina).

Con DM 20100 sólo puede haber definido un eje de refrentado por canal.

Sintaxis

```
DIAMON
DIAM90
DIAMOF
```

Descripción

`DIAMON`:	Comando para la activación de la programación por diámetro **independiente** específica del canal El efecto de `DIAMON` es independiente del modo de acotado programado (acotado absoluto `G90` o acotado incremental `G91`):	
	• Con G90:	Acotado en el diámetro
	• Con G91:	Acotado en el diámetro

DIAM90:	Comando para activar la programación por diámetro **dependiente** específica del canal El efecto de DIAM90 depende del modo de acotado programado:	
	• Con G90:	Acotado en el diámetro
	• Con G91:	Acotado en el radio
DIAMOF:	Comando para la desactivación de la programación por diámetro específica del canal Al desactivar la programación por diámetro, se activa la programación por radio específica del canal. El efecto de DIAMOF es independiente del modo de acotado programado:	
	• Con G90:	Acotado en el radio
	• Con G91:	Acotado en el radio
DIAMCYCOF:	Comando para la desactivación de la programación por diámetro específica del canal durante la ejecución de ciclos Por tanto, en el ciclo, los cálculos pueden realizarse siempre en el radio. Para la indicación de posición y la visualización de secuencia base permanece activo el último comando G activo de ese grupo.	

Nota

Con DIAMON o DIAM90 los valores reales del eje de refrentado se indican siempre como diámetro. Esto también es válido para la lectura de los valores reales en el sistema de coordenadas de pieza con MEAS, MEAW, $P_EP[x] y $AA_IW[x].

Ejemplo

Código de programa	**Comentarios**
N10 G0 X0 Z0	; Desplazamiento al punto inicial.
N20 DIAMOF	; Programación por diámetro desactivada.
N30 G1 X30 S2000 M03 F0.7	; Eje X = eje de refrentado, programación por radio activa, desplazamiento a posición de radio X30.
N40 DIAMON	; La programación por diámetro está activa para el eje de refrentado.
N50 G1 X70 Z-20	; Desplazamiento a la posición de diámetro X70 y Z-20.
N60 Z-30	
N70 DIAM90	; Programación por diámetro para acotado absoluto y programación por radio para acotado incremental.
N80 G91 X10 Z-20	; Acotado incremental activo.
N90 G90 X10	; Acotado absoluto activo.
N100 M30	; Fin del programa.

información adicional

Valores de diámetro (DIAMON/DIAM90)

Los valores asociados a los siguientes datos son tenidos en cuenta como diámetros:

- Visualización del valor real del eje de refrentado en el sistema de coordenadas de la pieza
- Modo JOG: Incrementos para medidas incrementales y desplazamiento manual mediante volante
- Programación de posiciones finales:
  Parámetros de interpolación `I`, `J`, `K` con `G2`/`G3`, si estos han sido programados de forma absoluta con `AC`.
  En la programación incremental (`IC`) de `I`, `J`, `K` se calcula siempre el radio.
- Lectura de valores reales en el sistema de coordenadas de pieza para:
  `MEAS`, `MEAW`, `$P_EP[X]`, `$AA_IW[X]`

2.8.4.7 Programación por diámetro/radio específica del eje (DIAMONA, DIAM90A, DIAMOFA, DIACYCOFA, DIAMCHANA, DIAMCHAN, DAC, DIC, RAC, RIC)

Además de la programación por diámetro específica de canal, la programación por diámetro específica de eje permite el acotado modalmente activo o válido secuencia a secuencia y la visualización en el diámetro para uno o varios ejes.

Nota

La programación por diámetro específica del eje sólo es posible en ejes admitidos como ejes de refrentado adicionales para la programación por diámetro específica del eje mediante DM 30460 $MA_BASE_FUNCTION_MASK (→ ver las indicaciones del fabricante de la máquina).

Sintaxis

Programación por diámetro específica del eje modalmente activa para varios ejes de refrentado en el canal:

```
DIAMONA[<Eje>]
DIAM90A[<Eje>]
DIAMOFA[<Eje>]
DIACYCOFA[<Eje>]
```

Aplicación de la programación por diámetro/radio específica del canal:

```
DIAMCHANA[<Eje>]
DIAMCHAN
```

Programación por diámetro/radio específica del eje válida secuencia a secuencia:

```
<Eje>=DAC(<Valor>)
<Eje>=DIC(<Valor>)
<Eje>=RAC(<Valor>)
<Eje>=RIC(<Valor>)
```

Descripción

Programación por diámetro específica del eje modalmente activa		
DIAMONA:	Comando para la activación de la programación por diámetro **independiente** específica del eje El efecto de DIAMONA es independiente del modo de acotado programado (G90/G91 o AC/IC):	
	• con G90, AC:	Acotado en el diámetro
	• con G91, IC:	Acotado en el diámetro
DIAM90A:	Comando para la activación de la programación por diámetro **dependiente** específica del eje El efecto de DIAM90A depende del modo de acotado programado:	
	• con G90, AC:	Acotado en el diámetro
	• con G91, IC:	Acotado en el radio
DIAMOFA:	Comando para la desactivación de la programación por diámetro específica del eje Al desactivar la programación por diámetro, se activa la programación por radio específica del eje. El efecto de DIAMOFA es independiente del modo de acotado programado:	
	• con G90, AC:	Acotado en el radio
	• con G91, IC:	Acotado en el radio
DIACYCOFA:	Comando para la desactivación de la programación por diámetro específica del eje durante la ejecución de ciclos Por tanto, en el ciclo, los cálculos pueden realizarse siempre en el radio. Para la indicación de posición y la visualización de secuencia base permanece activo el último comando G activo de ese grupo.	
<Eje>:	Identificador del eje que debe activarse para la programación por diámetro específica del eje Los identificadores de eje admisibles son: • Nombre del eje geométrico/de canal o bien • Nombre del eje de máquina	
	Rango de valores:	El eje indicado tiene que ser un eje conocido en el canal. Otras condiciones: • El eje debe estar admitido mediante DM30460 $MA_BASE_FUNCTION_MASK para la programación por diámetro específica del eje. • No se admiten ejes giratorios como ejes de refrentado.
Aplicación de la programación por diámetro/radio específica del canal		
DIAMCHANA:	Con el comando DIAMCHANA[<Eje>], el **eje especificado** toma el estado de canal de la programación por diámetro/radio y, en adelante, queda sujeto a la programación por diámetro/radio específica del canal.	
DIAMCHAN:	Con el comando DIAMCHAN, **todos** los ejes admitidos para la programación por diámetro específica del eje toman el estado de canal de la programación de diámetro/radio y, en adelante, quedan sujetos a la programación por diámetro/radio específica del canal.	

Programación por diámetro/radio específica del eje válida secuencia a secuencia La programación por diámetro/radio específica del eje válida secuencia a secuencia define el tipo de acotado como valor de diámetro o de radio en el programa de pieza y en acciones síncronas. No se modifica el estado modal de la programación por diámetro/radio.	
DAC:	Con el comando DAC el siguiente acotado es válido secuencia a secuencia para el eje indicado: Diámetro en cotas absolutas
DIC:	Con el comando DIC el siguiente acotado es válido secuencia a secuencia para el eje indicado: Diámetro en cotas incrementales
RAC:	Con el comando RAC el siguiente acotado es válido secuencia a secuencia para el eje indicado: Radio en cotas absolutas
RIC:	Con el comando RIC el siguiente acotado es válido secuencia a secuencia para el eje indicado: Radio en cotas incrementales

Nota

Con DIAMON[<Eje>] o DIAM90[<Eje>] los valores reales del eje de refrentado se indican siempre como diámetro. Esto también es válido para la lectura de los valores reales en el sistema de coordenadas de pieza con MEAS, MEAW, $P_EP[x] y $AA_IW[x].

Nota

En caso de cambio de un eje de refrentado adicional debido a una solicitud GET, se adopta con RELEASE[<Eje>] el estado de la programación por diámetro/radio en el otro canal.

Ejemplos

Ejemplo 1: programación por diámetro/radio específica del eje modalmente activa

X es el eje de refrentado en el canal, para Y se admite una programación por diámetro específica del eje.

Código de programa	**Comentarios**
N10 G0 X0 Z0 DIAMON	; Programación por diámetro específica del canal activa para X.
N15 DIAMOF	; Programación por diámetro específica del canal desactivada.
N20 DIAMONA[Y]	; Programación por diámetro específica del eje modalmente activa para Y.
N25 X200 Y100	; Programación por radio activa para X.
N30 DIAMCHANA[Y]	; Y toma el estado de la programación por diámetro/radio específica del canal y queda sujeto a ésta
N35 X50 Y100	; Programación por radio activa para X e Y.
N40 DIAMON	; Programación por diámetro específica del canal activada.
N45 X50 Y100	; Programación por diámetro activa para X e Y.

Ejemplo 2: programación por diámetro/radio específica del eje válida secuencia a secuencia

X es el eje de refrentado en el canal, para Y se admite una programación por diámetro específica del eje.

Código de programa	Comentarios
N10 DIAMON	; Programación por diámetro específica del canal activada.
N15 G0 G90 X20 Y40 DIAMONA[Y]	; Programación por diámetro específica del eje modalmente activa para Y.
N20 G01 X=RIC(5)	; Acotado activo para X en esta secuencia: radio en cotas incrementales.
N25 X=RAC(80)	; Acotado activo para X en esta secuencia: radio en cotas absolutas.
N30 WHEN $SAA_IM[Y]> 50 DO POS[X]=RIC(1)	; X es el eje de comando. Acotado activo para X en esta secuencia: radio en cotas incrementales.
N40 WHEN $SAA_IM[Y]> 60 DO POS[X]=DAC(10)	; X es el eje de comando. Acotado activo para X en esta secuencia: radio en cotas absolutas.
N50 G4 F3	

Información adicional

Valores de diámetro (DIAMONA/DIAM90A)

Los valores asociados a los siguientes datos son tenidos en cuenta como diámetros:

- Visualización del valor real del eje de refrentado en el sistema de coordenadas de la pieza
- Modo JOG: Incrementos para medidas incrementales y desplazamiento manual mediante volante
- Programación de posiciones finales:
  Parámetros de interpolación `I`, `J`, `K` con `G2`/`G3`, si estos han sido programados de forma absoluta con `AC`.
  En la programación incremental `IC` de `I`, `J`, `K` se calcula siempre el radio.
- Lectura de valores reales en el sistema de coordenadas de pieza para:
  `MEAS`, `MEAW`, `$P_EP[X]`, `$AA_IW[X]`

Programación por diámetro específica del eje válida secuencia a secuencia (DAC, DIC, RAC, RIC)

Las instrucciones DAC, DIC, RAC, RIC se admiten para todos los comandos para los cuales se tiene en cuenta la programación por diámetro específica del canal:

- Posición del eje: X..., POS, POSA
- Vaivén: OSP1, OSP2, OSS, OSE, POSP
- Parámetros de interpolación: I, J, K
- Sucesión de elementos de contorno: Recta con especificación de ángulo
- Retirada rápida: POLF[AX]
- Desplazamiento en la dirección de herramienta: MOVT
- Aproximación y retirada suaves:
  G140 a G143, G147, G148, G247, G248, G347, G348, G340, G341

2.8.5 Posición de la pieza en el torneado

Identificaciones de eje

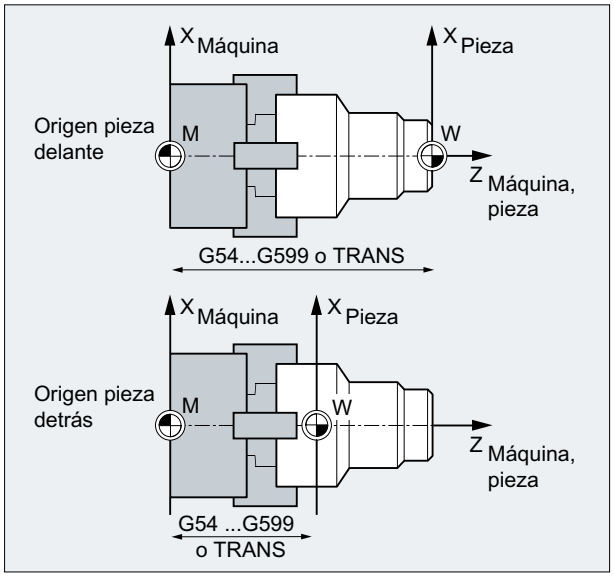
Generalmente los dos ejes geométricos perpendiculares de un torno se definen de la siguiente manera:

Eje longitudinal	= eje Z (abscisas)
Eje de refrentado	= eje X (ordenadas)

Origen de la pieza

Mientras que el origen de la máquina está predefinido, la posición del origen de la pieza en el eje longitudinal puede elegirse libremente. Por lo general, el origen de la pieza se encuentra en la parte delantera o trasera de ésta.

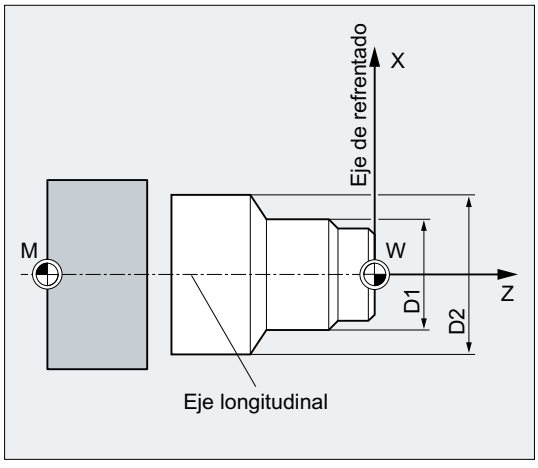
Tanto el origen de la máquina como el de la pieza están en sobre el eje de giro. De este modo, el decalaje ajustable en el eje X es cero.



M	Origen de la máquina
W	Origen de la pieza
`Z`	Eje longitudinal
`X`	Eje de refrentado
`G54` a `G599` o `TRANS`	Llamada para la posición del origen de pieza

Eje de refrentado

Por lo general, el acotado para el eje de refrentado se define como indicaciones de diámetro (cotas dobles en comparación con los otros ejes):



En los datos de máquina debe definirse qué eje geométrico sirve como eje de refrentado (→ fabricante de la máquina).

2.9 Instrucciones de desplazamiento

2.9.1 Información general sobre los comandos de desplazamiento

Elementos de contorno

El contorno de pieza programado puede estar compuesto por los siguientes elementos:

- Líneas rectas
- Arcos de circunferencia
- Hélices (mediante superposición de líneas rectas y arcos de circunferencia)

Comandos de desplazamiento

Para crear estos elementos de contorno existen diferentes comandos de desplazamiento:

- Desplazamiento en rápido (`G0`)
- Interpolación lineal (`G1`)
- Interpolación circular en sentido horario (`G2`)
- Interpolación circular en sentido antihorario (`G3`)

Los comandos de desplazamiento son modalmente activos.

Posiciones de destino

Una secuencia de desplazamiento contiene posiciones de destino para los ejes que deben desplazarse (ejes de contorneado, ejes síncronos, ejes de posicionado).

La programación de las posiciones de destino puede realizarse en coordenadas cartesianas o polares.

Nota

Solamente se puede programar una dirección de eje una sola vez por secuencia.

Punto inicial - punto final

Los desplazamientos siempre van desde la última posición alcanzada hasta el punto final programado. De hecho, el punto final programado será el nuevo punto inicial para el siguiente desplazamiento.

Contorno de la pieza

ATENCIÓN
Intervención de la herramienta no definida Antes de que se inicie un proceso de mecanizado, debe posicionarse la herramienta de manera que se evite causar cualquier daño a la pieza o a la herramienta.

Ejecutadas de forma sucesiva, las secuencias de desplazamiento dan como resultado el contorno de la pieza.

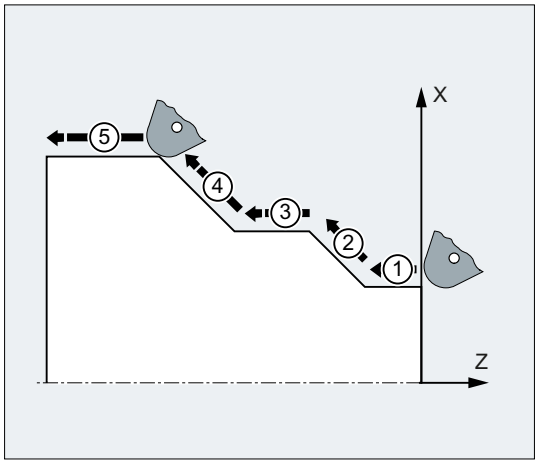


Figura 2-1 Secuencias de desplazamiento en el torneado

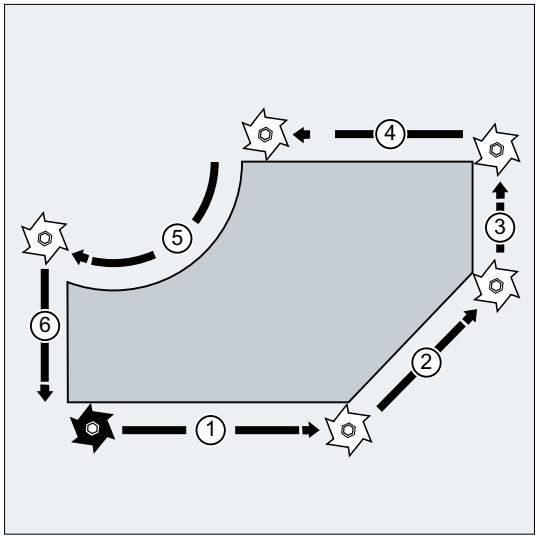


Figura 2-2 Secuencias de desplazamiento en el fresado

2.9.2 Comandos de desplazamiento con coordenadas cartesianas (G0, G1, G2, G3, X..., Y..., Z...)

La posición especificada en la secuencia CN con coordenadas cartesianas puede alcanzarse con un desplazamiento en rápido `G0`, una interpolación lineal `G1` o una interpolación circular `G2`/`G3`.

Sintaxis

```
G0 X... Y... Z...
G1 X... Y... Z...
G2 X... Y... Z... ...
G3 X... Y... Z... ...
```

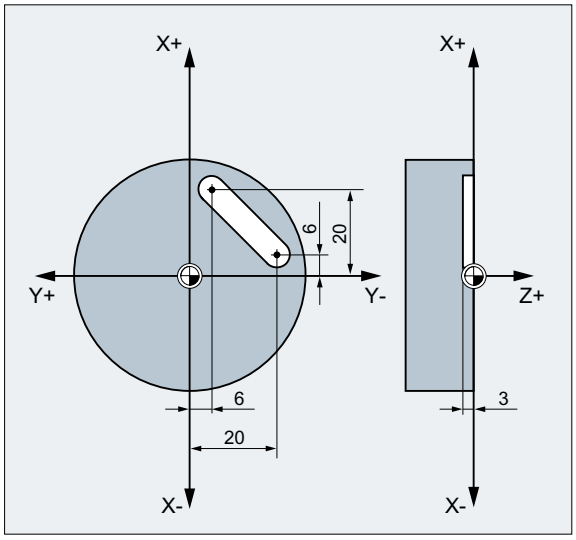
Descripción

`G0`:	Comando para la activación del desplazamiento en rápido
`G1`:	Comando para la activación de la interpolación lineal
`G2`:	Comando para la activación de la interpolación circular en sentido horario
`G3`:	Comando para la activación de la interpolación circular en sentido antihorario
`X...`:	Coordenada cartesiana de la posición de destino en dirección X
`Y...`:	Coordenada cartesiana de la posición de destino en dirección Y
`Z...`:	Coordenada cartesiana de la posición de destino en dirección Z

Nota

La interpolación circular `G2`/`G3` necesita, además de las coordenadas de la posición de destino `X...`, `Y...`, `Z...`, algunos datos más (p. ej., las coordenadas del centro del círculo; ver "Vista general (Página 194)").

Ejemplo



Código de programa	Comentarios
N10 G17 S400 M3	; Selección del plano de trabajo, cabezal a derechas
N20 G0 X40 Y-6 Z2	; Aproximación en rápido a la posición inicial indicada con coordenadas cartesianas
N30 G1 Z-3 F40	; Activación de la interpolación lineal, posicionado de la herramienta
N40 X12 Y-20	; Desplazamiento sobre una línea recta inclinada hasta la posición final indicada con coordenadas cartesianas
N50 G0 Z100 M30	; Retirada en rápido al punto de cambio de herramienta

2.9.3 Comandos de desplazamiento con coordenadas polares

2.9.3.1 Punto de referencia de las coordenadas polares (G110, G111, G112)

El punto donde se cortan los radios se denomina polo.

El polo puede indicarse con coordenadas cartesianas o polares.

Los comandos `G110` a `G112` permiten establecer de manera unívoca el punto de referencia de las coordenadas polares. Por este motivo, el acotado absoluto o incremental no tiene ninguna influencia.

Sintaxis

```
G110/G111/G112 X… Y… Z…
G110/G111/G112 AP=… RP=…
```

Descripción

`G110 ...`:	Con el comando `G110` las coordenadas polares sucesivas se refieren **a la última posición alcanzada**.		
`G111 ...`:	Con el comando `G111` las coordenadas polares sucesivas se refieren **al origen del sistema de coordenadas de pieza actual**.		
`G112 ...`:	Con el comando `G112` las coordenadas polares sucesivas se refieren **al último polo válido**.		
	Nota: los comandos `G110`...`G112` deben programarse en su propia secuencia CN.		
`X… Y… Z…`:	Indicación del polo en coordenadas cartesianas		
`AP=… RP=…`:	Indicación del polo en coordenadas polares		
	`AP=…`:	Ángulo polar Ángulo entre el radio polar y el eje horizontal del plano de trabajo (p. ej., eje X en `G17`). El sentido de giro positivo corresponde al giro en sentido antihorario.	
		Rango de valores:	± 0…360°
	`RP=…`:	Radio polar La indicación se realiza **siempre en valores absolutos positivos** en [mm] o [pulgadas].	

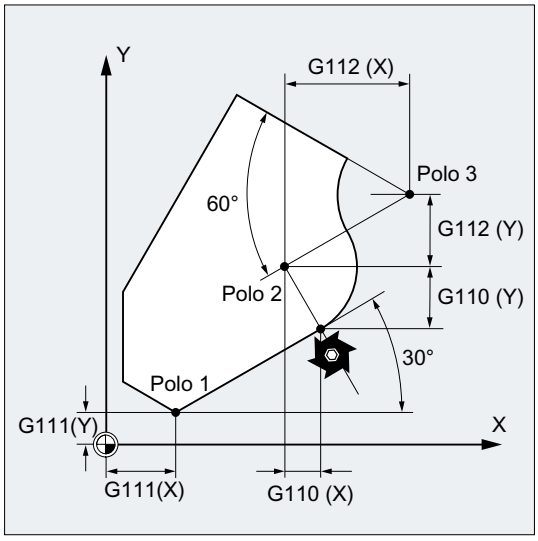
Nota

En el programa CN es posible cambiar secuencia a secuencia entre el acotado polar y el acotado cartesiano. Utilizando los identificadores de coordenadas cartesianas (X..., Y..., Z...) se vuelve directamente al sistema cartesiano. El polo definido se conserva más allá, hasta el fin del programa.

Nota

Si no se ha indicado un polo, se toma como origen el del sistema de coordenadas de pieza actual.

Ejemplo

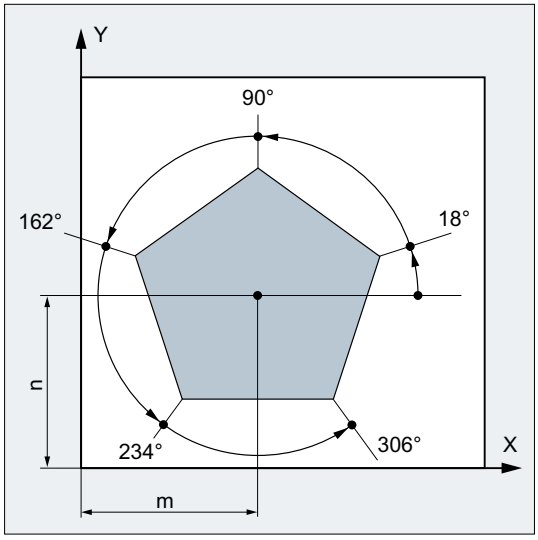


Los polos 1 a 3 se definen como sigue:

- Polo 1 con `G111 X… Y…`
- Polo 2 con `G110 X… Y…`
- Polo 3 con `G112 X… Y…`

2.9.3.2 Comandos de desplazamiento con coordenadas polares (G0, G1, G2, G3, AP, RP)

Los comandos de desplazamiento con coordenadas polares resultan útiles cuando el acotado de una pieza o de una parte de una pieza sale de un punto central y las cotas se indican con ángulos y radios (p. ej., figuras de taladros).



Sintaxis

```
G0/G1/G2/G3 AP=… RP=…
```

Descripción

G0:	Comando para la activación del desplazamiento en rápido	
G1:	Comando para la activación de la interpolación lineal	
G2:	Comando para la activación de la interpolación circular en sentido horario	
G3:	Comando para la activación de la interpolación circular en sentido antihorario	
AP:	Ángulo polar Ángulo entre el radio polar y el eje horizontal del plano de trabajo (p. ej., eje X en G17). El sentido de giro positivo corresponde al giro en sentido antihorario.	
	Rango de valores:	± 0…360°
	La especificación del ángulo puede ser tanto absoluta como incremental:	
	AP=AC(...):	Acotado absoluto
	AP=IC(...):	Acotado incremental En el acotado incremental se utiliza como referencia el último ángulo programado.
	El ángulo polar se memoriza hasta que se define un nuevo polo o bien se cambia de plano de trabajo.	
RP:	Radio polar La indicación se realiza **siempre en valores absolutos positivos** en [mm] o [pulgadas]. El radio polar queda memorizado hasta que se introduce un nuevo valor.	

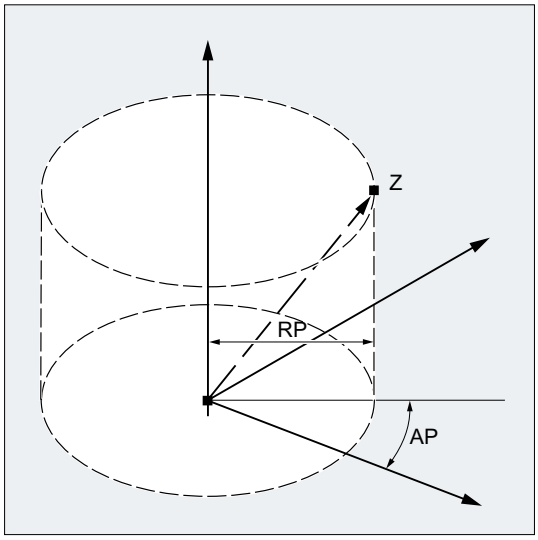
Nota

Las coordenadas polares se refieren al polo definido con G110 ... G112 y se aplican en el plano de trabajo seleccionado con G17 a G19.

Nota

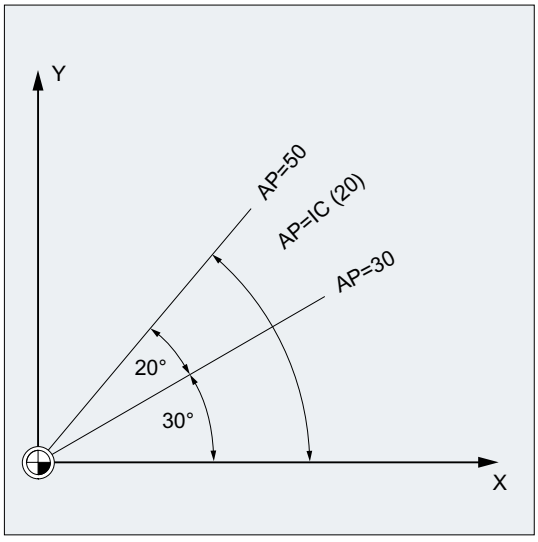
El 3.er eje geométrico que se encuentra perpendicular al plano de trabajo también puede especificarse en coordenadas cartesianas (ver figura siguiente). Esto permite programar parámetros en una tercera dimensión para definir posiciones en coordenadas cilíndricas.

Ejemplo: G17 G0 AP… RP… Z…



Condiciones

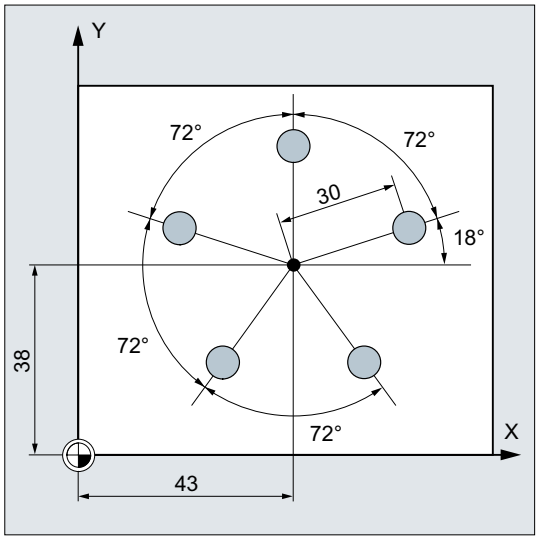
- En secuencias de CN con introducción del punto final en coordenadas polares no se pueden programar parámetros de interpolación, direcciones de ejes, etc., asociados al sistema de coordenadas cartesiano del plano de trabajo seleccionado.
- Si no se define ningún polo con `G110` ... `G112`, automáticamente se adopta como polo el origen del sistema de coordenadas de pieza actual.



- Radio polar RP = 0
  El radio polar se calcula a partir de la distancia entre el vector de posición inicial en el plano del polo y el vector polar activo. A continuación, se memoriza de forma modal el radio polar calculado.
  Esto se aplica independientemente de la definición de polo elegida (`G110` ... `G112`). Si la programación de ambos puntos es idéntica, este radio es = 0 y se genera la alarma 14095.
- Sólo está programado el ángulo polar AP
  Si no hay programado en la secuencia actual ningún radio polar RP, pero sí un ángulo polar AP, en caso de existir una diferencia entre la posición actual y el polo en coordenadas de pieza se utiliza esta diferencia como radio polar y se memoriza de forma modal. Si la diferencia = 0 se vuelven a especificar las coordenadas polares, y el radio polar modal signe siendo cero.

Ejemplo

mecanizado de una figura de taladro



Las posiciones de los taladros están definidas en coordenadas polares.

Cada taladro se ejecuta con el mismo proceso de fabricación.

Pretaladrado, taladrado a medida, escariado...

La secuencia de mecanizado de dicha geometría se memoriza en un subprograma.

Código de programa	**Comentarios**
`N10 G17 G54`	`; Plano de trabajo X/Y, origen de la pieza.`
`N20 G111 X43 Y38`	`; Definición del polo.`
`N30 G0 RP=30 AP=18 Z5`	`; Desplazamiento al punto inicial, indicación en coordenadas cilíndricas.`
`N40 L10`	`; Llamada de subprograma.`
`N50 G91 AP=72`	`; Desplazamiento a la siguiente posición en rápido, ángulo polar en cotas incrementales, el radio polar permanece memorizado desde la secuencia N30 y no es necesario especificarlo.`
`N60 L10`	`; Llamada de subprograma.`
`N70 AP=IC(72)`	`.`
`N80 L10`	…
`N90 AP=IC(72)`	

Código de programa	Comentarios
N100 L10	…
N110 AP=IC(72)	
N120 L10	…
N130 G0 X300 Y200 Z100 M30	; Retirada de la herramienta, fin del programa.

Consulte también

Vista general (Página 194)

2.9.4 Desplazamiento en rápido

2.9.4.1 Activar desplazamiento en rápido (G0)

El desplazamiento de los ejes de contorneado con velocidad en rápido se activa con el comando G G0.

Sintaxis

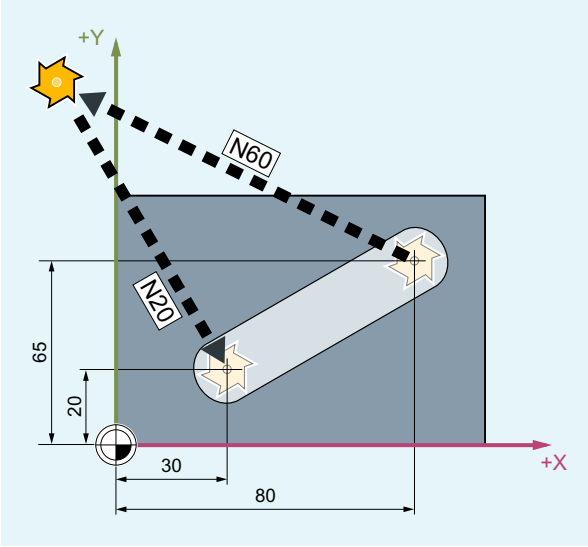
```
G0 X… Y… Z…
G0 RP=… AP=…
```

Descripción

G0:	Desplazamiento de los ejes con velocidad en rápido	
	Eficacia:	modal
X... Y... Z...:	Especificación del punto final en coordenadas cartesianas	
RP=... AP=... :	Especificación del punto final en coordenadas polares	

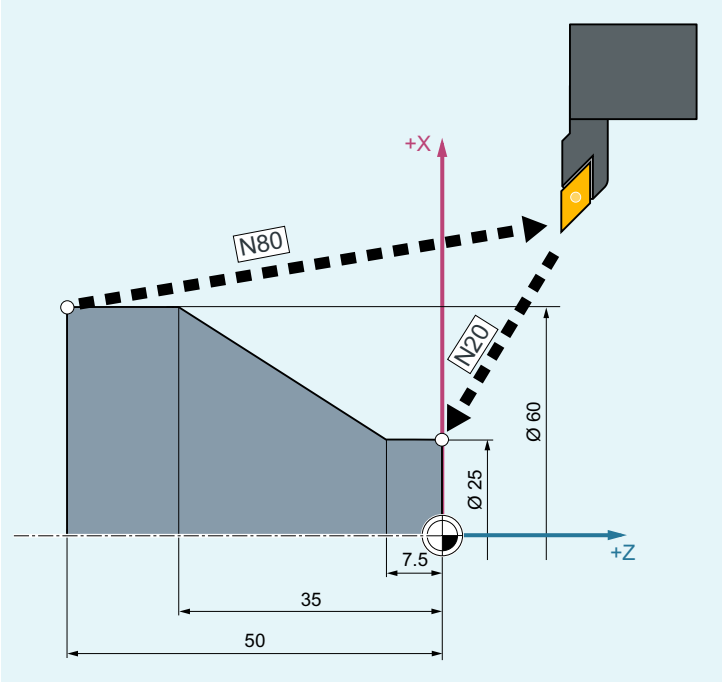
Ejemplos

Ejemplo 1: Fresado



Código de programa	Comentarios
N10 G90 S400 M3	; Acotado absoluto, cabezal a derechas
N20 G0 X30 Y20 Z2	; Desplazamiento a la posición inicial
N30 G1 Z-5 F1000	; Cota de penetración de la herramienta
N40 X80 Y65	; Desplazamiento lineal
N50 G0 Z2	
N60 G0 X-20 Y100 Z100 M30	; Retirada de la herramienta, fin del programa

Ejemplo 2: Torneado



Código de programa	Comentarios
N10 G90 S400 M3	; Acotado absoluto, cabezal a derechas
N20 G0 X25 Z5	; Desplazamiento a la posición inicial
N30 G1 G94 Z0 F1000	; Cota de penetración de la herramienta
N40 G95 Z-7.5 F0.2	
N50 X60 Z-35	; Desplazamiento lineal
N60 Z-50	
N70 G0 X62	
N80 G0 X80 Z20 M30	; Retirada de la herramienta, fin del programa

2.9.4.2 Activar/desactivar la interpolación lineal para desplazamientos en rápido (RTLION, RTLIOF)

Independientemente del ajuste predeterminado (MD20730 $MC_G0_LINEAR_MODE), el comportamiento de interpolación para desplazamientos en rápido también puede ajustarse en el programa de pieza con los comandos del grupo G 55.

Sintaxis

```
RTLIOF
...
RTLION
```

Descripción

RTLIOF:	Comando G para la desactivación de la interpolación lineal ⇒ Con el desplazamiento en rápido (G0) está activa la interpolación **no lineal**. Todos los ejes de contorneado alcanzan su punto final de forma independiente entre sí.	
	Eficacia:	modal
RTLION:	Comando G para la activación de la interpolación lineal ⇒ Con el desplazamiento en rápido (G0) está activa la interpolación **lineal**. Todos los ejes de contorneado alcanzan su punto final simultáneamente.	
	Eficacia:	modal

Nota

Requisitos para RTLIOF

Para que se efectúe una interpolación **no lineal** con RTLIOF deben cumplirse los siguientes requisitos:

- Ninguna transformada activa (TRAORI, TRANSMIT, ...).
- G60 activo (detener al final de la secuencia).
- Ningún compresor activo (COMPOF).
- Ninguna corrección del radio de herramienta activa (G40).
- Ningún volante de contorno seleccionado.
- Ningún punzonado activo.

En caso de que no se cumpla alguno de estos requisitos, con RTLION se efectúa una interpolación lineal.

Ejemplo

```
Código de programa                  Comentarios
                                    ; Está preajustada la interpolación lineal:
                                    ; MD20730 $MC_GO_LINEAR_MODE == TRUE
...
N30 RTLIOF                          ; Desactivar la interpolación lineal.
N40 G0 X0 Y10                       ; Las secuencias G0 se desplazan con interpola-
                                      ción no lineal.
N50 G41 X20 Y20                     ; Corrección del radio de herramienta activa ⇒
                                      las secuencias G0 se desplazan con interpolación
                                      lineal.
N60 G40 X30 Y30                     ; Corrección del radio de herramienta no activa ⇒
                                      las secuencias G0 se desplazan con interpolación
                                      no lineal.
N70 RTLION                          ; Activar interpolación lineal.
...
```

Información adicional

Lectura del comportamiento de interpolación actual

El comportamiento de interpolación actual se puede leer a través de las variables de sistema $AA_G0MODE.

2.9.4.3 Adaptación de tolerancias para desplazamientos en rápido (STOLF, CTOLG0 y OTOLG0)

Las tolerancias configuradas con datos de máquina para desplazamientos en rápido (tolerancias G0) pueden modificarse temporalmente en el programa de pieza. Los ajustes de los datos de máquina no se modifican. Después de un reset del canal o de fin de programa, vuelven a ser efectivas las tolerancias configuradas.

Requisitos

Las tolerancias G0 solo surten efecto si se cumplen las siguientes condiciones:

- Está activa una de las funciones siguientes:
  - Función de compresor COMP...
  - Función de matado de esquinas G642 o G645
  - Matado de esquinas de orientación OST
  - Alisado de orientación ORISON
  - Alisado en orientación relativa a la trayectoria ORIPATH
- Hay varias (≥2) secuencias G0 consecutivas en el programa de pieza.
  Cuando hay una sola secuencia G0, las tolerancias G0 no son efectivas, debido a que en la transición de un movimiento no G0 a un movimiento G0 (y viceversa) siempre se aplica la "**menor tolerancia**" (tolerancia de mecanizado de pieza).

Sintaxis

Adaptación de la tolerancia G0 relativa

```
STOLF=<Value>
```

Adaptación de las tolerancias G0 absolutas

```
CTOLG0=<Value>
OTOLG0=<Value>
```

Descripción

STOLF:	Dirección para programar un factor de tolerancia de efectividad temporal para desplazamientos en rápido			
	<Value>:	Factor de tolerancia G0		
		Tipo:	REAL	
		Valor:	≥ 0:	El factor de tolerancia G0 puede ser mayor o menor que 1.0. Si el factor es 1.0 (valor estándar), son efectivas las mismas tolerancias para desplazamientos en rápido que para desplazamientos no en rápido. Normalmente se ajusta un factor de tolerancia >1.0. El factor de tolerancia G0 programado es efectivo hasta que es sobrescrito por una nueva programación de STOLF, sustituido por una programación CTOLG0/ OTOLG0 o borrado por un reset de canal o de fin de programa.
			< 0:	Borrado del factor de tolerancia programado ⇒ Se aplica nuevamente el factor de tolerancia preajustado en el dato de máquina.
CTOLG0:	Dirección para programar una tolerancia de contorno de efectividad temporal para desplazamientos en rápido			
	<Value>:	Valor absoluto para la tolerancia de contorno		
		Tipo:	REAL	
		Valor:	≥ 0:	El valor absoluto programado para la tolerancia de contorno es efectivo hasta que es sobrescrito por una nueva programación de CTOLG0, sustituido por una programación de STOLF o borrado por un reset de canal o de fin de programa.
			< 0:	Borrado del valor de tolerancia programado ⇒ Se aplica nuevamente el valor de tolerancia preajustado en el dato de máquina.
OTOLG0:	Dirección para programar una tolerancia de orientación de efectividad temporal para desplazamientos en rápido			
	<Value>:	Valor absoluto para la tolerancia de orientación		
		Tipo:	REAL	
		Valor:	≥ 0:	El valor absoluto programado para la tolerancia de orientación es efectivo hasta que es sobrescrito por una nueva programación de OTOLG0, sustituido por una programación de STOLF o borrado por un reset de canal o de fin de programa.
			< 0:	Borrado del valor de tolerancia programado ⇒ Se aplica nuevamente el valor de tolerancia preajustado en el dato de máquina.

Nota

La última dirección programada siempre tiene prioridad, tal como muestran los ejemplos siguientes:

- Al programar CTOLG0 con STOLF, se utiliza el valor de tolerancia programado con CTOLG0 para el matado de esquinas del contorno.
- Asimismo, al programar OTOLG0 con STOLF, se utiliza el valor de tolerancia programado con OTOLG0 para el matado de esquinas de la orientación.
- Una vez reprogramado STOLF, se vuelve a utilizar el factor de tolerancia para la tolerancia de contorno y de orientación.

Ejemplos

Ejemplo 1: Adaptación de la tolerancia G0 relativa

Código de programa	Comentarios
COMPCAD G645 G1 F10000	; Función de compresor COMPCAD
X... Y... Z...	; Aquí actúan los datos de máquina y de operador.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
G0 X... Y... Z...	; Aquí actúa el dato de máquina $MC_G0_TOLERANCE_FACTOR (p. ej., = 3), es decir, una tolerancia de matado de esquinas de: $MC_G0_TOLERANCE_FACTOR * $MA_COMPRESS_POS_TOL
CTOL=0.02	
STOLF=4	
G1 X... Y... Z...	; A partir de aquí actúa una tolerancia de contorno de 0,02 mm.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
X... Y... Z...	; A partir de aquí actúa un factor de tolerancia G0 de 4, es decir, una tolerancia de contorno de 0,08 mm.
...	

Ejemplo 2: Adaptación de las tolerancias G0 absolutas

En los datos de máquina deben estar preajustadas las siguientes tolerancias G0 absolutas:

- Tolerancia de contorno G0: 0.1
- Tolerancia de orientación G0: 1.0

Estas tolerancias deben modificarse de forma temporal en el programa de pieza:

Código de programa	Comentarios
COMPCAD G645 G1 F10000	; Función de compresor COMPCAD

Código de programa	**Comentarios**
X... Y... Z...	; A partir de aquí actúan las tolerancias configuradas de mecanizado de pieza.
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
G0 X... Y... Z...	; Aquí actúan las tolerancias G0 absolutas configuradas.
CTOLG0=0.2 OTOLG0=2.0	; Programación de las tolerancias G0 absolutas.
G1 X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
G0 X... Y... Z...	
X... Y... Z...	; A partir de aquí actúan las tolerancias G0 programadas.
...	

Información adicional

Lectura del factor de tolerancia G0

El factor de tolerancia efectivo actualmente para desplazamientos en rápido se puede leer a través de variables de sistema:

- En acciones síncronas o con parada de decodificación previa en el programa de pieza, a través de la variable del sistema:

  $AC_STOLF — Factor de tolerancia G0 activo

  Factor de tolerancia G0 que estaba activo en el momento de preparar la secuencia del proceso principal actual.

- Sin parada de decodificación previa en el programa de pieza, a través de la variable del sistema:

  $P_STOLF — Factor de tolerancia G0 programado

Si no hay ningún valor programado con STOLF en el programa de pieza activo, estas dos variables de sistema devuelven el valor configurado en el dato de máquina.

Si en una secuencia no hay ningún desplazamiento en rápido (G0) activo, estas variables de sistema suministran siempre el valor 1.

Lectura de las tolerancias G0 absolutas

Las tolerancias absolutas efectivas actualmente para desplazamientos en rápido se pueden leer a través de variables de sistema.

- En acciones síncronas o con parada de decodificación previa en el programa de pieza, a través de las variables de sistema:

  $AC_CTOL_G0_ABS — Tolerancia de contorno activa en movimientos G0

  Tolerancia de contorno G0 que estaba activa en el momento de preparar la secuencia del proceso principal actual.

$AC_OTOL_ G0_ABS	Tolerancia de orientación activa en movimientos G0 Tolerancia de orientación G0 que estaba activa en el momento de preparar la secuencia del proceso principal actual.

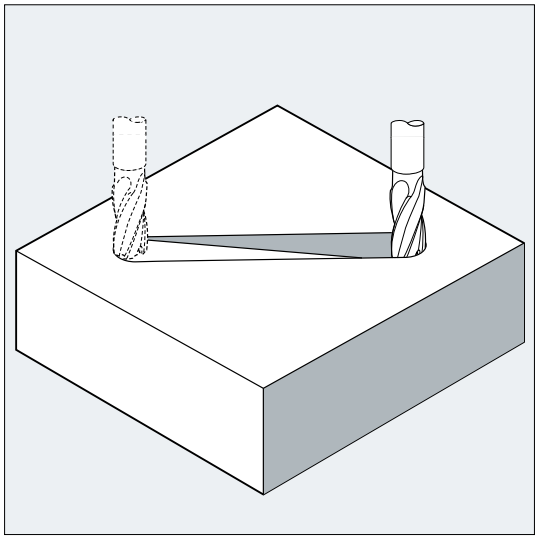
- Sin parada de decodificación previa en el programa de pieza, a través de las variables de sistema:

$P_CTOL_ G0_ABS	Tolerancia de contorno programada en movimientos G0
$P_OTOL_ G0_ABS	Tolerancia de orientación programada en movimientos G0

Si no hay tolerancias G0 absolutas programadas con CTOLG0 ni OTOLG0 en el programa de pieza activo, estas variables del sistema devuelven los valores configurados en los datos de máquina.

2.9.5 Interpolación lineal (G1)

Con `G1`, la herramienta se desplaza a lo largo de una línea recta paralela a un eje, inclinada o bien orientada de cualquier forma en el espacio. La interpolación lineal permite realizar mecanizados en 3D, ranuras, etc.



Sintaxis

```
G1 X… Y… Z … F…
G1 AP=… RP=… F…
```

Descripción

`G1`:	Interpolación lineal (interpolación lineal con avance)
`X... Y... Z...`:	Punto final en coordenadas cartesianas
`AP=...`:	Punto final en coordenadas polares (ángulo polar)

`RP=...`:	Punto final en coordenadas polares (radio polar)
`F...`:	Velocidad de avance en mm/min. La herramienta se desplaza con el avance F a lo largo de una línea recta desde el punto inicial hasta el punto final programado. La posición final se puede introducir en coordenadas cartesianas o polares. La pieza se mecaniza a lo largo de la trayectoria descrita. Ejemplo: `G1 G94 X100 Y20 Z30 A40 F100` El punto final en X, Y, Z se alcanza a la velocidad de 100 mm/min, el eje giratorio A se desplaza como un eje síncrono de forma que los movimientos de los cuatro ejes comienzan y finalizan al mismo tiempo.

Nota

`G1` es modalmente activo.

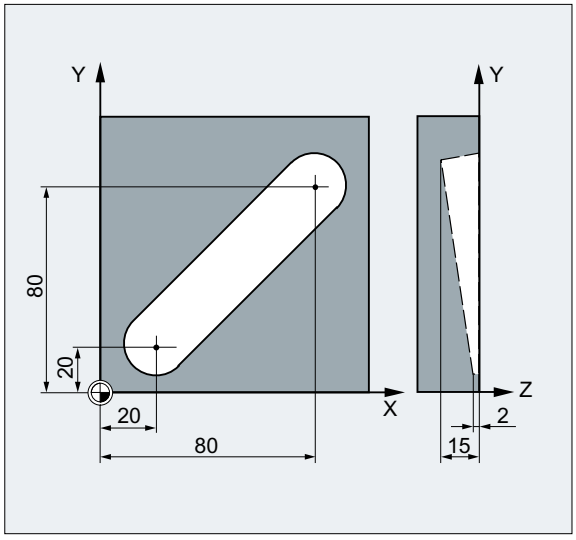
Para el mecanizado deben indicarse la velocidad de giro del cabezal `S` y el sentido de giro del cabezal `M3`/`M4`.

`FGROUP` permite definir grupos de ejes para los cuales se tiene en cuenta el avance de contorneado `F`. Para más información al respecto, ver capítulo "Comportamiento de contorneado".

Ejemplos

Ejemplo 1: mecanizado de una ranura (fresado)

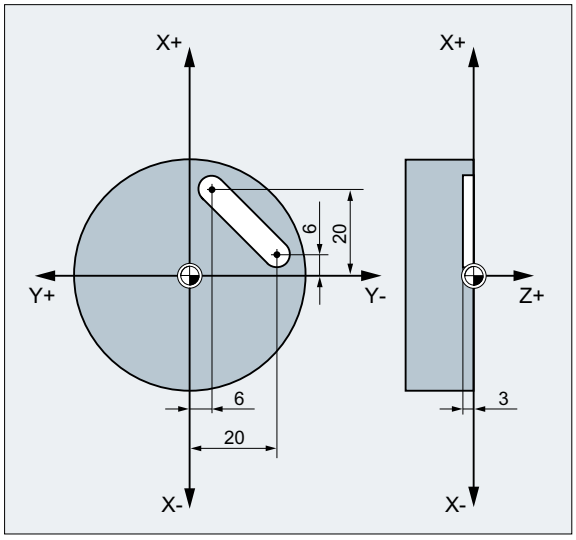
la herramienta se desplaza desde el punto inicial hasta el punto final en el plano X/Y. El avance se tiene en cuenta simultáneamente también en la dirección del eje Z.



Código de programa	**Comentarios**
`N10 G17 S400 M3`	`; Selección del plano de trabajo, cabezal a derechas`
`N20 G0 X20 Y20 Z2`	`; Desplazamiento a la posición inicial`
`N30 G1 Z-2 F40`	`; Cota de penetración de la herramienta`
`N40 X80 Y80 Z-15`	`; Desplazamiento siguiendo una recta inclinada`

Código de programa	Comentarios
N50 G0 Z100 M30	; Retirada al punto de cambio de herramienta

Ejemplo 2: mecanizado de una ranura (torneado)



Código de programa	Comentarios
N10 G17 S400 M3	; Selección del plano de trabajo, cabezal a derechas
N20 G0 X40 Y-6 Z2	; Desplazamiento a la posición inicial
N30 G1 Z-3 F40	; Cota de penetración de la herramienta
N40 X12 Y-20	; Desplazamiento siguiendo una recta inclinada
N50 G0 Z100 M30	; Retirada al punto de cambio de herramienta

2.9.6 Interpolación circular

2.9.6.1 Vista general

La interpolación circular permite mecanizar círculos completos o bien arcos de circunferencia.

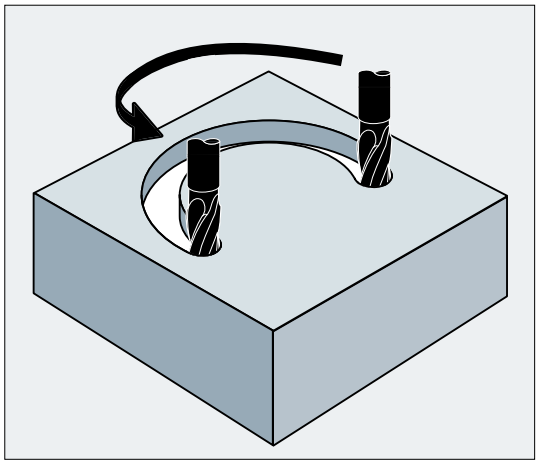


Figura 2-3 Ejemplo de aplicación: Fresado de una ranura circular

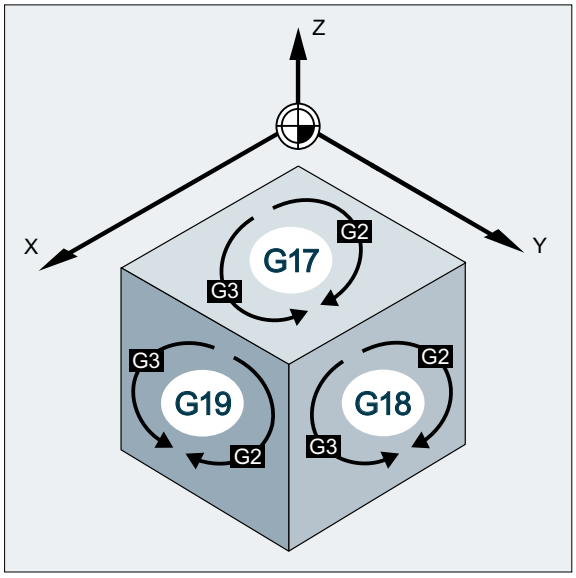
Variantes de programación

El control ofrece distintas posibilidades de programación de movimientos circulares. Esto permite al usuario aplicar de forma directa prácticamente todos los tipos de acotados de plano.

- Interpolación circular con centro y punto final (G2/G3, X... Y... Z..., I... J... K...) (Página 195)
- Interpolación circular con radio y punto final (G2/G3, X... Y... Z..., CR) (Página 197)
- Interpolación circular con ángulo en el vértice y punto final / centro (G2/G3, X... Y... Z... / I... J... K..., AR) (Página 199)
- Interpolación circular con coordenadas polares (G2/G3, AP, RP) (Página 201)
- Interpolación circular con punto intermedio y punto final (CIP, X... Y... Z..., I1... J1... K1...) (Página 203)
- Interpolación circular con transición tangencial (CT, X... Y... Z...) (Página 205)

Plano para la interpolación circular

Para poder calcular la dirección de rotación del círculo (G2 es sentido horario y G3 es sentido antihorario), el control numérico necesita conocer el plano de trabajo seleccionado (Página 149).



Excepción:

También es posible mecanizar círculos fuera del plano de trabajo seleccionado (aunque no con el ángulo en el vértice y los parámetros helicoidales). En este caso, las direcciones de los ejes especificadas por el programador como punto final del círculo determinan el plano del arco de circunferencia.

2.9.6.2 Interpolación circular con centro y punto final (G2/G3, X... Y... Z..., I... J... K...)

Variante de interpolación circular que utiliza el **centro** y el **punto final** de un elemento de contorno circular para la interpolación.

Si se define el círculo sin punto final, el control describe una circunferencia completa.

Sintaxis

```
G2/G3 X… Y… Z… I… J… K…
G2/G3 X… Y… Z… I=AC(…) J=AC(…) K=(AC…)
```

Descripción

G2:	Interpolación circular en sentido horario	
	Eficacia:	modal
G3:	Interpolación circular en sentido antihorario	
	Eficacia:	modal

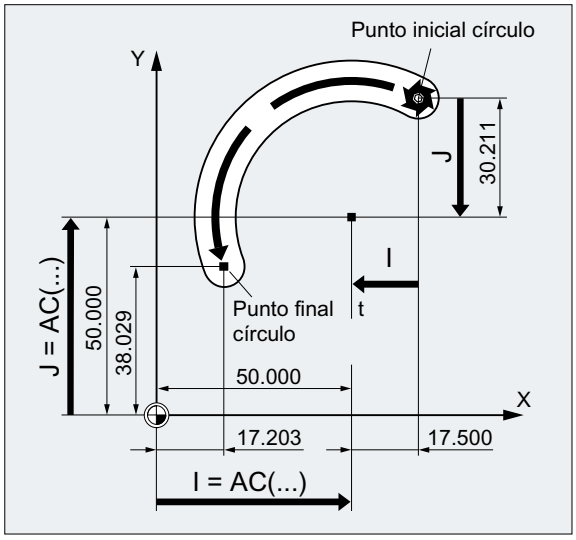
`X... Y... Z...`:	Punto final del círculo en coordenadas cartesianas Dependiendo del ajuste de medida `G90`/`G91` o bien `...=AC(...)` / `...=IC(...)` válido actualmente, las coordenadas del punto final del círculo se interpretan en cotas absolutas o en cotas incrementales.
`I... J... K...`:	Parámetros de interpolación para especificar las coordenadas del centro del círculo en dirección X, Y, Z. Las coordenadas del centro del círculo se especifican de forma estándar en cotas incrementales con respecto al punto inicial del círculo. Si las coordenadas del centro del círculo deben especificarse en cotas absolutas con respecto al origen de la pieza, los parámetros de interpolación I, J, K deben programarse de la siguiente forma: `I=AC(…) J=AC(…) K=AC(…)` **Nota** Un parámetro de interpolación con valor "0" se puede omitir, pero siempre se debe indicar el segundo parámetro asociado.

Nota

El ajuste estándar `G90`/`G91` especificado en cotas absolutas o incrementales sólo es válido para el punto final del círculo.

Ejemplos

Ejemplo 1: fresado



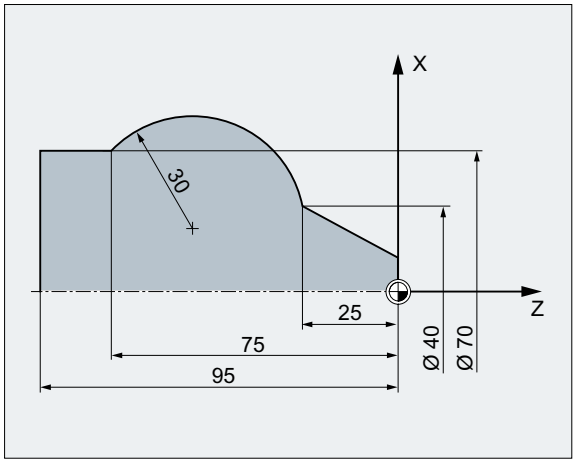
Introducción del centro en cotas incrementales

```
N10 G0 X67.5 Y80.211
N20 G3 X17.203 Y38.029 I-17.5 J-30.211 F500
```

Introducción del centro en cotas absolutas

```
N10 G0 X67.5 Y80.211
N20 G3 X17.203 Y38.029 I=AC(50) J=AC(50)
```

Ejemplo 2: torneado



Introducción del centro en cotas incrementales

```
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
```

Introducción del centro en cotas absolutas

```
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 Interpolación circular con radio y punto final (G2/G3, X... Y... Z..., CR)

Variante de interpolación circular que utiliza el **radio** y el **punto final** de un elemento de contorno circular para la interpolación.

Nota

Con esta variante **no** es posible programar circunferencias completas (ángulo de desplazamiento de 360°).

Sintaxis

```
G2/G3 X… Y… Z… CR=±...
```

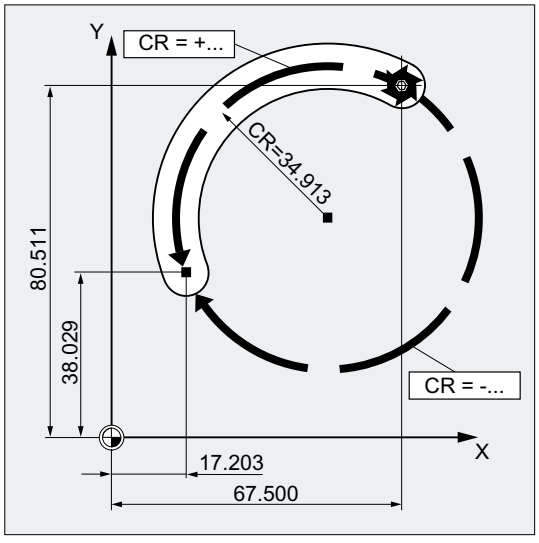
Descripción

G2:	Interpolación circular en sentido horario	
	Eficacia:	modal

G3:	Interpolación circular en sentido antihorario	
	Eficacia:	modal
X... Y... Z...:	Punto final del círculo en coordenadas cartesianas Dependiendo del ajuste de medida G90/G91 o bien ...=AC(...) / ...=IC(...) válido actualmente, las coordenadas del punto final se interpretan en cotas absolutas o en cotas incrementales.	
CR=±...:	Radio del círculo Con el signo correspondiente se especifica el ángulo de desplazamiento: positivo si es mayor que 180° o negativo si es menor que dicho valor. El signo positivo se puede omitir.	
	CR=+...:	Ángulo de desplazamiento ≤ 180°
	CR=-...:	Ángulo de desplazamiento > 180 °
	Nota En la práctica no existe limitación importante para el máximo valor programable del radio.	

Ejemplos

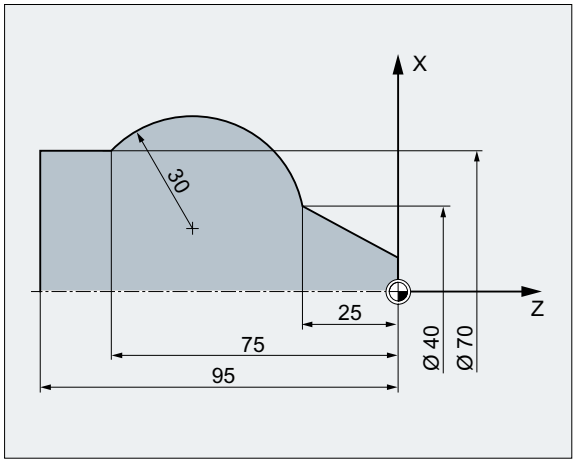
Ejemplo 1: fresado



Código de programa

```
N10 G0 X67.5 Y80.511
N20 G3 X17.203 Y38.029 CR=34.913 F500
...
```

Ejemplo 2: torneado



Código de programa

```
...
N125 G1 X40 Z-25 F0.2
N130 G3 X70 Z-75 CR=30
N135 G1 Z-95
...
```

2.9.6.4 Interpolación circular con ángulo en el vértice y punto final / centro (G2/G3, X... Y... Z... / I... J... K..., AR)

Variante de interpolación circular que utiliza el **ángulo en el vértice** y el **centro** o el **punto final** de un elemento de contorno circular para la interpolación.

Nota

Con esta variante **no** es posible programar circunferencias completas (ángulo de desplazamiento de 360°).

Sintaxis

```
G2/G3 X… Y… Z… AR=...
G2/G3 I… J… K… AR=...
```

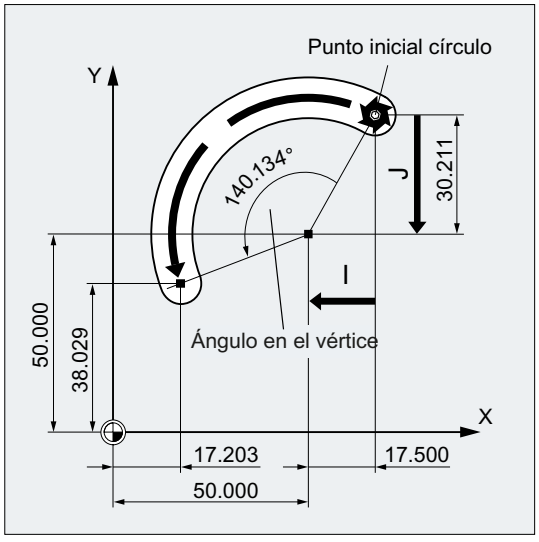
Descripción

G2:	Interpolación circular en sentido horario	
	Eficacia:	modal
G3:	Interpolación circular en sentido antihorario	
	Eficacia:	modal

`X... Y... Z...`:	Punto final del círculo en coordenadas cartesianas Dependiendo del ajuste de medida `G90`/`G91` o bien `...=AC(...)` / `...=IC(...)` válido actualmente, las coordenadas del punto final del círculo se interpretan en cotas absolutas o en cotas incrementales.	
`I... J... K...`:	Parámetros de interpolación para especificar las coordenadas del centro del círculo en dirección X, Y, Z. Las coordenadas del centro del círculo se especifican de forma estándar en cotas incrementales con respecto al punto inicial del círculo. Si las coordenadas del centro del círculo deben especificarse en cotas absolutas con respecto al origen de la pieza, los parámetros de interpolación I, J, K deben programarse de la siguiente forma: `I=AC(...) J=AC(...) K=AC(...)` **Nota** Un parámetro de interpolación con valor "0" se puede omitir, pero siempre se debe indicar el segundo parámetro asociado.	
`AR=...`:	Ángulo en el vértice	
	Rango de valores:	0° ... 360°

Ejemplos

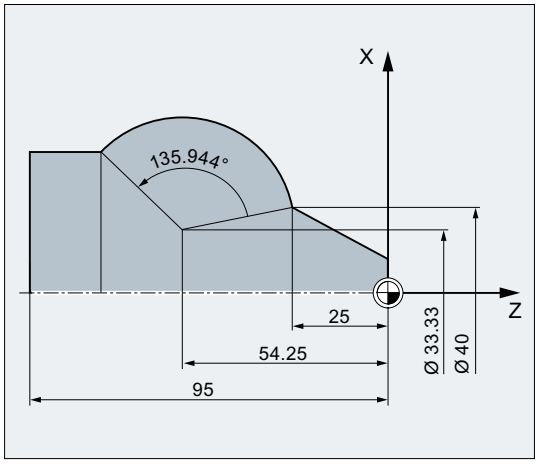
Ejemplo 1: fresado



Código de programa

```
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
```

Ejemplo 2: torneado



Código de programa

```
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 Interpolación circular con coordenadas polares (G2/G3, AP, RP)

Variante de interpolación circular que utiliza el **punto final del círculo en coordenadas polares** para la interpolación.

Se tienen en cuenta las siguientes reglas:

- El polo se encuentra en el centro del arco de circunferencia.
- El radio polar coincide con el radio del círculo.

Sintaxis

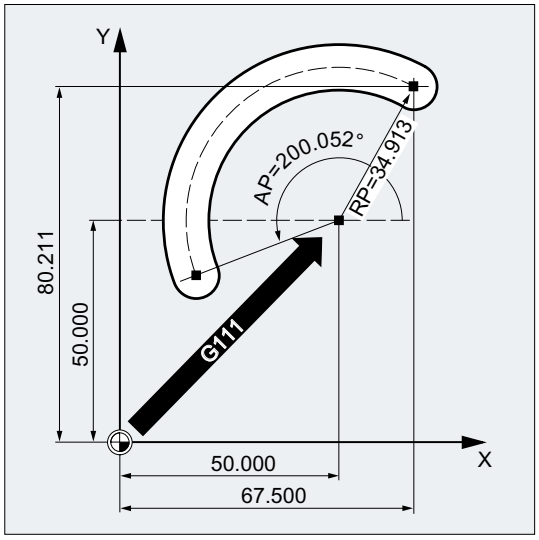
```
G2/G3 AP=... RP=...
```

Descripción

G2:	Interpolación circular en sentido horario	
	Eficacia:	modal
G3:	Interpolación circular en sentido antihorario	
	Eficacia:	modal
AP=... RP=...:	Punto final del círculo en coordenadas polares	
	AP=...:	Ángulo polar
	RP=...:	Radio polar (≙ radio del círculo)

Ejemplos

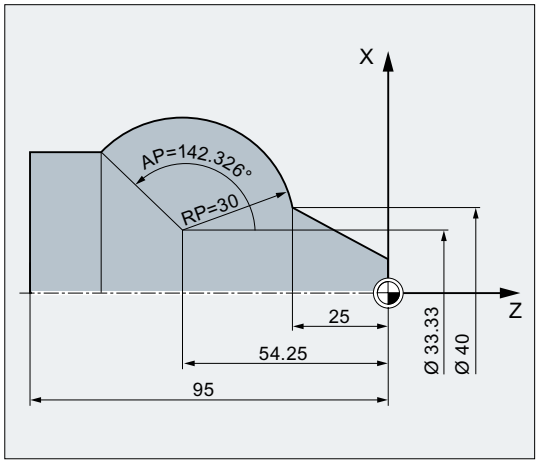
Ejemplo 1: fresado



Código de programa

```
N10 G0 X67.5 Y80.211
N20 G111 X50 Y50
N30 G3 RP=34.913 AP=200.052 F500
```

Ejemplo 2: torneado



Código de programa

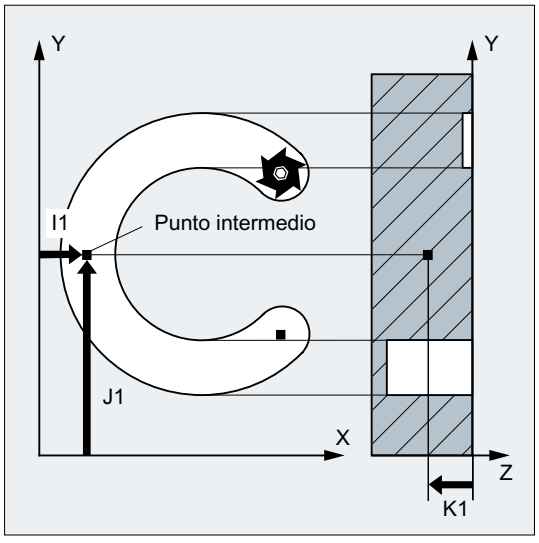
```
N125 G1 X40 Z-25 F0.2
N130 G111 X33.33 Z-54.25
N135 G3 RP=30 AP=142.326
N140 G1 Z-95
```

2.9.6.6 Interpolación circular con punto intermedio y punto final (CIP, X... Y... Z..., I1... J1... K1...)

La variante de interpolación circular programada con el comando G `CIP` permite interpolar arcos de circunferencia inclinados.

El **punto intermedio** y el **punto final** del contorno circular describen la trayectoria del movimiento circular.

La dirección del desplazamiento viene determinada por el siguiente orden: punto inicial → punto intermedio → punto final.



Sintaxis

```
CIP X… Y… Z… I1=AC(…) J1=AC(…) K1=(AC…)
```

Descripción

`CIP`:	Interpolación circular a través de punto intermedio	
	Eficacia:	Modal
`X... Y... Z...`:	Punto final del círculo en coordenadas cartesianas Dependiendo del ajuste de medida `G90`/`G91` o bien `...=AC(...)` / `...=IC(...)` válido actualmente, las coordenadas del punto final del círculo se interpretan en cotas absolutas o en cotas incrementales.	
`I1... J1... K1...`:	Parámetros de interpolación para especificar las coordenadas del punto intermedio del círculo en dirección X, Y, Z. Dependiendo del ajuste de medida `G90`/`G91` o bien `...=AC(...)` / `...=IC(...)` válido actualmente, las coordenadas del punto intermedio del círculo se interpretan en cotas absolutas o en cotas incrementales. **Nota** Un parámetro de interpolación con valor "0" se puede omitir, pero siempre se debe indicar el segundo parámetro asociado.	

Nota

Los ajustes estándar `G90`/`G91` especificados en cotas absolutas o incrementales sólo son válidos para el punto intermedio y el punto final del círculo.

Si está activa la indicación en cotas incrementales `G91` o bien `...=IC(...)`, el punto inicial del círculo sirve de referencia para el punto intermedio y el punto final.

Nota

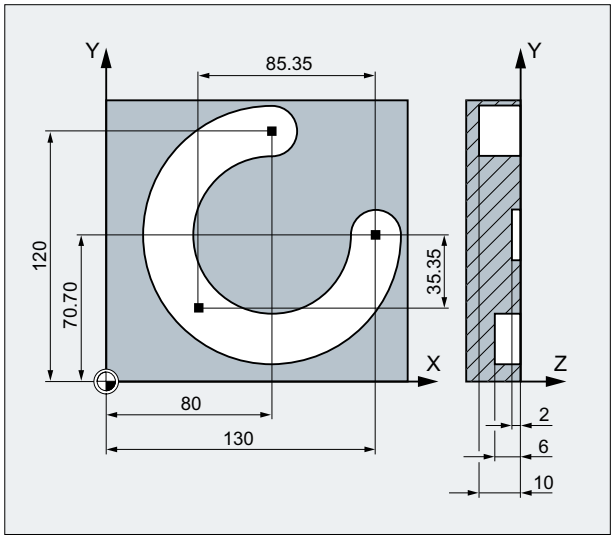
Tecnología de torneado

La programación de círculos con `CIP` no soporta la programación por diámetro del parámetro de interpolación para el eje de refrentado. Por consiguiente, el parámetro de interpolación para el eje de refrentado debe programarse en **Radio**.

Ejemplos

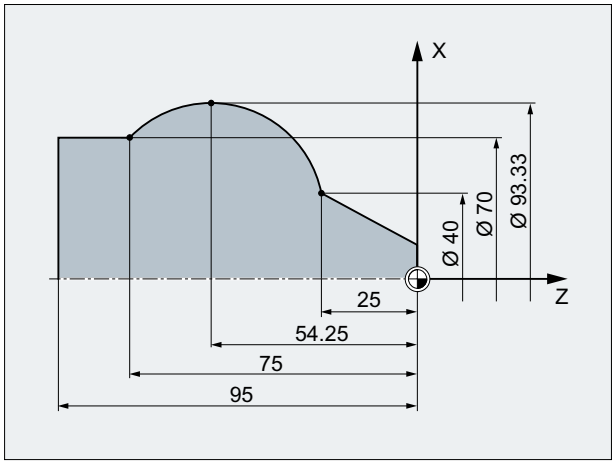
Ejemplo 1: Fresado

Para mecanizar una ranura circular inclinada, se necesita indicar un punto intermedio con 3 parámetros de interpolación y el punto final también con sus 3 coordenadas.



Código de programa	Comentarios
`N10 G0 G90 X130 Y70.70 S800 M3`	`; Desplazamiento al punto inicial.`
`N20 G17 G1 Z-2 F100`	`; Cota de penetración de la herramienta.`
`N30 CIP X80 Y120 Z-10 I1=IC(-85.35) J1=IC(-35.35) K1=-6`	`; Punto final del arco y punto intermedio.`
	`; Coordenadas para los 3 ejes geométricos.`
`N40 M30`	`; Fin del programa.`

Ejemplo 2: Torneado



Código de programa	**Comentarios**
...	
N125 G1 G90 X40 Z-25 F0.2	
N130 CIP X70 Z-75 **I1=IC(26.665)** K1=IC(-29.25)	; El parámetro de interpolación I1 para eje de refrentado debe estar programado en el radio.
; o bien	
; N130 CIP X70 Z-75 **I1=46.665** K1=-54.25	
N135 G1 Z-95	

2.9.6.7 Interpolación circular con transición tangencial (CT, X... Y... Z...)

La variante de interpolación circular programada con el comando G `CT` permite interpolar arcos de circunferencia que empalman tangencialmente con el elemento de contorno programado previamente.

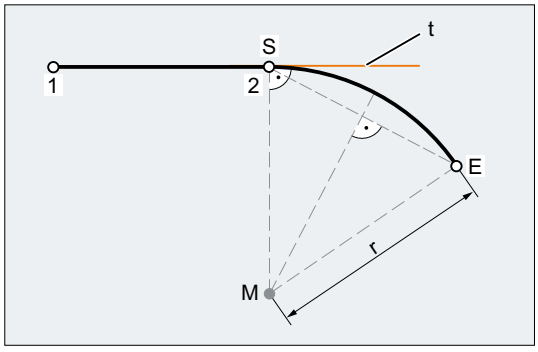
El círculo viene determinado por el **punto inicial y final** y por la **dirección tangencial en el punto inicial**.

Nota

Dirección tangencial en el punto inicial

La dirección de la tangente en el punto inicial de una secuencia CT queda definida por la tangente final al elemento de contorno programado en la secuencia anterior que contenga un desplazamiento.

Entre esta secuencia y la secuencia actual se puede insertar una cantidad cualquiera de secuencias sin desplazamiento.



S Punto inicial
E Punto final
M Centro del círculo
r Radio del círculo
t Tangente final al elemento de contorno programado en la secuencia anterior que contenga un desplazamiento

Figura 2-4 Trayectoria circular S-E que empalma con el segmento de recta 1-2

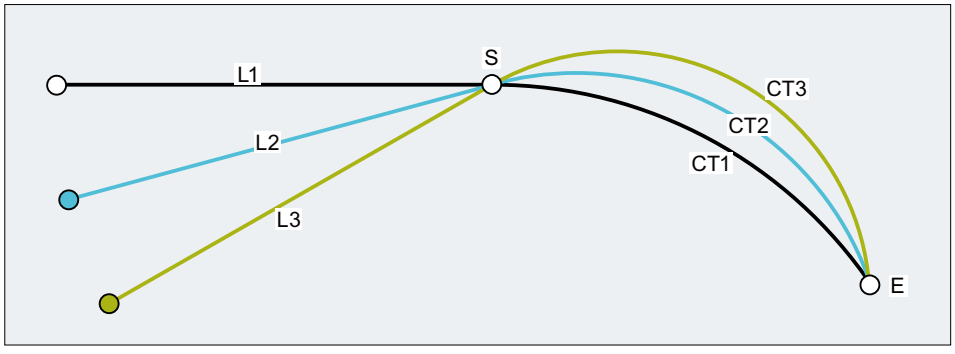


Figura 2-5 Las trayectorias circulares que empalman tangencialmente dependen del elemento de contorno inmediatamente anterior

Sintaxis

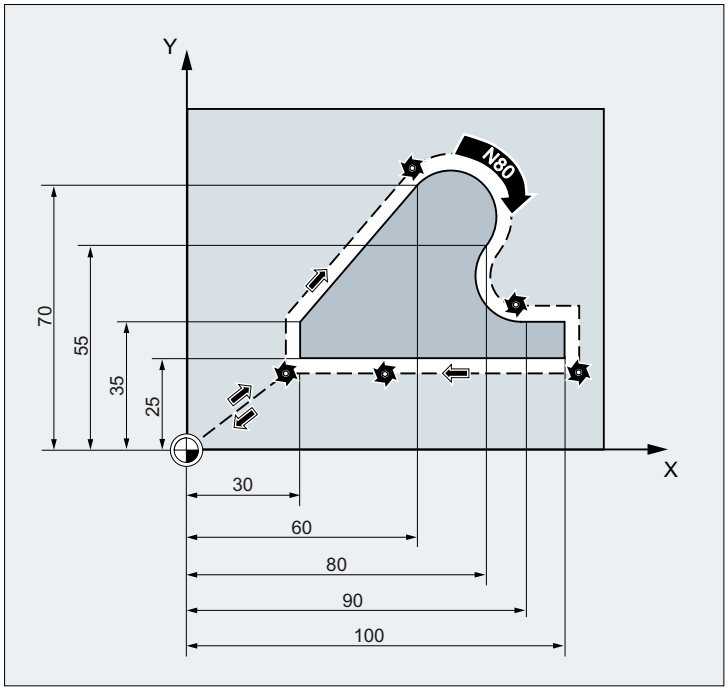
```
CT X... Y... Z...
```

Descripción

`CT`:	Interpolación circular con transición tangencial	
	Eficacia:	modal
`X... Y... Z...`:	Punto final del círculo en coordenadas cartesianas Dependiendo del ajuste de medida `G90`/`G91` o bien `...=AC(...)` / `...=IC(...)` válido actualmente, las coordenadas del punto final del círculo se interpretan en cotas absolutas o en cotas incrementales.	

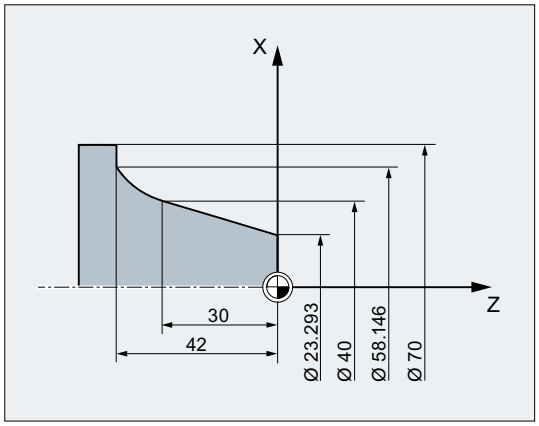
Ejemplos

Ejemplo 1: fresado



Código de programa	Comentarios
N10 G0 Z100	
N20 G17 T1 M6	
N30 G0 X0 Y0 Z2 M3 S300 D1	
N40 Z-5 F1000	; Penetración de herramienta.
N50 G41 X30 Y25 G1 F1000	; Activar corrección del radio de herramienta.
N60 Y35	; Fresado del contorno.
N70 X60 Y70	
N80 CT X80 Y55	**; Programación de círculos definiendo transición tangencial.**
N90 X90 Y35	
N100 G1 X100	
N110 Y25	
N120 X30	
N130 G0 G40 X0 Y0	; Desactivar corrección del radio de herramienta.
N140 Z100	; Retirar herramienta.
N140 M30	

Ejemplo 2: torneado



Código de programa	Comentarios
...	
N110 G1 X23.293 Z0 F10	
N115 X40 Z-30 F0.2	
N120 CT X58.146 Z-42	**; Programación de círculos definiendo transición tangencial.**
N125 G1 X70	
...	

Información adicional

Splines

Al utilizar splines, la dirección de la tangente queda definida mediante la recta que pasa por los últimos dos puntos. Esta dirección no es idéntica a la dirección del spline en su punto final cuando están activados ENAT o EAUTO y cuando se utilizan splines del tipo A y C.

La transición de splines B siempre es tangencial y la dirección de la tangente queda definida cuando está activado ETAN al igual que para los splines del tipo A y C.

Cambio de frame

Si se produce un cambio de frame entre la secuencia que define la tangente y la secuencia CT, la tangente también se ve afectada por dicho cambio.

Caso extremo

Si la dirección de la tangente es tal que pasa por el punto final, se genera una línea recta en lugar de una tangente (caso extremo: círculo con radio infinito). En este caso especial, o bien no se programa TURN o bien se programa TURN=0.

Nota

Cuando nos acercamos al caso extremo, se generan arcos con un radio ilimitado, de forma que cuando se programa TURN distinto de 0, el mecanizado se interrumpe debido a una alarma por violación del final de carrera software.

Posición del plano del arco de circunferencia

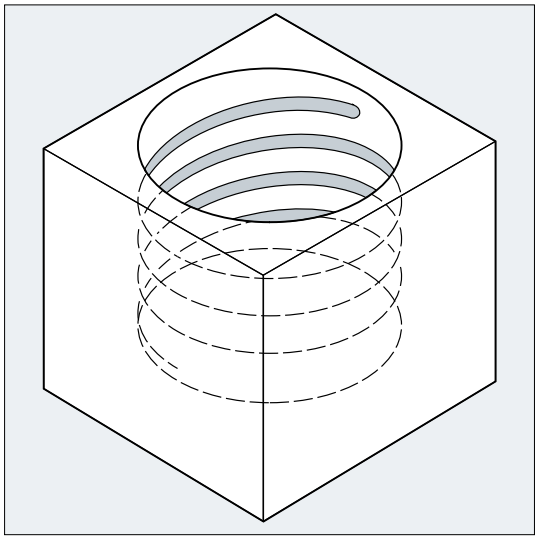
La posición del plano del arco de circunferencia depende del plano activado (G17-G19).

Si la tangente de la secuencia anterior no se encuentra en el plano activado, se utiliza su proyección en el plano activado.

Si los puntos inicial y final no tienen los mismos componentes de posición respecto a la perpendicular del plano activado, en lugar de un arco, se genera una hélice.

2.9.7 Interpolación helicoidal (G2/G3, TURN)

La interpolación helicoidal permite crear roscas o ranuras de lubricación.



En la interpolación helicoidal se combinan dos desplazamientos que se ejecutan de forma paralela:

- Un movimiento circular plano
- Un movimiento lineal vertical superpuesto al anterior

Sintaxis

```
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=
```

Descripción

`G2`:	Desplazamiento circular en sentido horario
`G3`:	Desplazamiento circular en sentido antihorario
`X Y Z`:	Punto final en coordenadas cartesianas

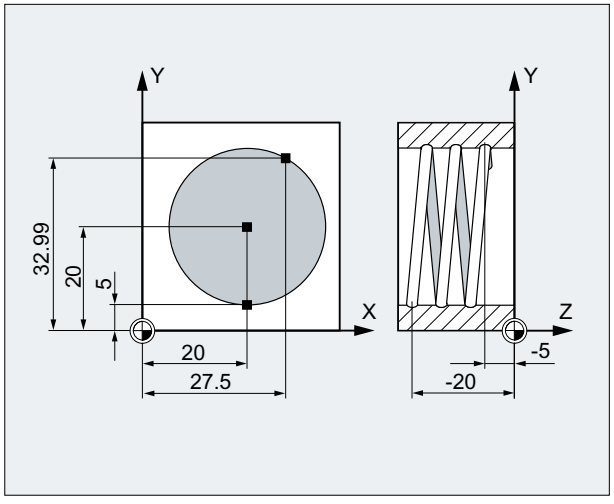
I J K:	Coordenadas cartesianas del centro de la circunferencia
AR:	Ángulo en el vértice
TURN= :	Cantidad de pasadas adicionales por la circunferencia, rango de 0 a 999
AP= :	Ángulo polar
RP= :	Radio polar

Nota

G2 y G3 son modalmente activos.

El desplazamiento circular se realiza en el plano de trabajo previamente definido.

Ejemplo



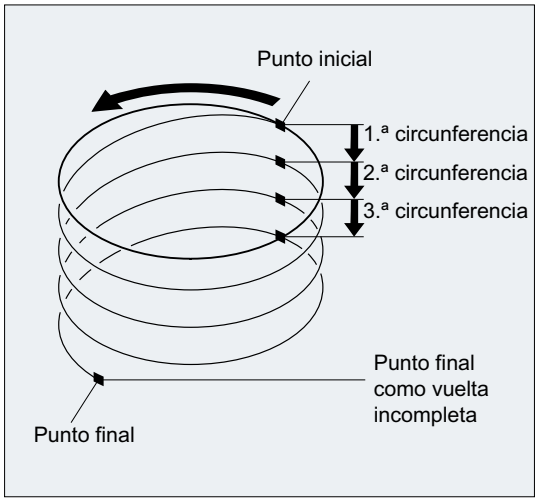
Código de programa	Comentarios
N10 G17 G0 X27.5 Y32.99 Z3	; Desplazamiento a la posición inicial.
N20 G1 Z-5 F50	; Cota de penetración de la herramienta.
N30 G3 X20 Y5 Z-20 I=AC(20) J=AC(20) TURN=2	; Línea helicoidal con los datos: ejecutar 2 circunferencias completas desde la posición inicial, a continuación, efectuar desplazamiento al punto final.
N40 M30	; Fin del programa.

Información adicional

Secuencia de movimiento

1. Desplazamiento al punto inicial
2. Ejecución del círculo completo programado con `TURN=`.
3. Desplazamiento al punto final del círculo, p. ej., siguiendo una circunferencia incompleta.
4. Desplazamiento a los puntos 2 y 3 mediante la profundidad de penetración.

El paso de la espiral que se pretende mecanizar se calcula a partir del número de círculos completos programados, más la profundidad asociada a la última circunferencia incompleta.



Programación del punto final interpolación helicoidal

Para una explicación más detallada de los parámetros de interpolación, ver el capítulo Interpolación circular.

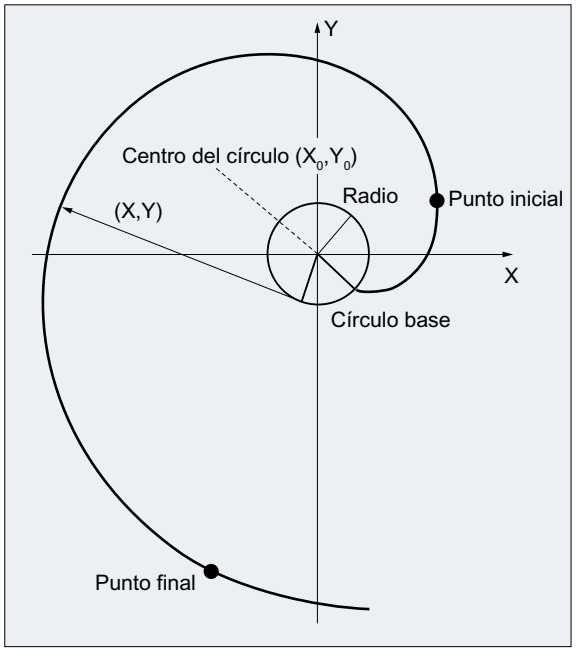
Avance programado

Es aconsejable especificar una corrección del avance programado (`CFC`) para interpolaciones helicoidales. `FGROUP` se puede utilizar para definir qué ejes van a desplazarse con el avance programado. Para más información, ver el capítulo Comportamiento de contorneado.

2.9.8 Interpolación de evolutas (INVCW, INVCCW)

La evoluta de la circunferencia es una curva descrita por el punto final de un hilo tensado, desarrollado desde una circunferencia.

La interpolación de evoluta posibilita trayectorias a lo largo de una evoluta. Se ejecuta en el plano en el cual está definido el círculo básico y va desde el punto inicial hasta el punto final programados.



La programación del punto final puede realizarse de dos formas:

1. Directamente mediante coordenadas cartesianas
2. Indirectamente indicando un ángulo en el vértice (a este respecto, ver también la programación del ángulo en el vértice en la programación del círculo)

Si los puntos inicial y final no se sitúan en el plano del círculo básico, se produce, de forma análoga a la interpolación helicoidal en círculos, una superposición respecto a una curva en el espacio.

En caso de especificación adicional de trayectorias verticales al plano activo, se puede (de forma comparable a la interpolación helicoidal en círculos) desplazar una evoluta en el espacio.

Sintaxis

```
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=...
```

Descripción

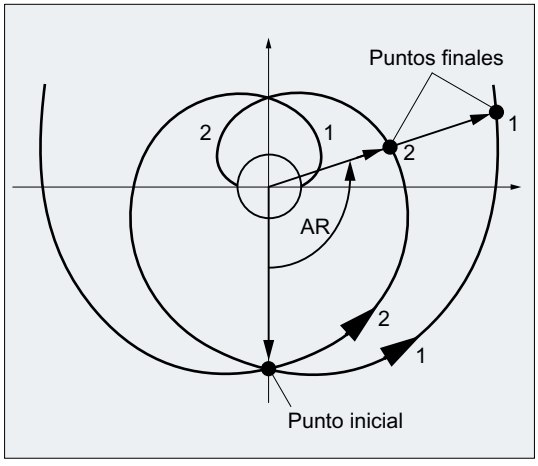
`INVCW`:	Comando para el desplazamiento a una evoluta en sentido horario
`INVCCW`:	Comando para el desplazamiento a una evoluta en sentido antihorario
`X... Y... Z...`:	Programación directa del punto final en coordenadas cartesianas
`I... J... K...`:	Parámetros de interpolación para describir el centro del círculo básico en coordenadas cartesianas **Nota:** las coordenadas introducidas se refieren al punto inicial de la evoluta.
`CR=...`:	Radio del círculo básico

AR=... :	Programación indirecta del punto final indicando un ángulo en el vértice (ángulo de rotación) El origen del ángulo en el vértice es la línea recta que va desde el centro del arco hasta el punto inicial.	
	AR > 0:	La trayectoria en las evolutas se desplaza **alejándose del círculo básico**.
	AR < 0:	La trayectoria en las evolutas se desplaza **acercándose al círculo básico**. Para AR < 0, el ángulo de rotación máximo queda limitado por el hecho de que el punto final debe situarse siempre fuera del círculo básico.

Programación indirecta del punto final indicando un ángulo en el vértice

ATENCIÓN
Ángulo en el vértice, no definido
En la programación indirecta del punto final indicando un ángulo en el vértice AR debe tenerse en cuenta el signo del ángulo, ya que un cambio de signo tendría como consecuencia otra evoluta y, por ello, otra trayectoria.

El siguiente ejemplo lo explica:

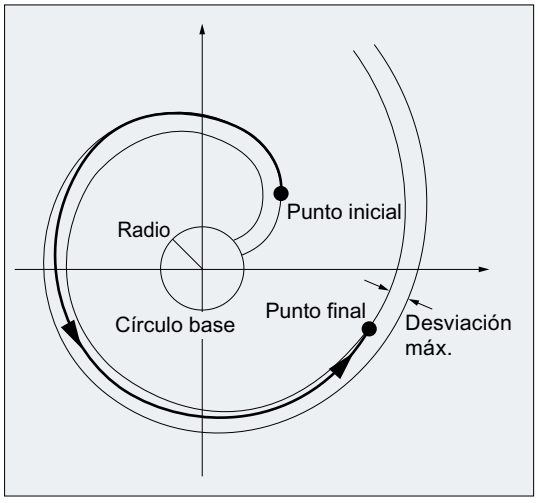


Para las evolutas 1 y 2 coinciden el radio y el centro indicados del círculo básico, así como del punto inicial y del sentido de giro (INVCW/INVCCW). La única diferencia radica en el signo del ángulo en el vértice:

- Con AR > 0, la trayectoria se desplaza sobre la evoluta 1 y se aproxima al punto final 1.
- Con AR < 0, la trayectoria se desplaza sobre la evoluta 2 y se aproxima al punto final 2.

Condiciones

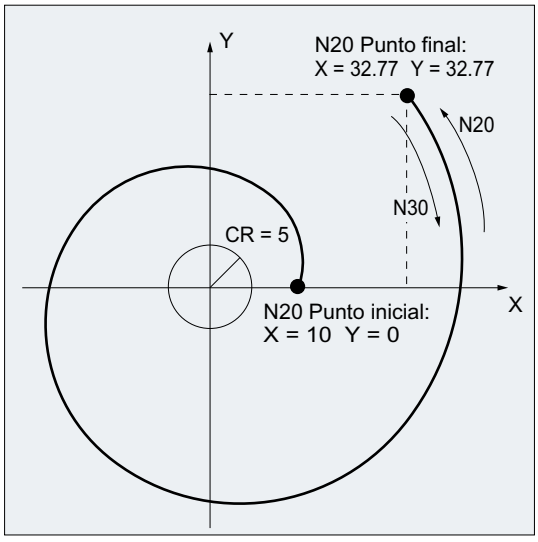
- Tanto el punto inicial como el punto final deben situarse fuera de la superficie del círculo básico de la evoluta (círculo con radio CR alrededor del centro definido con I, J, K). Si esta condición no se cumple, se genera una alarma y se interrumpe la ejecución del programa.
- Ambas posibilidades de programación del punto final (directamente mediante coordenadas cartesianas o indirectamente indicando un ángulo en el vértice) se excluyen mutuamente. Por ello, en una secuencia sólo debe utilizarse una de las dos posibilidades de programación.
- Si el punto final programado no se sitúa exactamente en la evoluta definida por el punto inicial y el círculo básico, se efectúa la interpolación entre las dos evolutas definidas por los puntos inicial o final (ver la figura siguiente).



La desviación máxima del punto final se determina con un dato de máquina (→ fabricante de la máquina). Si la desviación del punto final programado en dirección radial es superior al valor establecido por este DM, se genera una alarma y se interrumpe la ejecución del programa.

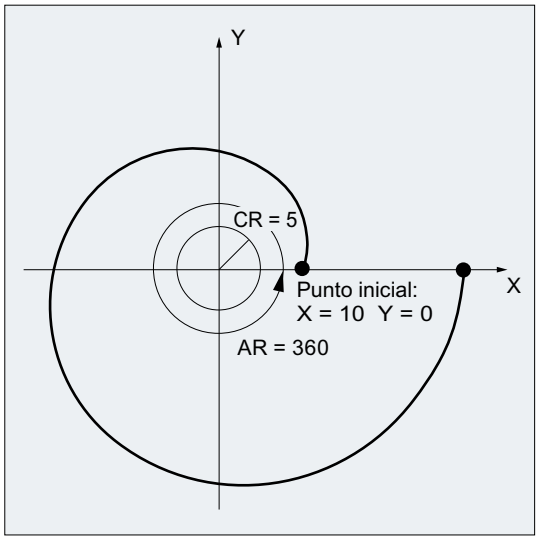
Ejemplos

Ejemplo 1: evoluta con giro a la izquierda desde el punto inicial hasta el punto final programado y retorno como evoluta con giro a la derecha



Código de programa	Comentarios
N10 G1 X10 Y0 F5000	; Desplazamiento a la posición inicial.
N15 G17	; Selección del plano X/Y como plano de trabajo.
N20 INVCCW X32.77 Y32.77 CR=5 I-10 J0	; Evoluta en sentido antihorario, punto final en coordenadas cartesianas.
N30 INVCW X10 Y0 CR=5 I-32.77 J-32.77	; Evoluta en sentido horario, el punto inicial es el punto final de N20, el nuevo punto final es el punto inicial de N20, el nuevo centro del círculo se refiere al nuevo punto inicial y es igual que el anterior centro del círculo.
...	

Ejemplo 2: evoluta con giro a la izquierda con programación indirecta del punto final indicando un ángulo en el vértice



Código de programa	Comentarios
`N10 G1 X10 Y0 F5000`	`; Desplazamiento a la posición inicial.`
`N15 G17`	`; Selección del plano X/Y como plano de trabajo.`
`N20 INVCCW CR=5 I-10 J0 AR=360`	`; Evoluta en sentido antihorario alejándose del círculo básico (puesto que se indica un ángulo positivo) con una vuelta completa (360 grados).`
`...`	

2.9.9 Sucesión de elementos de contorno

2.9.9.1 Programación de la sucesión de elementos de contorno

Función

La programación de la sucesión de elementos de contorno sirve para introducir rápidamente contornos simples.

Pueden programarse sucesiones de elementos de contorno con 1, 2, 3 o más puntos con los elementos de transición chaflán o redondeo indicando coordenadas cartesianas y/o ángulos (ANG o ANG1 y ANG2).

En las secuencias que describen sucesiones de elementos de contorno puede utilizarse cualquier otra dirección CN, como, p. ej., letras de dirección para otros ejes (ejes individuales

o eje perpendicular al plano de mecanizado), datos de funciones auxiliares, comandos G, velocidades, etc.

Nota

Calculadora de contornos

La programación de la sucesión de elementos de contorno también puede realizarse fácilmente con ayuda de una calculadora de contornos. Se trata de una herramienta de la interfaz de usuario que permite la programación y la representación gráfica de contornos de pieza simples y complejos. Los contornos programados por medio de la calculadora de contornos se incorporan en el programa de pieza.

Información adicional: Manual del usuario

Parametrización

Los identificadores para ángulo, radio y chaflán se definen a través de datos de máquina.

MD10652 $MN_CONTOUR_DEF_ANGLE_NAME (nombre del ángulo para sucesiones de elementos de contorno)

MD10654 $MN_RADIUS_NAME (nombre del radio para sucesiones de elementos de contorno)

MD10656 $MN_CHAMFER_NAME (nombre del chaflán para sucesiones de elementos de contorno)

Nota

Ver las indicaciones del fabricante de la máquina.

2.9.9.2 Sucesiones de elementos de contorno: una recta

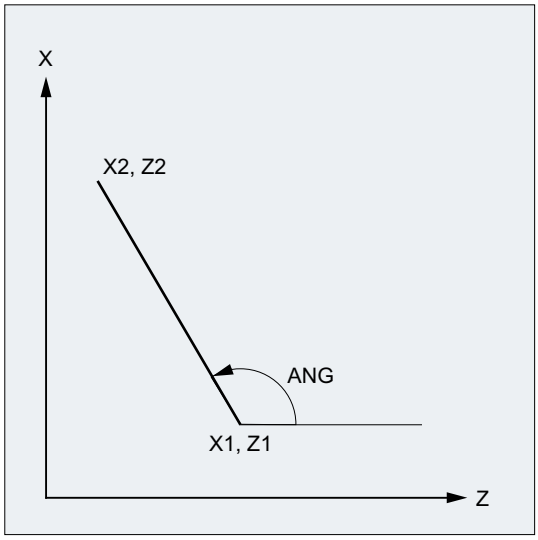
Nota

En la siguiente descripción se parte de la base de que:

- G18 está activo (⇒ el plano de trabajo activo es el plano Z/X).
  (Sin embargo, la programación de sucesiones de elementos de contorno también es posible en G17 o G19 sin limitaciones)
- Para el ángulo, el radio y el chaflán se han definido los siguientes identificadores:
  - ANG (ángulo)
  - RND (radio)
  - CHR (chaflán)

El punto final de las líneas rectas se define indicando:

- Ángulo ANG
- **Una** coordenada cartesiana del punto final (X2 o Z2)



ANG: Ángulo de las líneas rectas
X1, Z1: Coordenadas iniciales
X2, Z2: Coordenadas del punto final de las líneas rectas

Sintaxis

```
X... ANG=...
Z... ANG=...
```

Descripción

`X...`:	Coordenadas del punto final en dirección X
`Z...`:	Coordenadas del punto final en dirección Z
`ANG`:	Identificador para la programación de ángulos El valor indicado (ángulo) se refiere a la abscisa del plano de trabajo activo (eje Z en `G18`).

Ejemplo

Código de programa	**Comentarios**
`N10 X5 Z70 F1000 G18`	`; Desplazamiento a la posición inicial`
`N20 X88.8 ANG=110`	`; Recta con especificación de ángulo`
`N30 ...`	

O bien:

Código de programa	**Comentarios**
`N10 X5 Z70 F1000 G18`	`; Desplazamiento a la posición inicial`
`N20 Z39.5 ANG=110`	`; Recta con especificación de ángulo`
`N30 ...`	

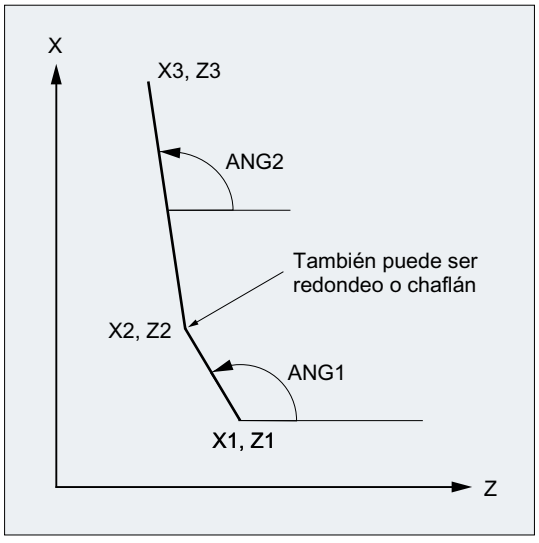
2.9.9.3 Sucesiones de elementos de contorno: dos rectas

Nota

En la siguiente descripción se parte de la base de que:

- G18 está activo (⇒ el plano de trabajo activo es el plano Z/X).
  (Sin embargo, la programación de sucesiones de elementos de contorno también es posible en G17 o G19 sin limitaciones)
- Para el ángulo, el radio y el chaflán se han definido los siguientes identificadores:
  – ANG (ángulo)
  – RND (radio)
  – CHR (chaflán)

El punto final de la primera línea recta puede programarse indicando las coordenadas cartesianas o especificando los ángulos de ambas líneas rectas. El punto final de la segunda línea recta debe programarse siempre en coordenadas cartesianas. El punto de intersección de las dos líneas rectas se puede ejecutar como vértice, redondeo o chaflán.



ANG1:	Ángulo de la primera línea recta
ANG2:	Ángulo de la segunda línea recta
X1, Z1:	Coordenadas iniciales de la primera línea recta
X2, Z2:	Coordenadas del punto final de la primera línea recta o coordenadas iniciales de la segunda línea recta
X3, Z3:	Coordenadas del punto final de la segunda línea recta

Sintaxis

Programación del punto final de la primera línea recta indicando los ángulos

- Vértice como transición entre dos líneas rectas:

```
ANG=…
X… Z… ANG=…
```

- Redondeo como transición entre dos líneas rectas:

```
ANG=… RND=...
X… Z… ANG=…
```

- Chaflán como transición entre dos líneas rectas:

```
ANG=… CHR=...
X… Z… ANG=…
```

Programación del punto final de la primera línea recta indicando las coordenadas

- Vértice como transición entre dos líneas rectas:

```
X… Z…
X… Z…
```

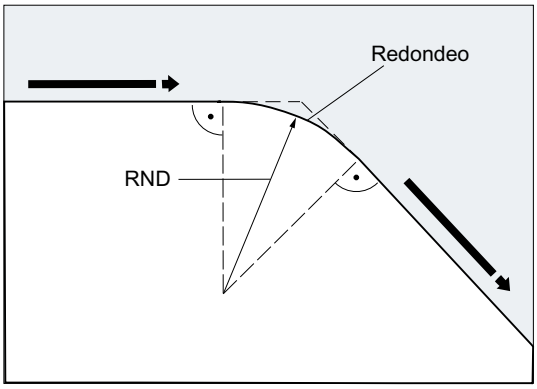
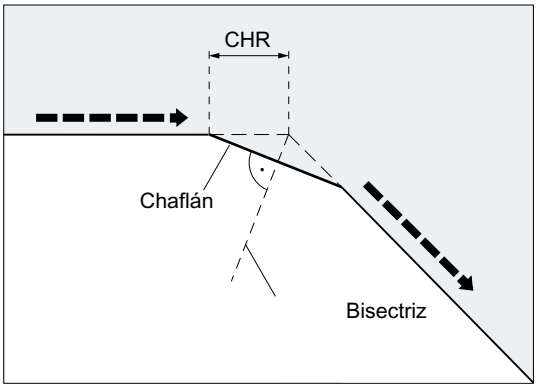
- Redondeo como transición entre dos líneas rectas:

```
X… Z… RND=...
X… Z…
```

- Chaflán como transición entre dos líneas rectas:

```
X… Z… CHR=...
X… Z…
```

Descripción

`ANG=...`:	Identificador para la programación de ángulos El valor indicado (ángulo) se refiere a la abscisa del plano de trabajo activo (eje Z en `G18`).
`RND=...`:	Identificador para la programación de un redondeo El valor indicado corresponde al radio del redondeo: 
`CHR=...`:	Identificador para la programación de un chaflán El valor indicado corresponde a la anchura del chaflán en el sentido de desplazamiento: 
`X...`:	Coordenadas en dirección X
`Z...`:	Coordenadas en dirección Z

Nota

Para más información sobre la programación de un chaflán o de un redondeo, ver "Chaflán, redondeo (CHF, CHR, RND, RNDM, FRC, FRCM) (Página 251)".

Ejemplo

Código de programa	Comentarios
`N10 X10 Z80 F1000 G18`	`; Desplazamiento a la posición inicial.`
`N20 ANG=148.65 CHR=5.5`	`: Línea recta con especificación de ángulo y chaflán.`
`N30 X85 Z40 ANG=100`	`; Línea recta con especificación de ángulo y punto final.`
`N40 ...`	

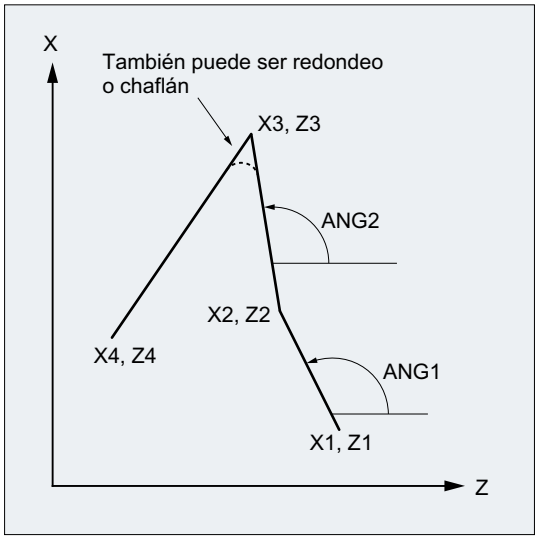
2.9.9.4 Sucesiones de elementos de contorno: tres rectas

Nota

En la siguiente descripción se parte de la base de que:

- G18 está activo (⇒ el plano de trabajo activo es el plano Z/X).
  (Sin embargo, la programación de sucesiones de elementos de contorno también es posible en G17 o G19 sin limitaciones)
- Para el ángulo, el radio y el chaflán se han definido los siguientes identificadores:
  – ANG (ángulo)
  – RND (radio)
  – CHR (chaflán)

El punto final de la primera línea recta puede programarse indicando las coordenadas cartesianas o especificando los ángulos de ambas líneas rectas. El punto final de la segunda y de la tercera línea recta debe programarse siempre en coordenadas cartesianas. El punto de intersección de las líneas rectas se puede ejecutar como vértice, redondeo o chaflán.



ANG1:	Ángulo de la primera línea recta
ANG2:	Ángulo de la segunda línea recta
X1, Z1:	Coordenadas iniciales de la primera línea recta
X2, Z2:	Coordenadas del punto final de la primera línea recta o coordenadas iniciales de la segunda línea recta
X3, Z3:	Coordenadas del punto final de la segunda línea recta o coordenadas iniciales de la tercera línea recta
X4, Z4:	Coordenadas del punto final de la tercera línea recta

Nota

La programación explicada aquí para una sucesión de elementos de contorno de 3 puntos puede continuarse como se desee para sucesiones de elementos de contorno con más de tres puntos.

Sintaxis

Programación del punto final de la primera línea recta indicando los ángulos

- Vértice como transición entre dos líneas rectas:

```
ANG=…
X… Z… ANG=…
X… Z…
```

- Redondeo como transición entre dos líneas rectas:

```
ANG=… RND=...
X… Z… ANG=… RND=...
X… Z…
```

- Chaflán como transición entre dos líneas rectas:

```
ANG=… CHR=...
X… Z… ANG=… CHR=...
X… Z…
```

Programación del punto final de la primera línea recta indicando las coordenadas

- Vértice como transición entre dos líneas rectas:

```
X… Z…
X… Z…
X… Z…
```

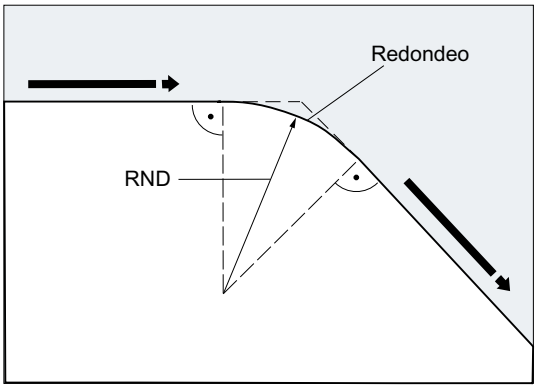
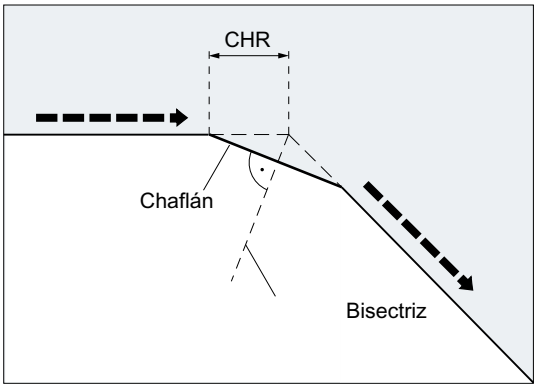
- Redondeo como transición entre dos líneas rectas:

```
X… Z… RND=...
X… Z… RND=...
X… Z…
```

- Chaflán como transición entre dos líneas rectas:

```
X… Z… CHR=...
X… Z… CHR=...
X… Z…
```

Descripción

`ANG=...`:	Identificador para la programación de ángulos El valor indicado (ángulo) se refiere a la abscisa del plano de trabajo activo (eje Z en `G18`).
`RND=...`:	Identificador para la programación de un redondeo El valor indicado corresponde al radio del redondeo: 
`CHR=...`:	Identificador para la programación de un chaflán El valor indicado corresponde a la anchura del chaflán en el sentido de desplazamiento: 
`X...`:	Coordenadas en dirección X
`Z...`:	Coordenadas en dirección Z

Nota

Para más información sobre la programación de un chaflán o de un redondeo, ver "Chaflán, redondeo (CHF, CHR, RND, RNDM, FRC, FRCM) (Página 251)".

Ejemplo

Código de programa	Comentarios
`N10 X10 Z100 F1000 G18`	`; Desplazamiento a la posición inicial`
`N20 ANG=140 CHR=7.5`	`; Recta con especificación de ángulo y chaflán`
`N30 X80 Z70 ANG=95.824 RND=10`	`; Recta en punto intermedio con especificación de ángulo y redondeo`
`N40 X70 Z50`	`; Recta al punto final`

2.9.9.5 Sucesiones de elementos de contorno: Programación de punto final con ángulo

Función

Si aparece en una secuencia de CN el carácter de dirección A, no deben estar programados adicionalmente ninguno, uno o ambos ejes del plano activo.

Número de ejes programados

- Si **ningún eje** del plano activo está programado, se trata de la primera o la segunda secuencia de una sucesión de elementos de contorno compuesta por dos secuencias. Si se trata de la segunda secuencia de tal sucesión de elementos de contorno, significa que los puntos inicial y final en el plano activo son idénticos. La sucesión de elementos de contorno consta entonces, en su caso, de un movimiento vertical al plano activo.
- Si está programado **exactamente un eje** del plano activo, se trata de una línea recta individual cuyo punto final queda determinado unívocamente por el ángulo y la coordenada cartesiana programada, o bien de la segunda secuencia de una sucesión de elementos de contorno compuesta por dos secuencias. En el segundo caso, la coordenada faltante se define igual a la última posición (modal) alcanzada.
- Si están programados **dos ejes** del plano activo, se trata de la segunda secuencia de una sucesión de elementos de contorno compuesta por dos secuencias. Si la secuencia actual no está precedida por una secuencia con programación de ángulos y sin ejes programados del plano activo, esta secuencia no se admite.

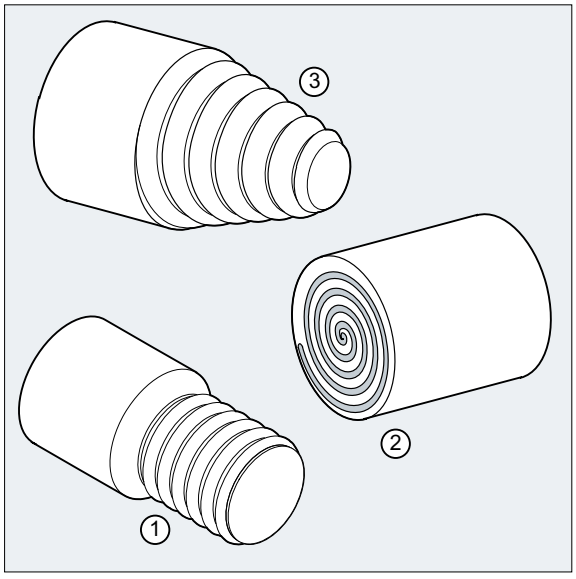
El ángulo A sólo se debe programar en la interpolación lineal o spline.

2.9.10 Tallado de roscas

2.9.10.1 Tallado de roscas con paso constante (G33, SF)

`G33` permite mecanizar roscas con un paso constante:

- Rosca cilíndrica ①
- Rosca transversal ②
- Rosca cónica ③

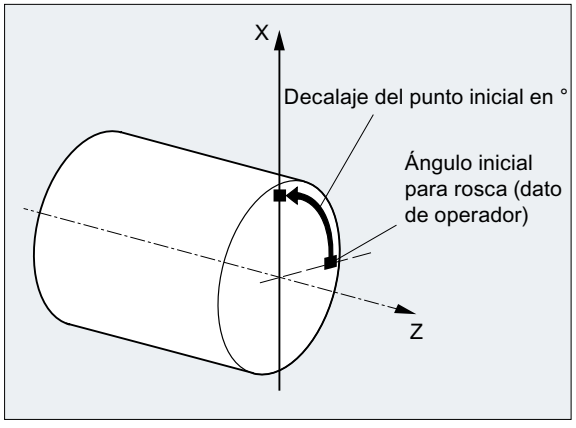


Nota

Para el tallado de roscas con `G33` se necesita un cabezal regulado por velocidad y sistema de medición de trayecto.

Rosca de varios hilos

Las roscas de varios hilos (roscas con decalajes entre los cortes) pueden mecanizarse especificando un decalaje del punto inicial. La programación se realiza en la secuencia `G33` bajo la dirección `SF`.

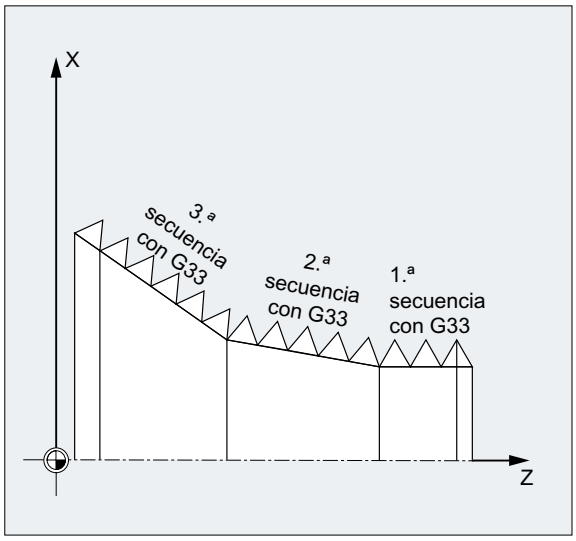


Nota

Si no se introduce un decalaje para el punto inicial, se toma el valor introducido en los datos de operador "ángulo inicial para roscados".

Cadena de roscas

Puede mecanizarse una cadena de roscas mediante varias secuencias `G33` programadas sucesivamente:



Nota

Con el modo de contorneado `G64` se concatenan las secuencias mediante control anticipativo de la velocidad de forma que no se produzcan saltos en ésta.

Sentido de giro de la rosca

El sentido de giro de la rosca viene determinado por el sentido de giro del cabezal:

- El sentido de giro a derechas con `M3` crea una rosca a derechas
- El sentido de giro a izquierdas con `M4` crea una rosca a izquierdas

Sintaxis

Rosca cilíndrica:

```
G33 Z… K…
G33 Z… K… SF=…
```

Rosca transversal:

```
G33 X… I…
G33 X… I… SF=…
```

Rosca cónica:

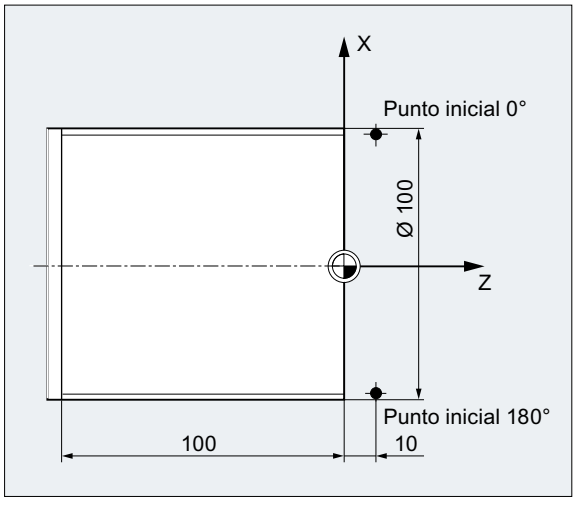
```
G33 X… Z… K…
G33 X… Z… K… SF=…
G33 X… Z… I…
G33 X… Z… I… SF=…
```

Descripción

G33:	Comando para el tallado de roscas con paso constante	
X... Y... Z...:	Puntos finales en coordenadas cartesianas	
I...:	Paso de rosca en dirección X	
J...:	Paso de rosca en dirección Y	
K...:	Paso de rosca en dirección Z	
Z:	Eje longitudinal	
X:	Eje de refrentado	
Z... K...:	Longitud y paso de rosca para rosca cilíndrica	
X... I...:	Diámetro y paso de la rosca para rosca transversal	
I... o K...:	Paso de rosca para rosca cónica La especificación (I... o K...) se ajusta al ángulo en cono:	
	< 45°:	El paso de rosca se indica con K... (paso de rosca en sentido longitudinal).
	> 45°:	El paso de rosca se indica con I... (paso de rosca en sentido transversal).
	= 45°:	El paso de rosca puede indicarse con I... o K....
SF=...:	Decalaje del punto inicial (sólo necesario en roscas de varios hilos) El decalaje del punto inicial se indica como posición angular absoluta.	
	Rango de valores:	0.0000 a 359.999 grados

Ejemplos

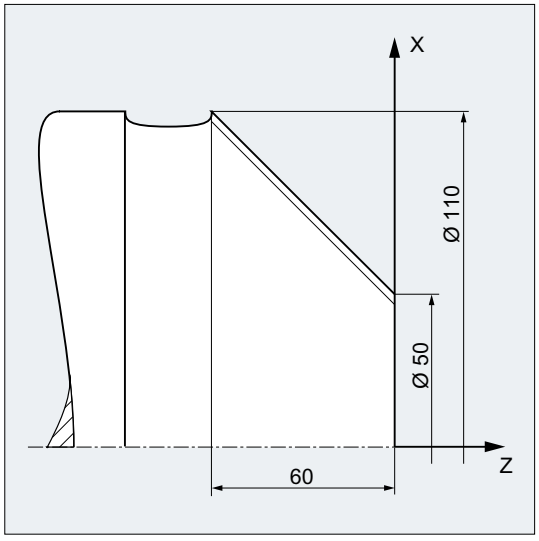
Ejemplo 1: rosca cilíndrica de dos filetes con decalaje del punto inicial de 180°



Código de programa	Comentarios
N10 G1 G54 X99 Z10 S500 F100 M3	; Decalaje de origen, desplazamiento al punto inicial, activación del cabezal.
N20 G33 Z-100 K4	; Rosca cilíndrica: punto final en Z

Código de programa	Comentarios
N30 G0 X102	; Retirada a la posición inicial.
N40 G0 Z10	
N50 G1 X99	
N60 G33 Z-100 K4 SF=180	; 2. Sección: decalaje del punto inicial 180°
N70 G0 X110	; Retirada de la herramienta.
N80 G0 Z10	
N90 M30	; Fin del programa.

Ejemplo 2: rosca cónica con un ángulo menor de 45°

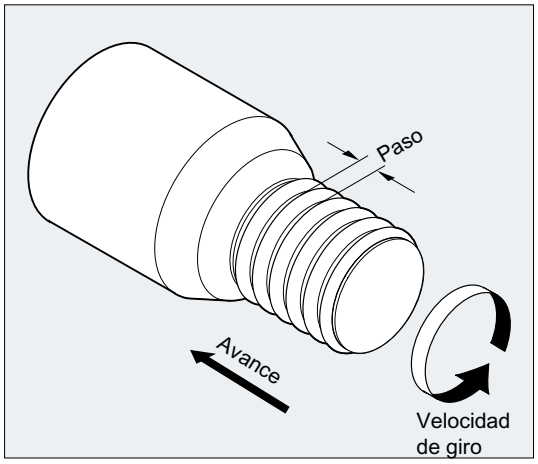


Código de programa	Comentarios
N10 G1 X50 Z0 S500 F100 M3	; Desplazamiento al punto inicial, activación del cabezal.
N20 G33 X110 Z-60 K4	; Rosca cónica: punto final en X y Z, indicación del paso de rosca con K... en dirección Z (ya que el ángulo en cono < 45°).
N30 G0 Z0 M30	; Retirar, fin de programa.

Información adicional

Avance en el tallado de roscas con G33

A partir de la velocidad programada del cabezal y del paso de rosca deseado el control calcula el avance necesario con el que la cuchilla se desplaza a lo largo de la longitud de la rosca en sentido longitudinal y/o transversal. El avance `F` no se tiene en cuenta en `G33`; el control vigila que no se sobrepase la velocidad de eje máxima (rápido).



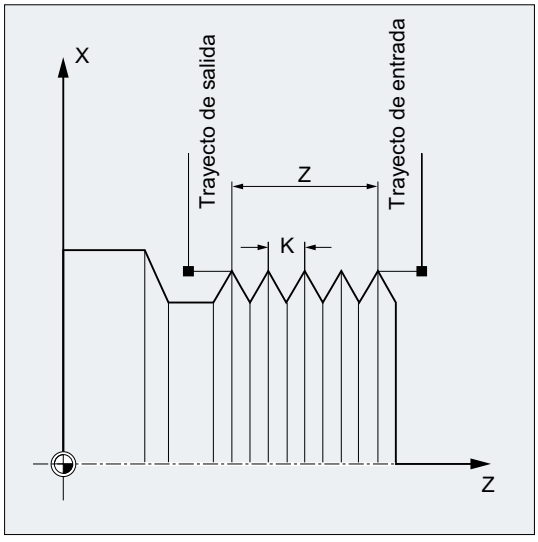
Rosca cilíndrica

La rosca cilíndrica se describe a través de:

- Longitud de la rosca
- Paso de rosca

La longitud de la rosca se indica con una de las coordenadas cartesianas X, Y o Z o en acotado absoluto o incremental (en tornos se prefiere la dirección Z). Adicionalmente se deben tener en cuenta los trayectos de entrada y salida para permitir la aceleración y frenado del eje de avance.

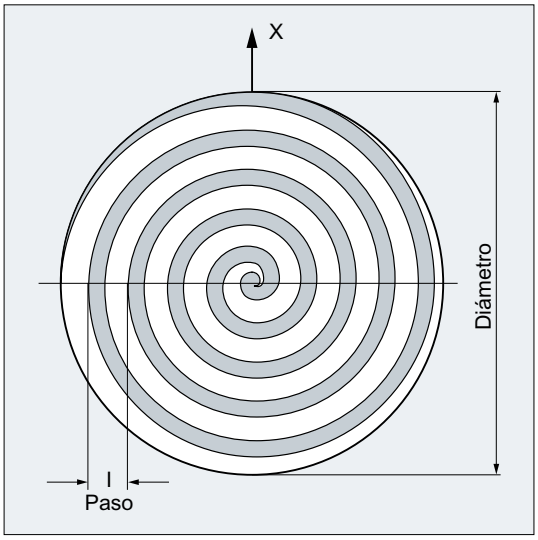
El paso de rosca se introduce en las direcciones `I`, `J`, `K` (en tornos, preferentemente con `K`).



Rosca transversal

La rosca transversal se describe a través de:

- Diámetro de la rosca (preferentemente en dirección X)
- Paso de rosca (preferentemente con `I`)



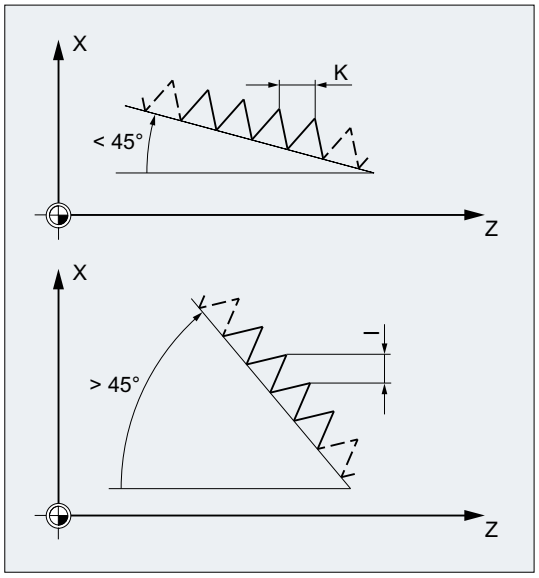
Rosca cónica

La rosca cónica se describe a través de:

- Punto final en dirección longitudinal y transversal (perfil del cono)
- Paso de rosca

El perfil del cono se introduce en coordenadas cartesianas X, Y, Z en cotas absolutas o incrementales; en el mecanizado en tornos se rosca preferentemente a lo largo de las direcciones X y Z. Adicionalmente se deben tener en cuenta los trayectos de entrada y salida para permitir la aceleración y frenado del eje de avance.

La indicación del paso de rosca se ajusta al ángulo en cono (ángulo entre el eje longitudinal y la superficie cónica):



2.9.10.2 Tallado de roscas con paso creciente o decreciente (G34, G35)

Los comandos G34 y G35 han permitido ampliar la funcionalidad de G33 con la posibilidad de programar adicionalmente en la dirección F una modificación del paso de rosca. En el caso de G34, esto produce un crecimiento lineal del paso de rosca; en el caso de G35, causa un decrecimiento lineal del paso de rosca. Los comandos G34 y G35 pueden por ello utilizarse para el mecanizado de roscas autorroscantes.

Sintaxis

Rosca cilíndrica con paso creciente:
```
G34 Z… K… F...
```

Rosca cilíndrica con paso decreciente:
```
G35 Z… K… F...
```

Rosca transversal con paso creciente:
```
G34 X… I… F...
```

Rosca transversal con paso decreciente:
```
G35 X… I… F...
```

Rosca cónica con paso creciente:
```
G34 X… Z… K… F...
G34 X… Z… I… F...
```

Rosca cónica con paso decreciente:
```
G35 X… Z… K… F...
G35 X… Z… I… F...
```

Descripción

`G34`:	Comando para el tallado de roscas con paso linealmente **creciente**	
`G35`:	Comando para el tallado de roscas con paso linealmente **decreciente**	
`X... Y... Z...`:	Puntos finales en coordenadas cartesianas	
`I...`:	Paso de rosca en dirección X	
`J...`:	Paso de rosca en dirección Y	
`K...`:	Paso de rosca en dirección Z	
`F...`:	Cambio del paso de rosca Si se conocen el paso inicial y el paso final de una rosca, el cambio del paso de rosca a programar se puede calcular según la siguiente ecuación: $F = \frac{k_e^2 - k_a^2}{2 * l_G}$ [mm/vuelta²] Significado:	
	k_e:	Paso final de la rosca (paso de rosca de la coordenada del punto de destino del eje) [mm/vuelta]
	k_a:	Paso inicial de la rosca (programado en I, J o K) [mm/vuelta]
	l_G:	Longitud de la rosca [mm]

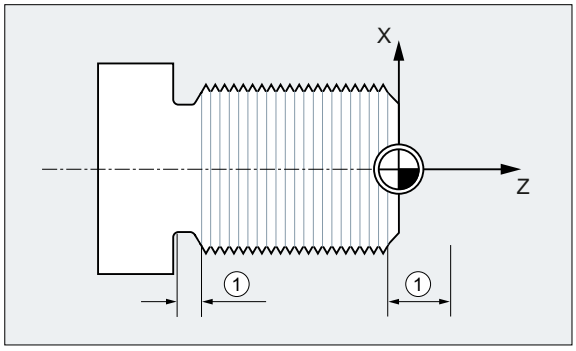
Ejemplo

Código de programa	Comentarios
N1608 M3 S10	; Activación de cabezal.
N1609 G0 G64 Z40 X216	; Desplazamiento al punto inicial.
N1610 G33 Z0 K100 SF=R14	; Tallado de roscas con paso constante (100 mm/vuelta)
N1611 G35 Z-200 K100 F17.045455	; Reducción del paso: 17.0454 mm/vuelta2
	Paso al final de la secuencia: 50 mm/vuelta
N1612 G33 Z-240 K50	; Ejecutar secuencia de roscado sin jerk.
N1613 G0 X218	
N1614 G0 Z40	
N1615 M17	

2.9.10.3 Trayectos de entrada y salida programados en G33, G34 y G35 (DITS, DITE)

Con las direcciones DITS y DITE puede especificarse en el programa de pieza el trayecto de entrada y salida de la rosca.

El eje de roscado se acelera o se frena dentro del trayecto especificado.



① Trayecto de entrada o de salida según sentido de mecanizado

Trayecto de entrada corto

Debido al collar del eje a la entrada del roscado hay poco espacio para la rampa de arranque de la herramienta. Por tanto,
esta debe definirse más corta a través de DITS.

Trayecto de salida corto

Debido al collar del eje a la salida del roscado se dispone de poco espacio para la rampa de frenado de la herramienta, por lo cual existe peligro de colisión entre la pieza y el filo. La rampa de frenado se puede definir más corta a través de DITE. Sin embargo, se puede producir una colisión debido a la inercia de los componentes mecánicos.

Remedio: programar la rosca más corta; reducir la velocidad de giro del cabezal.

Nota

DITE actúa al final de la rosca como una separación de redondeo o matado al final de la rosca. Con ello se consigue modificar el desplazamiento del eje sin sacudidas.

Efectos

El trayecto de entrada y salida programado actúa en la trayectoria solamente para acelerar la velocidad. Si uno de los dos trayectos se define con un valor mayor que el necesario para el eje de roscado con aceleración activa, este se acelera con aceleración máxima o se frena.

Sintaxis

```
DITS=<Valor> DITE=<Valor>
```

Descripción

DITS:	Definir el trayecto de entrada para la rosca
DITE:	Definir el trayecto de salida para la rosca
<Valor>:	¡Con DITS y DITE se programan solo trayectos, no posiciones! El trayecto de entrada o salida programado se interpreta según el ajuste de medida activo en ese momento (en pulgadas o métrico).

Ejemplo

Código de programa	**Comentarios**
...	
N40 G90 G0 Z100 X10 SOFT M3 S500	
N50 G33 Z50 K5 SF=180 DITS=1 DITE=3	; Comienzo del matado de esquinas con Z = 53
N60 G0 X20	

Información adicional

SD42010 $SC_THREAD_RAMP_DISP

Al introducir una secuencia con DITS o DITE en el proceso principal, se aplica el trayecto de entrada o salida programado en el dato de operador SD42010 $SC_THREAD_RAMP_DISP:

- SD42010 $SC_THREAD_RAMP_DISP[**0**] = valor programado de DITS
- SD42010 $SC_THREAD_RAMP_DISP[**1**] = valor programado de DITE

Si no se ha programado ningún trayecto de entrada/salida antes o durante la primera secuencia de roscado, se utiliza el valor actual del dato de operador correspondiente.

Comportamiento después de un reset al final de programa/canal/GMO

Los valores de SD42010 sobrescritos con DITS o DITE permanecen activos después de un reset al final de programa/GMO/canal.

Comportamiento tras arranque en caliente

Cuando se ha realizado un arranque en caliente, el dato de operador se restablece a los valores que estaban activos (comportamiento estándar) antes de sobrescribirlos con DITS o DITE.

Sin embargo, si los valores programados con DITS y DITE deben seguir activos tras un arranque en caliente, se debe indicar el dato de operador SD42010 $SC_THREAD_RAMP_DISP en el dato de máquina MD10710 $MN_PROG_SD_RESET_SAVE_TAB:

MD10710 $MN_PROG_SD_RESET_SAVE_TAB[<n>] = 42010

Comportamiento con trayectos de entrada/salida muy cortos

Con trayectos de entrada/salida muy cortos, el eje de roscado se somete a una aceleración mayor que la dimensionada en la configuración. El eje se sobrecarga debido a la aceleración.

Para la entrada de rosca se visualizará entonces la alarma 22280 "Trayecto de entrada programado muy corto" (si así se ha configurado en el DM11411 $MN_ENABLE_ALARM_MASK). La alarma es meramente informativa y no afecta a la ejecución del programa de pieza.

2.9.10.4 Retirada rápida durante el tallado de roscas (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN)

La función "Retirada rápida durante tallado de roscas (G33)" permite una interrupción durante el tallado de rosca sin dañar el contorno en las siguientes situaciones:

- Parada CN disparada con DB21...DBX7.3
- Parada CN disparada implícitamente por medio de alarma
- Conexión de una entrada rápida (Página 560)

El movimiento de retroceso puede programarse a través de:

- Distancia y dirección de la retirada (relativas)
- Posición de retirada (absoluta)

Nota

Señales de Parada CN

Las siguientes señales de Parada CN no provocan la retirada rápida durante el tallado de roscas:

- DB21, ... DBX3.4 (ejes de Parada CN más cabezales)
- DB21, ... DBX7.2 (Parada CN en límite de secuencia)

Roscado

La función "retirada rápida" **no** puede aplicarse al roscado de **taladros** (G331/G332).

Sintaxis

Habilitar retirada rápida, movimiento de retroceso a través de la distancia y dirección de la retirada:
```
G33 ... LFON DILF=<Valor> LFTXT/LFWP ALF=<Valor>
```

Habilitar retirada rápida, movimiento de retroceso a través de la posición de retirada:

```
POLF[<Identificador de eje>]=<Valor> LFPOS
POLFMASK/POLFMLIN(<Nombre de eje1>,<Nombre de eje2>,...)
G33 ... LFON
```

Bloquear retirada rápida durante tallado de roscas:
```
LFOF
```

Descripción

LFON:	Habilitar retirada rápida durante tallado de roscas (G33)	
LFOF:	Bloquear retirada rápida durante tallado de roscas (G33)	
DILF= :	Determinar longitud de la distancia de retirada	
	El valor preajustado en la configuración del DM (MD21200 $MC_LIFTFAST_DIST) se puede modificar en el programa de pieza programando DILF. **Nota:** Tras RESET CN está siempre activo el valor de DM configurado.	
LFTXT LFWP:	La dirección de retirada se controla en combinación con ALF con los comandos G LFTXT y LFWP.	
	LFTXT:	El plano en el que se ejecuta el movimiento de retirada se calcula a partir de la tangente de trayectoria y de la dirección de herramienta (ajuste por defecto).
	LFWP:	El plano en el que se ejecuta el movimiento de retirada es el plano de trabajo activo.
ALF= :	La dirección se programa en el plano de retirada con ALF en pasos discretos de grados.	
	Con LFTXT la retirada se define en la dirección de la herramienta para ALF=1. Con LFWP, la dirección en el plano de mecanizado se asigna de la siguiente manera: • G17 (plano X/Y) ALF=1 ; retirada en dirección X ALF=3 ; retirada en dirección Y • G18 (plano Z/X) ALF=1 ; retirada en dirección Z ALF=3 ; retirada en dirección X • G19 (plano Y/Z) ALF=1 ; retirada en dirección Y ALF=3 ; retirada en dirección Z	
	Encontrará **más información** sobre las posibilidades de programación con ALF en el capítulo "Sentido de desplazamiento en la retirada rápida del contorno (Página 562)".	
LFPOS:	Retirada del eje declarado con POLFMASK o POLFMLIN a la posición de eje absoluta programada con POLF.	
POLFMASK:	Habilitación de los ejes (<Nombre de eje1>,<Nombre de eje1>,...) para la retirada independiente a la posición absoluta	

POLFMLIN:	Habilitación de los ejes para la retirada a la posición absoluta en el contexto lineal **Nota:** según el comportamiento dinámico de todos los ejes afectados, el contexto lineal no siempre se puede establecer hasta alcanzar la posición de retirada.	
POLF[]:	Definición de la posición de retirada absoluta para el eje geométrico o el eje de máquina indicados en el índice.	
	Efecto:	modal
	=<Valor>:	El valor asignado se interpreta, en el caso de ejes geométricos, como posición en el sistema de coordenadas de pieza (WKS); en el caso de ejes de máquina, como posición en el sistema de coordenadas de máquina (MKS). La asignación de valores también es programable como acotado incremental: =IC<Valor>
<Identificador de eje>:	Identificador de un eje geométrico o de máquina	

Nota

LFON y LFOF pueden programarse siempre; no obstante, la evaluación se realiza solamente en el tallado de roscas (G33).

Nota

POLF con POLFMASK/POLFMLIN no están limitados al uso en el tallado de roscas.

Ejemplos

Ejemplo 1: habilitar la retirada rápida durante el tallado de roscas

Código de programa	Comentarios
N55 M3 S500 G90 G18	; Plano de mecanizado activo
...	; Desplazamiento a la posición inicial
N65 MSG ("Roscado")	; Cota de penetración de la herramienta
MM_THREAD:	
N67 $AC_LIFTFAST=0	; Resetear antes de iniciar el roscado.
N68 G0 Z5	
N68 X10	
N70 G33 Z30 K5 LFON DILF=10 LFWP ALF=7	; Habilitar la retirada rápida durante el tallado de roscas. Distancia de retirada=10 mm Plano de retirada: Z/X (debido a G18) Dirección de retirada: -X (con ALF=3: dirección de retirada +X)
N71 G33 Z55 X15	
N72 G1	; Cancelar tallado de roscas.
N69 IF $AC_LIFTFAST GOTOB MM_THREAD	; Cuando se haya interrumpido el tallado de roscas.

Código de programa	Comentarios
N90 MSG("")	
...	
N70 M30	

Ejemplo 2: Desactivación de la retirada rápida antes del roscado

Código de programa	Comentarios
N55 M3 S500 G90 G0 X0 Z0	
...	
N87 MSG ("Roscado")	
N88 LFOF	; Desactivar retirada rápida antes del roscado.
N89 CYCLE...	; Ciclo de roscado con G33.
N90 MSG("")	
...	
N99 M30	

Ejemplo 3: Retirada rápida a la posición de retirada absoluta

Con una parada se suprime la interpolación de trayectoria de X y se interpola en su lugar un desplazamiento con velocidad máxima a la posición POLF [X]. El desplazamiento de los demás ejes sigue siendo determinado por el contorno programado o el paso de rosca y la velocidad de giro del cabezal.

Código de programa	Comentarios
N10 G0 G90 X200 Z0 S200 M3	
N20 G0 G90 X170	
N22 POLF[X]=210 LFPOS	
N23 POLFMASK(X)	; Activar (habilitar) la retirada rápida del eje 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()	; Bloquear retirada para todos los ejes.
M30	

2.9.10.5 Roscas convexas (G335, G336)

Los comandos G G335 y G336 ofrecen la posibilidad de tornear roscas convexas (= diferentes de la forma cilíndrica). Esta aplicación se usa para mecanizar piezas de tamaño extremadamente grande que, debido a su propio peso, se pandean en la máquina. Si se usan roscas paralelas al eje, el paso de rosca en la parte central de la pieza resulta insuficiente. Las roscas convexas compensan este efecto.

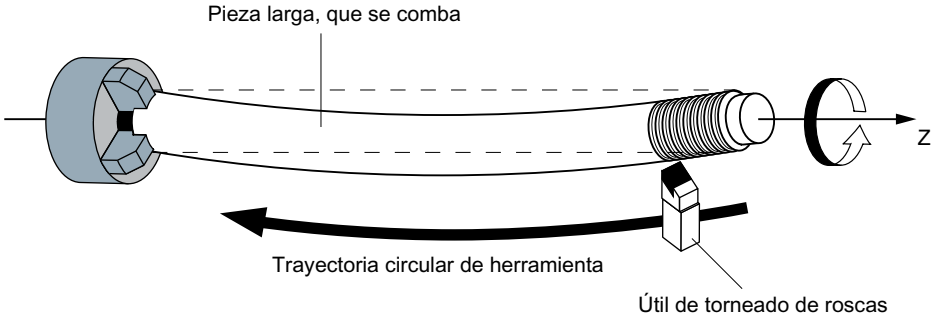


Figura 2-6 Torneado de una rosca convexa

Programación

El torneado de roscas convexas se programa con `G335` o `G336`:

`G335`:	Torneado de una rosca convexa en una trayectoria de herramienta circular orientada **en sentido horario**
`G336`:	Torneado de una rosca convexa en una trayectoria de herramienta circular orientada **en sentido antihorario**

En principio, la programación se realiza del mismo modo que para una rosca lineal, es decir, especificando los puntos finales de secuencia por eje y el paso mediante los parámetros `I`, `J` y `K` (ver "Tallado de roscas con paso constante (G33, SF) (Página 226)").

Además debe indicarse un arco de circunferencia. Al igual que con `G2` o `G3`, dicho arco puede programarse especificando el centro, el radio, el ángulo en el vértice o los puntos intermedios (ver "Interpolación circular (Página 194)"). Si se programa la rosca convexa especificando el centro, debe tenerse en cuenta lo siguiente: Dado que `I`, `J` y `K` se utilizan en el tallado de roscas para indicar el paso de rosca, al programar especificando el centro, los parámetros de la circunferencia deberán programarse con `IR=...`, `JR=...` y `KR=...`.

`IR=...`:	Coordenada cartesiana del centro de la circunferencia en dirección del eje X
`JR=...`:	Coordenada cartesiana del centro de la circunferencia en dirección del eje Y
`KR=...`:	Coordenada cartesiana del centro de la circunferencia en dirección del eje Z

Nota

IR, JR y KR son los valores estándar ajustables mediante dato de máquina (DM10651 $MN_IPO_PARAM_THREAD_NAME_TAB) de los parámetros de interpolación para roscas convexas.

Para conocer las diferencias respecto a estos valores estándar, consulte los datos del fabricante de la máquina.

Opcionalmente puede indicarse además un decalaje del punto inicial `SF` (ver "Tallado de roscas con paso constante (G33, SF) (Página 226)").

Sintaxis

Por lo tanto, la sintaxis para la programación de una rosca convexa tiene la siguiente forma general:

```
G335/G336 <coordenada(s) del punto de destino del eje> <paso> <arco de circunferencia> [<decalaje del punto inicial>]
```

Ejemplos

Ejemplo 1: Rosca convexa en sentido horario con programación de punto final y centro

Código de programa	Comentarios
`N5 G0 G18 X50 Z50`	`; Desplazamiento a la posición inicial.`
`N10 G335 Z100 K=3.5 KR=25 IR=-20 SF=90`	`; Tornear rosca convexa en sentido horario.`

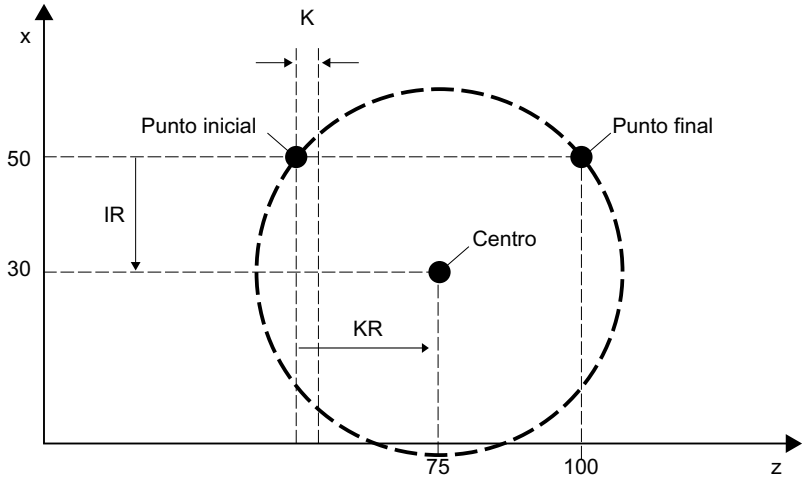


Figura 2-7 Rosca convexa en sentido horario con programación de punto final y centro

Ejemplo 2: Rosca convexa en sentido antihorario con programación de punto final y centro

Código de programa	**Comentarios**
N5 G0 G18 X50 Z50	; Desplazamiento a la posición inicial.
N10 **G336 Z100** K=3.5 **KR=25 IR=20** SF=90	; Tornear rosca convexa en sentido antihorario.

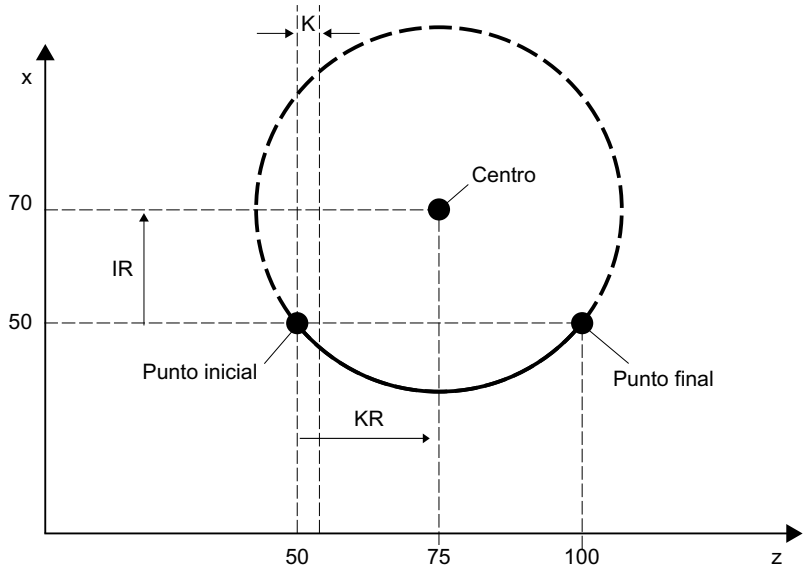


Figura 2-8 Rosca convexa en sentido antihorario con programación de punto final y centro

Ejemplo 3: Rosca convexa en sentido horario con programación de punto final y radio

Código de programa
N5 G0 G18 X50 Z50
N10 **G335 Z100** K=3.5 **CR=32** SF=90

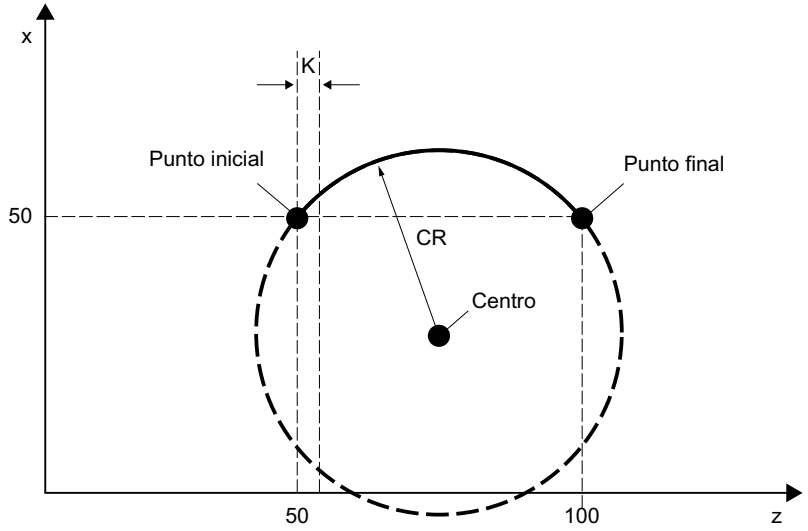


Figura 2-9 Rosca convexa en sentido horario con programación de punto final y radio

Ejemplo 4: Rosca convexa en sentido horario con programación de punto final y ángulo en el vértice

Código de programa

```
N5 G0 G18 X50 Z50
N10 G335 Z100 K=3.5 AR=102.75 SF=90
```

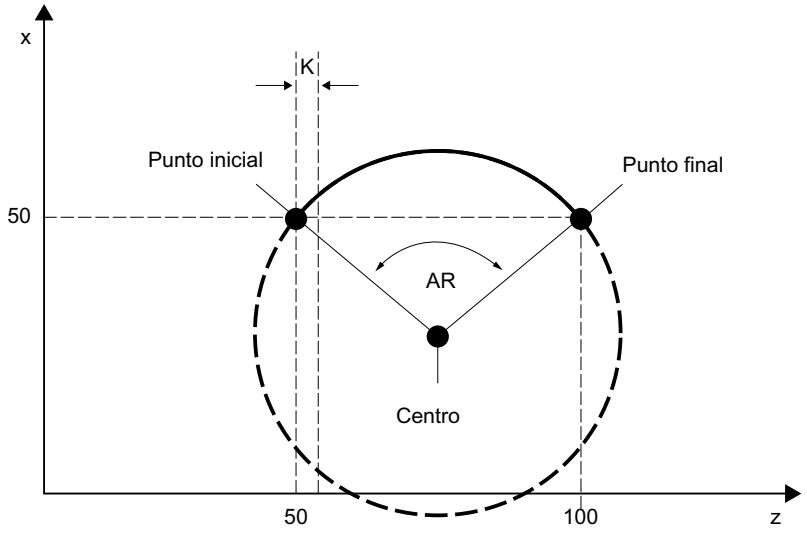


Figura 2-10 Rosca convexa en sentido horario con programación de punto final y ángulo en el vértice

Ejemplo 5: Rosca convexa en sentido horario con programación de centro y ángulo en el vértice

Código de programa

```
N5 G0 G18 X50 Z50
N10 G335 K=3.5 KR=25 IR=-20 AR=102.75 SF=90
```

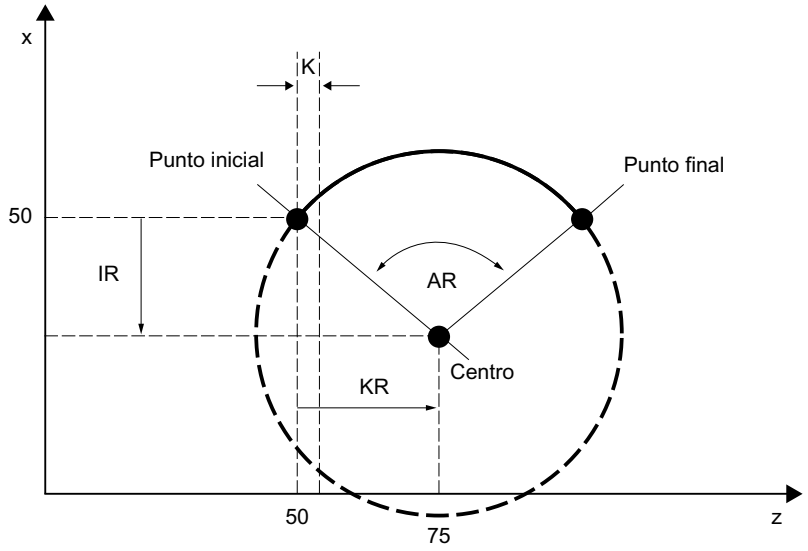


Figura 2-11 Rosca convexa en sentido horario con programación de centro y ángulo en el vértice

Ejemplo 6: Rosca convexa en sentido horario con programación de punto final y punto intermedio

Código de programa

```
N5 G0 G18 X50 Z50
N10 G335 Z100 K=3.5 I1=60 K1=64
```

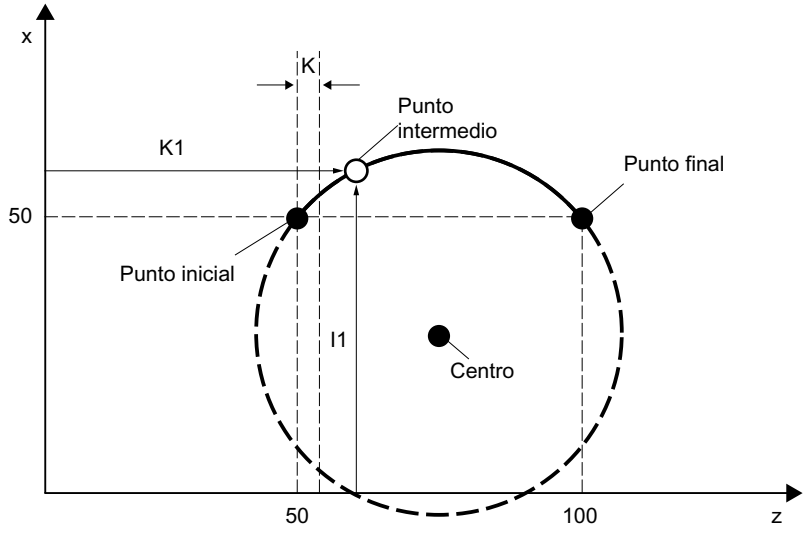
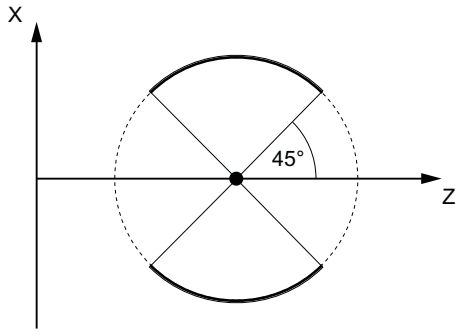


Figura 2-12 Rosca convexa en sentido horario con programación de punto final y punto intermedio

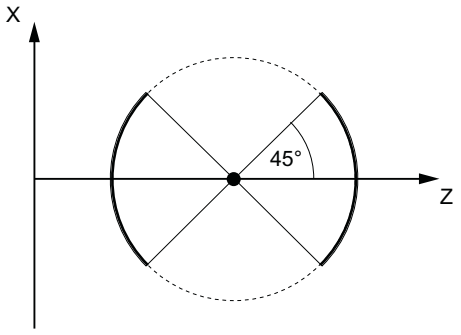
Información adicional

Rangos de arco de circunferencia autorizados

El arco de circunferencia programado en `G335` o `G336` debe encontrarse dentro de un rango en el que el eje principal de rosca especificado (`I`, `J` o `K`) ocupe el componente axial principal a lo largo de todo el arco:



Rangos autorizados para el eje **Z** (paso de rosca programado con `K`)



Rangos autorizados para el eje **X** (paso de rosca programado con `I`)

No se permiten cambios del eje principal de la rosca como el que se muestra en la siguiente figura:

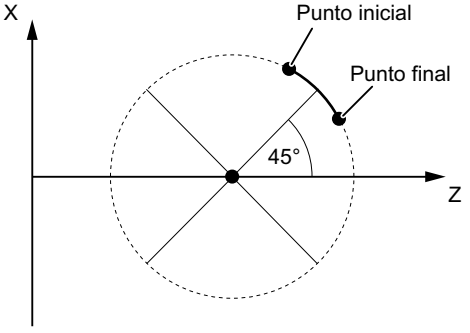


Figura 2-13 Roscas convexas: Rango no permitido

Frames

G335 y G336 también son posibles con frames activos. Sin embargo, es necesario asegurarse de que en el sistema de coordenadas básico (BKS) se respeten los rangos de arco de circunferencia autorizados.

Condiciones para la programación de círculos

A la programación de círculos con G335 o G336 se aplican las condiciones descritas para la programación de círculos con G2 o G3 (ver "Interpolación circular (Página 194)").

2.9.11 Roscado con macho sin mandril de compensación

2.9.11.1 Roscado con macho sin mandril de compensación y movimiento de retirada (G331, G332)

En el roscado con macho sin mandril de compensación se utilizan los comandos `G331` y `G332` para ejecutar los siguientes movimientos de desplazamiento:

- `G331`: roscado en el sentido de taladrado hasta el punto final de rosca
- `G332`: movimiento de retirada para la secuencia de roscado con macho `G331` con inversión automática del sentido de giro del cabezal

Sintaxis

```
G331 <Eje> <Paso de rosca>
G331 <Eje> <Paso de rosca> S...
G332 <Eje> <Paso de rosca>
```

Descripción

`G331`:	Roscado El taladro roscado se describe mediante el movimiento de desplazamiento del eje (profundidad de taladrado) y el paso de rosca.	
	Efecto:	modal
`G332`:	Movimiento de retirada del roscado El movimiento de retirada debe ejecutarse con el mismo paso que el roscado con macho (`G331`). La inversión del sentido de giro del cabezal se realiza de forma automática.	
	Efecto:	modal
`<Eje>`:	Recorrido de desplazamiento/posición del eje geométrico (X, Y o Z) respecto al punto final de rosca, p. ej., Z50	
`<Paso de rosca>`:	Paso de rosca I (X), J (Y) o K (Z): • Paso positivo: rosca a derechas, p. ej., K1,25 • Paso negativo: rosca a izquierdas, p. ej., K-1,25	
	Rango de valores:	±0.001 a ±2000.00 mm/vuelta
`S...`:	Velocidad de giro del cabezal Si no se ha especificado la velocidad de giro del cabezal, el desplazamiento se realiza con la última velocidad de giro activa.	

Nota

Segundo juego de datos de escalón de reducción

Con objeto de conseguir una adaptación efectiva de la velocidad del cabezal y el par motor en el roscado y poder acelerar más rápidamente, en datos de máquina específicos del eje puede estar predefinido -a diferencia del primer bloque de datos de escalón de reducción y también independientemente de estos umbrales de velocidad de giro- un segundo juego de datos de escalón de reducción para dos umbrales de conmutación adicionales configurables (velocidad máxima y velocidad mínima). Debe tenerse en cuenta lo especificado por el fabricante de la máquina.

Encontrará más información en el manual de funciones "Ejes y cabezales".

Ejemplos

- Ejemplo: Roscado con macho con G331/G332 (Página 247)
- Ejemplo: emitir la velocidad de taladrado programada en el escalón de reducción actual (Página 248)
- Ejemplo: aplicación del segundo juego de datos de escalón de reducción (Página 248)
- Ejemplo: sin programación de velocidad de giro, vigilancia del escalón de reducción (Página 249)
- Ejemplo: cambio del escalón de reducción no posible, vigilancia del escalón de reducción (Página 249)
- Ejemplo: programación sin SPOS (Página 249)

2.9.11.2 Ejemplo: Roscado con macho con G331/G332

Código de programa	Comentarios
N10 SPOS[n]=0	; cabezal: modo de regulación de posición
	; posición inicial (0 grados)
N20 G0 X0 Y0 Z2	; ejes: desplazamiento a la posición inicial
N30 G331 Z-50 K-4 S200	; roscado en Z,
	; paso K-4 negativo =>
	; sentido de giro del cabezal: giro a izquierdas,
	; velocidad de giro del cabezal: 200 r/min
N40 G332 Z3 K-4	; movimiento de retirada en Z,
	; paso K-4 negativo (giro a izquierdas),
	; inversión autom. del sentido de giro =>
	; sentido de giro del cabezal a derechas

Código de programa	Comentarios
N50 G1 F1000 X100 Y100 Z100 S300 M3	; cabezal en modo cabezal.

2.9.11.3 Ejemplo: emitir la velocidad de taladrado programada en el escalón de reducción actual

Código de programa	Comentarios
N05 M40 S500	; velocidad de giro del cabezal programada: 500 r/min =>
	; escalón de reducción 1 (20 a 1028 r/min)
...	
N55 SPOS=0	; posicionar cabezal
N60 G331 Z-10 K5 S800	; roscado
	; velocidad de giro del cabezal: 800 r/min => escalón de reducción 1

El escalón de reducción a `M40` adecuado para la velocidad de giro del cabezal `S500` se determina a partir del primer juego de datos de escalón de reducción. La velocidad de taladrado programada `(S800)` se emite en el escalón de reducción actual y está limitada, en su caso, a la velocidad máxima del escalón de reducción. No es posible un cambio automático del escalón de reducción después de `SPOS`. El requisito para el cambio automático del escalón de reducción es que el cabezal esté en el modo de control de velocidad en lazo abierto.

Nota

Si se debe seleccionar el escalón de reducción 2 con una velocidad de giro del cabezal de 800 r/min, los umbrales de conmutación para la velocidad máxima y mínima se tienen que configurar al efecto en los correspondientes datos de máquina del segundo juego de datos de escalón de reducción (ver los siguientes ejemplos).

2.9.11.4 Ejemplo: aplicación del segundo juego de datos de escalón de reducción

Los umbrales de conmutación del segundo juego de datos de escalón de reducción para la velocidad máxima y la velocidad mínima se evalúan con `G331`/`G332` y programación de un valor `S` para el cabezal maestro activo. El cambio automático del escalón de reducción `M40` tiene que estar activo. El escalón de reducción determinado de esta manera se compara con el escalón de reducción activo. Si existe una diferencia entre los dos, se ejecuta un cambio de escalón de reducción.

Código de programa	Comentarios
N05 M40 S500	; velocidad de giro del cabezal programada: 500 r/min
...	
N50 G331 S800	; cabezal maestro: se selecciona el escalón de reducción 2
N55 SPOS=0	; posicionar cabezal
N60 G331 Z-10 K5	; roscado
	; aceleración del cabezal desde juego de datos de escalón de reducción 2

2.9.11.5 Ejemplo: sin programación de velocidad de giro, vigilancia del escalón de reducción

Si no se programa ninguna velocidad de giro con G331 cuando se aplica el segundo juego de datos de escalón de reducción, la rosca se mecaniza con la última velocidad de giro programada. No se efectúa ningún cambio del escalón de reducción. No obstante, en este caso se vigila si la última velocidad de giro programada se encuentra en el rango de velocidad especificado (umbrales de conmutación para la velocidad máxima y mínima) del escalón de reducción activo. De lo contrario se muestra la alarma 16748.

Código de programa	**Comentarios**
N05 M40 S800	; velocidad de giro del cabezal programada: 800 r/min
...	
N55 SPOS=0	; posicionar cabezal
N60 G331 Z-10 K5	; roscado
	; vigilancia de la velocidad del cabezal a 800 r/min
	; el escalón de reducción 1 está activo
	; el escalón de reducción 2 debería estar activo => alarma 16748

2.9.11.6 Ejemplo: cambio del escalón de reducción no posible, vigilancia del escalón de reducción

Si, cuando se aplica el segundo juego de datos de escalón de reducción, en la secuencia G331 se programa, además de la geometría, la velocidad de giro del cabezal, en el caso de que la velocidad no se encuentre en el rango especificado (umbrales de conmutación para la velocidad máxima y mínima) del escalón de reducción activo, no se puede realizar el cambio de escalón de reducción porque, de lo contrario, no se mantendría el desplazamiento de trayectoria del cabezal y de los ejes de penetración.

Como en el ejemplo anterior, en la secuencia G331 se vigila la velocidad de giro y el escalón de reducción; en caso necesario, se muestra la alarma 16748.

Código de programa	**Comentarios**
N05 M40 S500	; velocidad de giro del cabezal programada: 500 r/min =>
	; escalón de reducción 1
...	
N55 SPOS=0	; posicionar cabezal
N60 G331 Z-10 K5 S800	; roscado
	; el cambio de escalón no es posible,
	; vigilancia de la velocidad del cabezal a 800 r/min
	; con juego de datos de escalón de reducción 1: escalón de reducción 2
	; debería estar activo => alarma 16748

2.9.11.7 Ejemplo: programación sin SPOS

Código de programa	**Comentarios**
N05 M40 S500	; velocidad de giro del cabezal programada: 500 r/min =>
	; escalón de reducción 1 (20 a 1028 r/min)

Código de programa	**Comentarios**
...	
N50 G331 S800	; cabezal maestro: se selecciona el escalón de reducción 2
N60 G331 Z-10 K5	; roscado
	; aceleración del cabezal desde juego de datos de escalón de reducción 2

La interpolación de roscas para el cabezal comienza desde la posición actual, la cual depende del área de programa de pieza anteriormente ejecutada, p. ej., cuando se ha ejecutado un cambio del escalón de reducción. Por este motivo, puede que no sea posible repasar la rosca.

Nota

En el mecanizado con varios cabezales, se debe procurar que el cabezal de taladrado sea también el cabezal maestro. Programando `SETMS(<Número de cabezal>)`, el cabezal de taladrado se puede convertir en cabezal maestro.

2.9.12 Roscado con macho con mandril de compensación

2.9.12.1 Roscado con macho con mandril de compensación y movimiento de retirada (G63)

En el roscado con macho con mandril de compensación se ejecutan los siguientes movimientos de desplazamiento con el comando `G63`:

- `G63`: roscado en el sentido de taladrado hasta el punto final de rosca
- `G63`: movimiento de retirada con inversión del sentido de giro del cabezal programada

Nota

Después de una secuencia `G63` vuelve a activarse el último tipo de interpolación `G0`, `G1`, `G2` que estaba activo.

Sintaxis

```
G63 <Eje> <Sentido de giro> <Velocidad de giro> <Avance>
```

Descripción

`G63`:	Roscado de taladros con mandril de compensación	
	Efecto:	secuencial
`<Eje>`:	Recorrido de desplazamiento/posición del eje geométrico (X, Y o Z) respecto al punto final de rosca, p. ej., Z50	
`<Sentido de giro>`:	Sentido de giro del cabezal: • `M3`: sentido de giro horario, rosca a derechas • `M4`: sentido de giro antihorario, rosca a izquierdas	

<Velocidad de giro>:	velocidad de giro máxima admisible del cabezal durante el roscado, p. ej., S100
<Avance>:	avance del eje de taladrado F, donde F = velocidad de giro del cabezal * paso de rosca

Ejemplo

Taladrado de una rosca M5:

- Paso de rosca según norma: 0,8 mm/vuelta
- Velocidad de giro del cabezal S: 200 r/min
- Avance F = 200 r/min * 0,8 mm/vuelta = 160 mm/min.

```
Código de programa                  Comentarios
N10 G1 X0 Y0 Z2 F1000 S200 M3       ; desplazar al punto inicial
                                    ; cabezal en sentido horario, 200 r/min
N20 G63 Z-50 F160                   ; roscado con macho con mandril de compensación
                                    ; profundidad de taladrado: Z absoluto=50 mm
                                    ; avance: 160 mm/min
N30 G63 Z3 M4                       ; movimiento de retirada: Z absoluto=3 mm
                                    ; inversión del sentido de giro
                                    ; cabezal en sentido antihorario, 200 r/min
```

2.9.13 Chaflán, redondeo (CHF, CHR, RND, RNDM, FRC, FRCM)

Las esquinas de contorno dentro del plano de trabajo activo se pueden ejecutar como redondeo o chaflán.

Para optimizar la calidad de la superficie se puede programar un avance propio para el achaflanado/redondeado. Si no se programa ningún avance, actúa el avance de contorneado F normal.

Con la función de "Redondeo modal" se pueden redondear del mismo modo varias esquinas de contorno sucesivas.

Sintaxis

Achaflanar esquina de contorno:

```
G... X... Z... CHR/CHF=<Valor> FRC/FRCM=<Valor>
G... X... Z...
```

Redondear esquina de contorno:

```
G... X... Z... RND=<Valor> FRC=<Valor>
G... X... Z...
```

Redondeo modal:

```
G... X... Z... RNDM=<Valor> FRCM=<Valor>
...
```

```
RNDM=0
```

Nota

La tecnología (avance, tipo de avance, comandos M...) para el achaflanado/redondeado se deriva de la secuencia anterior o de la secuencia posterior dependiendo del ajuste de bit 0 en el dato de máquina DM20201 $MC_CHFRND_MODE_MASK (comportamiento chaflán/redondeo). El ajuste recomendado se deriva de la secuencia anterior (bit 0 = 1).

Descripción

CHF=… :	Achaflanar esquina del contorno	
	<Valor>:	Longitud del chaflán (unidades de medida acordes con G70/G71)
CHR=… :	Achaflanar esquina del contorno	
	<Valor>:	Anchura del chaflán en el sentido de desplazamiento original (unidades de medida acordes con G70/G71)
RND=… :	Redondear esquina de contorno	
	<Valor>:	Radio del redondeo (unidades de medida acordes con G70/G71)
RNDM=… :	Redondeo modal (redondear varias esquinas de contorno sucesivas de la misma manera)	
	<Valor>:	Radio de los redondeos (unidades de medida acordes con G70/G71)
		La desactivación del redondeo modal se realiza con RNDM=0.
FRC=… :	Avance por secuencia activo para achaflanado/redondeado	
	<Valor>:	Velocidad de avance en mm/min (con G94 activo) o mm/vuelta (con G95 activo)
FRCM=… :	Avance modal activo para achaflanado/redondeado	
	<Valor>:	Velocidad de avance en mm/min (con G94 activo) o mm/vuelta (con G95 activo)
		Con FRCM=0 se desactiva el avance modal activo para el achaflanado/redondeado y se activa el avance programado en F.

Nota

Chaflán/redondeo demasiado grande

Si los valores programados para el chaflán (CHF/CHR) o el redondeo (RND/RNDM) son demasiado grandes para los elementos de contorno implicados, el chaflán o el redondeo se adaptan automáticamente:

1. Si está activado el MD11411 $MN_ENABLE_ALARM_MASK bit 4, se emitirá la alarma 10833 "Chaflán o redondeo debe ser acortado" (alarma cancelable).
2. El chaflán/redondeo debe reducirse hasta que case con la esquina de contorno. Se generará al menos una secuencia sin movimiento. En esta secuencia el movimiento se detiene necesariamente.

Nota

Chaflán/redondeo no posible

El redondeo o el chaflán no se inserta cuando:

- No existe un elemento lineal o circular en el contorno para el plano.
- Se está realizando un desplazamiento fuera del plano.
- Se ha cambiado de plano.
- Se ha superado el número de secuencias sin desplazamiento definido en el dato de máquina
  (p. ej., mediante secuencias con funciones auxiliares).

Nota

FRC/FRCM

FRC/FRCM no actúan cuando se realiza un chaflán con G0; es posible realizar la programación del correspondiente valor F sin aviso de error.

FRC solamente actúa cuando en una secuencia se ha programado un chaflán/redondeo o se ha activado RNDM.

FRC sobrescribe en la secuencia actual el valor de F o de FRCM.

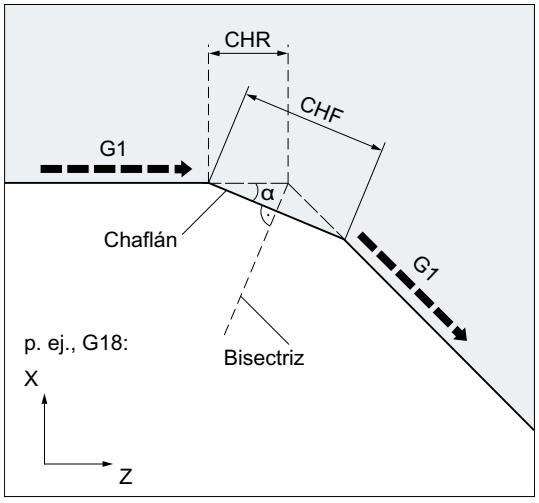
El valor de avance programado para FRC debe de ser mayor que cero.

FRCM=0 activa el avance programado en F para el achaflanado/redondeado.

Si se programa FRCM, al cambiar entre G94 ↔ G95 se debe volver a programar el valor FRCM, en forma análoga al valor F. Si solamente se reprograma el valor F y antes del cambio del tipo de avance FRCM > 0, se obtiene un aviso de error.

Ejemplos

Ejemplo 1: Achaflanado entre dos rectas



- MD20201 Bit 0 = 1 (derivada de la secuencia anterior)
- G71 está activo.
- La anchura del chaflán en el sentido de desplazamiento (CHR) debe ser 2 mm; el avance para el achaflanado, 100 mm/min.

La programación se puede realizar de dos maneras:

- Programación con CHR

Código de programa

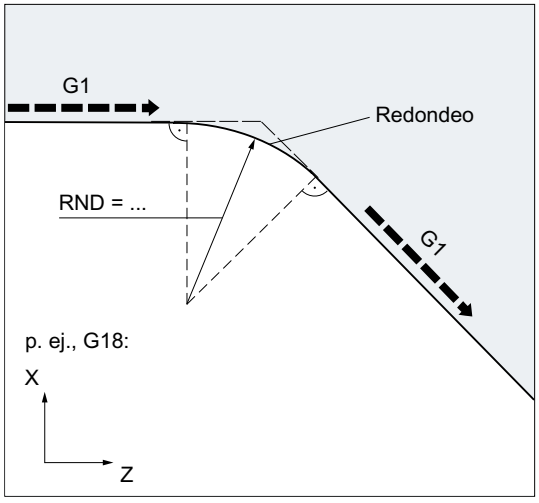
```
...
N30 G1 Z… CHR=2 FRC=100
N40 G1 X…
...
```

- Programación con CHF

Código de programa

```
...
N30 G1 Z… CHF=2(cosα*2) FRC=100
N40 G1 X…
...
```

Ejemplo 2: Redondeado entre dos rectas



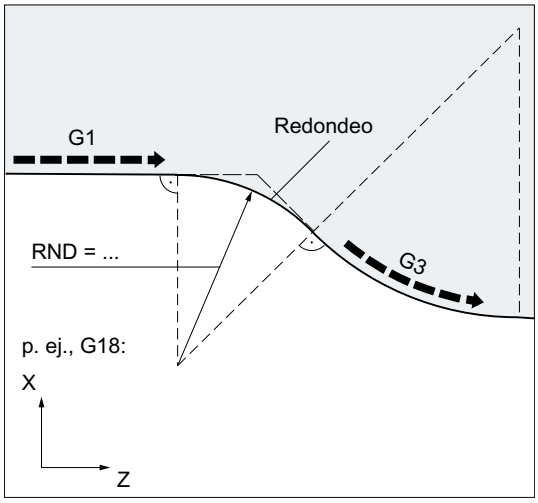
- MD20201 Bit 0 = 1 (derivada de la secuencia anterior)
- G71 está activo.
- El radio del redondeo debe ser 2 mm; el avance para el redondeado, 50 mm/min.

Código de programa

```
...
N30 G1 Z… RND=2 FRC=50
N40 G1 X…
...
```

Ejemplo 3: Redondeado entre recta y círculo

Entre dos elementos del contorno lineales o circulares combinados de cualquier manera se puede insertar, con la función RND, un redondeo de forma tangencial a los dos elementos.



- MD20201 Bit 0 = 1 (derivada de la secuencia anterior)
- G71 está activo.
- El radio del redondeo debe ser 2 mm; el avance para el redondeado, 50 mm/min.

Código de programa
...
N30 G1 Z… RND=2 FRC=50
N40 G3 X… Z… I… K…
...

Ejemplo 4: Redondeado modal para desbarbar cantos afilados de una pieza

Código de programa	**Comentarios**
...	
N30 G1 X… Z… RNDM=2 FRCM=50	; Activar redondeo modal.
	Radio del redondeo: 2 mm
	Avance para el redondeado: 50 mm/min
N40...	
N120 RNDM=0	; Desactivar redondeo modal.
...	

Ejemplo 5: Aplicación de la tecnología de la secuencia posterior o la secuencia anterior

- DM20201 Bit 0 = 0: Derivada de la secuencia posterior (ajuste por defecto)

Código de programa	**Comentarios**
N10 G0 X0 Y0 G17 F100 G94	
N20 G1 X10 CHF=2	; Chaflán N20-N30 con F=100 mm/min
N30 Y10 CHF=4	; Chaflán N30-N40 con FRC=200 mm/min
N40 X20 CHF=3 FRC=200	; Chaflán N40-N60 con FRCM=50 mm/min
N50 RNDM=2 FRCM=50	

Código de programa	**Comentarios**
N60 Y20	; Redondeo modal N60-N70 con FRCM=50 mm/min
N70 X30	; Redondeo modal N70-N80 con FRCM=50 mm/min
N80 Y30 CHF=3 FRC=100	; Chaflán N80-N90 con FRC=100 mm/min
N90 X40	; Redondeo modal N90-N100 con F=100 mm/min (desactivación de FRCM)
N100 Y40 FRCM=0	; Redondeo modal N100-N120 con G95 FRC=1 mm/ vuelta
N110 S1000 M3	
N120 X50 G95 F3 FRC=1	
...	
M02	

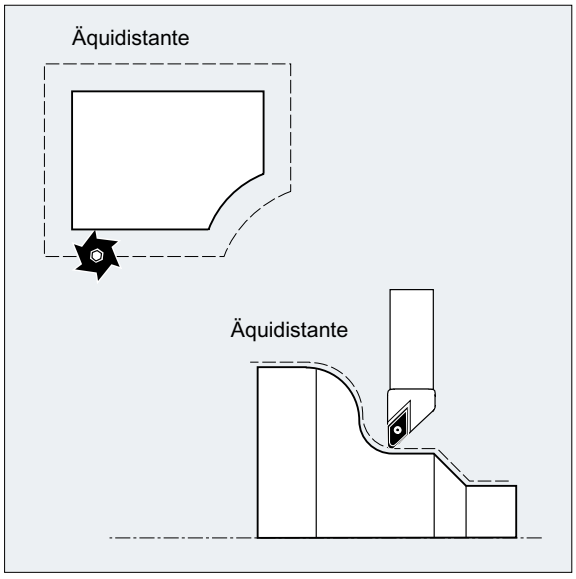
- DM20201 Bit 0 = 1: Derivada de la secuencia anterior (ajuste recomendado)

Código de programa	**Comentarios**
N10 G0 X0 Y0 G17 F100 G94	
N20 G1 X10 CHF=2	; Chaflán N20-N30 con F=100 mm/min
N30 Y10 CHF=4 FRC=120	; Chaflán N30-N40 con FRC=120 mm/min
N40 X20 CHF=3 FRC=200	; Chaflán N40-N60 con FRC=200 mm/min
N50 RNDM=2 FRCM=50	
N60 Y20	; Redondeo modal N60-N70 con FRCM=50 mm/min
N70 X30	; Redondeo modal N70-N80 con FRCM=50 mm/min
N80 Y30 CHF=3 FRC=100	; Chaflán N80-N90 con FRC=100 mm/min
N90 X40	; Redondeo modal N90-N100 con FRCM=50 mm/min
N100 Y40 FRCM=0	; Redondeo modal N100-N120 con F=100 mm/min
N110 S1000 M3	
N120 X50 CHF=4 G95 F3 FRC=1	; Chaflán N120-N130 con G95 FRC=1 mm/vuelta
N130 Y50	; Redondeo modal N130-N140 con F=3 mm/vuelta
N140 X60	
...	
M02	

2.10 Correcciones del radio de herramienta

2.10.1 Corrección del radio de herramienta (G40, G41, G42, OFFN)

Cuando se activa la corrección del radio de herramienta (WRK), el control calcula automáticamente trayectorias equidistantes al contorno para las distintas herramientas de mecanizado.



Sintaxis

G0/G1 X... Y… Z... **G41**/**G42** [**OFFN=<Valor>**]	
...	
G40 X... Y… Z...	

Descripción

G41:	Activar WRK con sentido de mecanizado a LA **izquierda** del contorno
G42:	Activar WRK con sentido de mecanizado a la **derecha** del contorno
OFFN=<Valor>:	Demasía/creces para el contorno programado (offset normal al contorno) (opcional) Z. ej., para crear trayectorias equidistantes para el desbaste basto.
G40:	Desactivar corrección de radio de herramienta

Nota

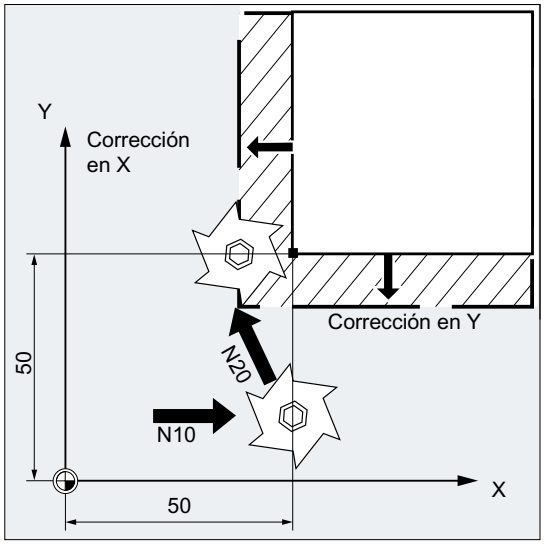
En la secuencia CN con G40/G41/G42 es necesario que G0 o G1 estén activos y que se defina al menos uno de los ejes del plano de trabajo seleccionado.

Si solamente se especifica un eje al conectar, se complementa automáticamente la última posición del eje no especificado, con lo cual se desplazan **ambos** ejes.

Los dos ejes tienen que estar activos en el canal como ejes geométricos. Esto se puede asegurar con la programación de GEOAX.

Ejemplos

Ejemplo 1: fresado

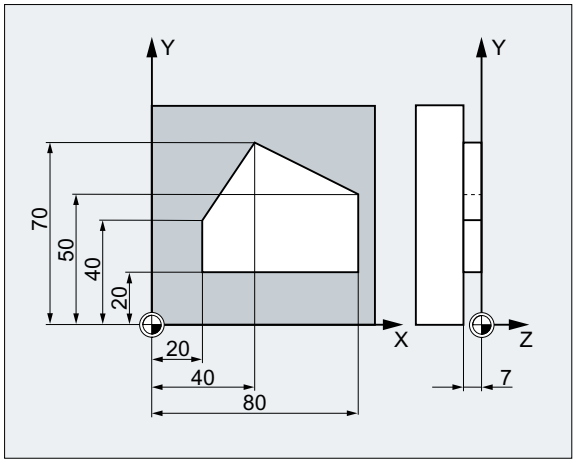


Código de programa	Comentarios
N10 G0 X50 T1 D1	; Sólo se activa la corrección longitudinal de herramienta. El desplazamiento a X50 se realiza sin corrección del radio de la herramienta.
N20 G1 G41 Y50 F200	; Se activa la corrección de radio, el desplazamiento a X50/Y50 se realiza corregido.
N30 Y100	
…	

Ejemplo 2: procedimiento "clásico" tomando como ejemplo el fresado

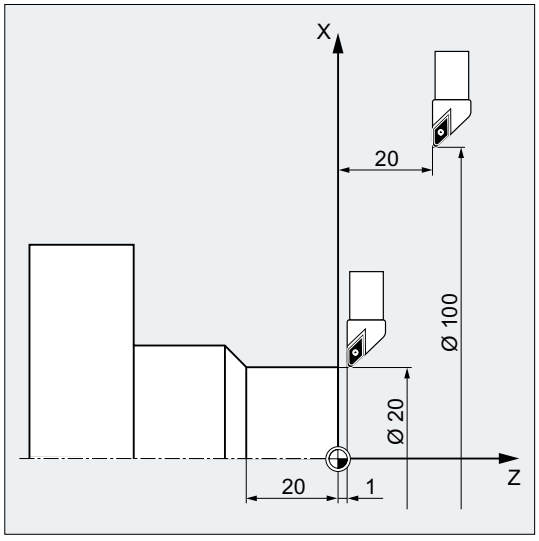
Procedimiento "clásico":

1. Llamada a herramienta
2. Inserción de la herramienta.
3. Activación del plano de trabajo y de la corrección del radio de herramienta.



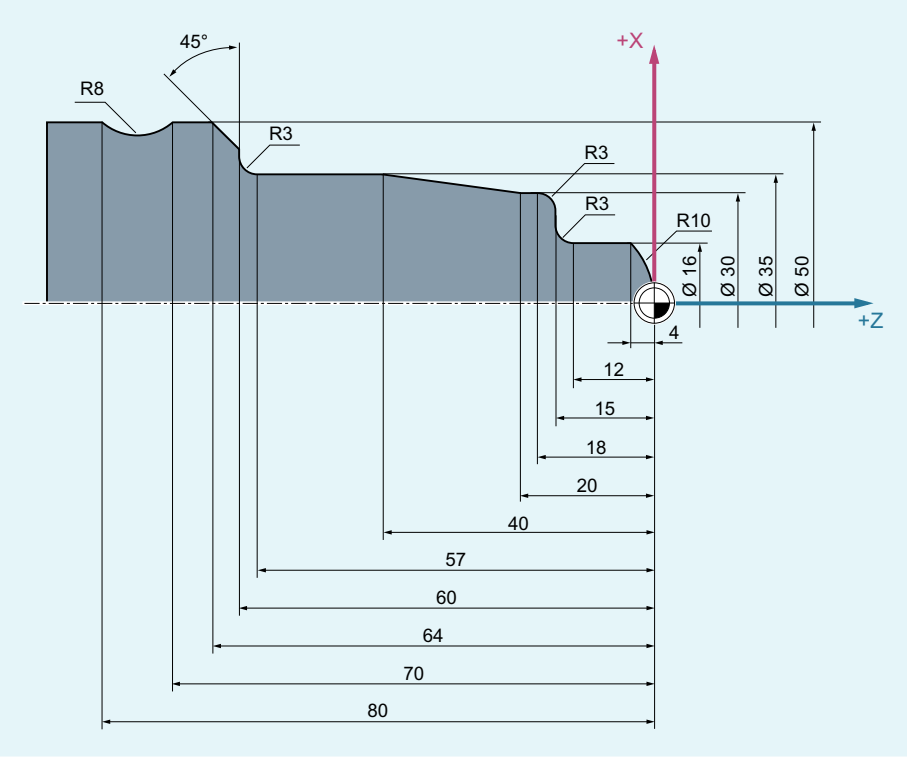
Código de programa	**Comentarios**
N10 G0 Z100	; Retirada al punto de cambio de herramienta.
N20 G17 T1 M6	; Cambio de herramienta
N30 G0 X0 Y0 Z1 M3 S300 D1	; Llamada a los valores de corrección de herramienta, selección de corrección longitudinal.
N40 Z-7 F500	; Penetración de herramienta.
N50 G41 X20 Y20	; Activación de la corrección del radio de herramienta; la herramienta se desplaza a la izquierda del contorno.
N60 Y40	; Fresado del contorno.
N70 X40 Y70	
N80 X80 Y50	
N90 Y20	
N100 X20	
N110 G40 G0 Z100 M30	; Retirada de la herramienta, fin del programa.

Ejemplo 3: torneado



Código de programa	**Comentarios**
…	
N20 T1 D1	; Sólo se activa la corrección longitudinal de herramienta.
N30 G0 X100 Z20	; El desplazamiento a X100 Z20 se realiza sin corrección.
N40 G42 X20 Z1	; Se activa la corrección de radio; el desplazamiento a X20/Z1 se realiza corregido.
N50 G1 Z-20 F0.2	
…	

Ejemplo 4: torneado



Código de programa	Comentarios
N5 G0 G53 X280 Z380 D0	; Punto inicial
N10 TRANS X0 Z250	; Decalaje de origen
N15 LIMS=4000	; Limitación de la velocidad de giro (G96)
N20 G96 S250 M3	; Selección del avance constante
N25 G90 T1 D1 M8	; Selección de la herramienta y la corrección
N30 G0 G42 X-1.5 Z1	; Posicionar herramienta con corrección del radio de herramienta
N35 G1 X0 Z0 F0.25	
N40 G3 X16 Z-4 I0 K-10	; Tornear radio 10
N45 G1 Z-12	
N50 G2 X22 Z-15 CR=3	; Tornear radio 3
N55 G1 X24	
N60 G3 X30 Z-18 I0 K-3	; Tornear radio 3
N65 G1 Z-20	
N70 X35 Z-40	
N75 Z-57	
N80 G2 X41 Z-60 CR=3	; Tornear radio 3
N85 G1 X46	
N90 X52 Z-63	

Código de programa	Comentarios
N95 G0 G40 G97 X100 Z50 M9	; Desactivar corrección del radio de herramienta y desplazar a punto de cambio de herramienta
N100 T2 D2	; Llamada de herramienta y selección de corrección
N105 G96 S210 M3	; Seleccionar velocidad de corte constante
N110 G0 G42 X50 Z-60 M8	; Posicionar herramienta con corrección del radio de herramienta
N115 G1 Z-70 F0.12	; Tornear diámetro 50
N120 G2 X50 Z-80 I6.245 K-5	; Tornear radio 8
N125 G0 G40 X100 Z50 M9	; Retirar la herramienta y desactivar la corrección del radio de herramienta
N130 G0 G53 X280 Z380 D0 M5	; Desplazamiento a punto de cambio de herramienta
N135 M30	; Fin de programa

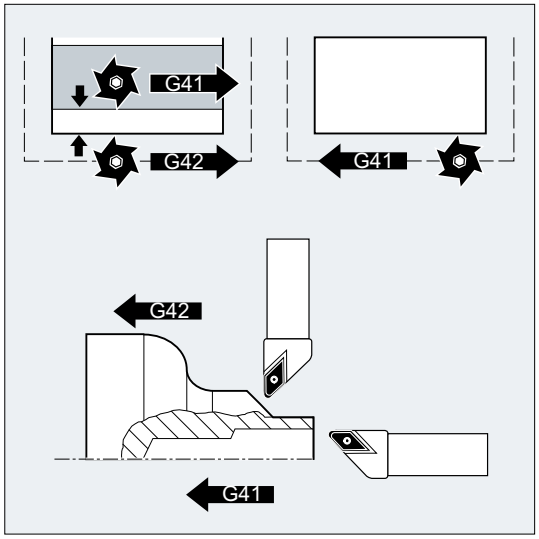
Información adicional

El control necesita la siguiente información para calcular la trayectoria de las herramientas:

- N.º de herramienta (T...), n.º de filo (D...)
- Sentido de mecanizado (G41/G42)
- Plano de trabajo (G17/G18/G19)

N.º de herramienta (T...), n.º de filo (D...)

La distancia entre la trayectoria de la herramienta y el contorno de la pieza se calcula a partir de los radios de fresa o bien los radios del filo y los datos sobre la posición del filo.



Sentido de mecanizado (G41/G42)

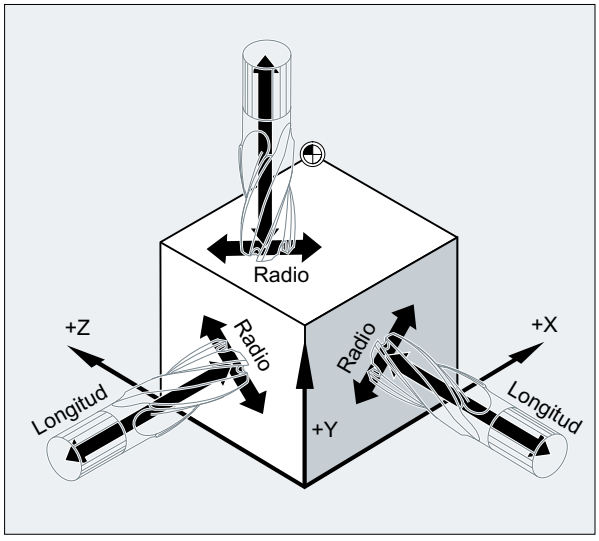
A partir de esta información, el control detecta el sentido en el que se debe desplazar la trayectoria de la herramienta.

Nota

Un valor de corrección negativo equivale a un cambio del lado de corrección (`G41` ↔ `G42`).

Plano de trabajo (G17/G18/G19)

Con esta información el CN detecta el plano de trabajo y, por lo tanto, las direcciones de eje en las que se corrige.



Ejemplo: Herramienta de fresado

Código de programa	Comentarios
...	
N10 G17 G41 …	; La corrección del radio de herramienta se realiza en el plano X/Y, la corrección de la longitud de herramienta se realiza en la dirección Z.
...	

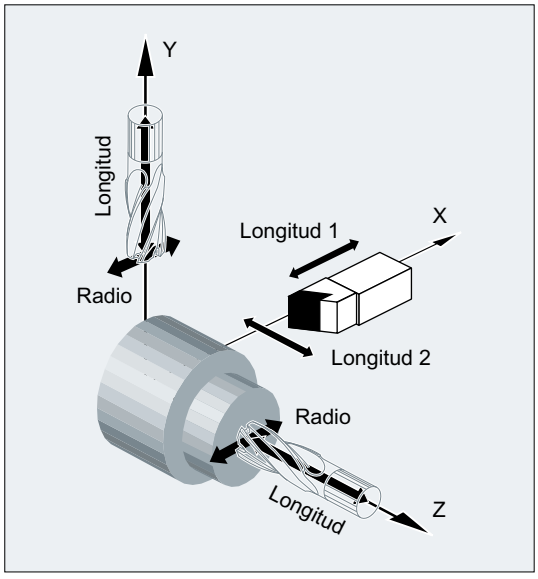
Nota

En el caso de máquinas de 2 ejes, la corrección del radio de herramienta solamente es posible en planos "reales", en general con `G18`.

Corrección de longitud de herramienta

El parámetro de desgaste asignado con la selección de herramienta del eje de diámetro se puede definir como valor de diámetro mediante un dato de máquina. En un siguiente cambio de plano se modificará automáticamente esta asignación. Para esto se deberá seleccionar nuevamente la herramienta después del cambio de plano.

Torneado:



Con NORM y KONT se puede definir la trayectoria de la herramienta al activar y desactivar el modo de corrección (ver "Aproximación y retirada del contorno (NORM, KONT, KONTC, KONTT) (Página 266)").

Punto de intersección

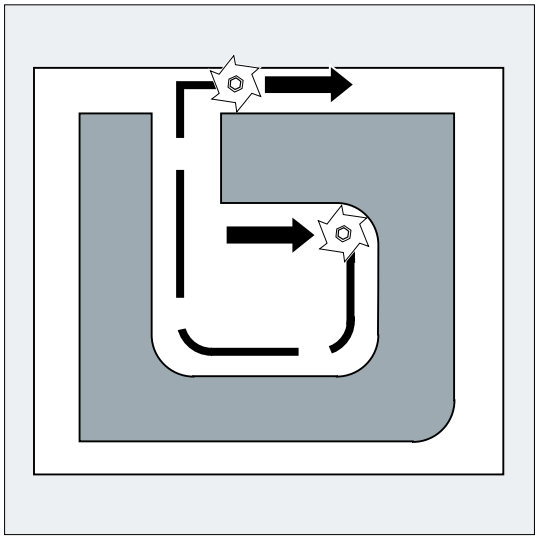
La selección del punto de intersección se realiza mediante el dato de operador:

SD42496 $SC_CUTCOM_CLSD_CONT (comportamiento de la corrección del radio de herramienta con contorno cerrado)

Valor	Descripción
FALSE	Si, en un contorno (prácticamente) cerrado compuesto por dos secuencias circulares consecutivas o una secuencia circular y una secuencia lineal, se producen dos puntos de intersección en la corrección en el lado interior, se elige, según el procedimiento estándar, el punto de intersección situado en el primer contorno parcial más cerca del fin de la secuencia. Un contorno se considera (prácticamente) cerrado cuando la distancia entre el punto inicial de la primera secuencia y el punto final de la segunda secuencia es inferior al 10% del radio de corrección activo, pero no superior a 1000 incrementos de trayecto (equivale a 1 mm con 3 decimales).
TRUE	En la situación anteriormente descrita se elige el punto de intersección situado en el primer contorno parcial más cercano al inicio de la secuencia.

Cambio de la dirección de corrección (G41 ↔ G42)

Un cambio de la dirección de corrección (G41 ↔ G42) puede programarse sin necesidad de intercalar G40.



Cambio del plano de trabajo

No es posible realizar un cambio del plano de trabajo (G17/G18/G19) con G41/G42 activos.

Cambio del bloque de datos de corrección de herramienta (D…)

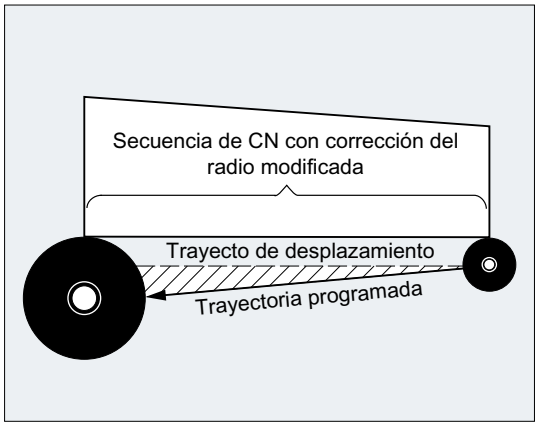
El bloque de datos de corrección de herramienta se puede cambiar en el modo de corrección.

El nuevo valor del radio de la herramienta se tiene en cuenta a partir de la secuencia en la que se encuentra el nuevo número D.

Nota

El cambio de radio o el movimiento de compensación se realizan a lo largo de toda la secuencia de manera que la nueva posición equidistante se alcanza en el punto final del desplazamiento programado.

En movimientos lineales, la herramienta se desplaza a lo largo de una trayectoria inclinada que une los puntos inicial y final:



En el caso de realizar interpolaciones circulares, la trayectoria que se obtiene es una espiral.

Cambio del radio de la herramienta

El cambio se puede realizar, p. ej., mediante variables del sistema. Para la ejecución se aplica lo mismo que para el cambio del bloque de datos de corrección de herramienta (D…).

Nota

Los valores modificados solo son válidos tras volver a programar T o D. La modificación se aplica en la siguiente secuencia.

Modo de corrección

El modo de corrección solo puede ser interrumpido por un número determinado de secuencias sucesivas o comandos M que no contengan comandos o indicaciones de desplazamiento en el plano de corrección.

Nota

El número de secuencias consecutivas o comandos M se puede parametrizar con un dato de máquina (ver las indicaciones del fabricante de la máquina).

Nota

¡Una secuencia con desplazamiento cero nulo también se cuenta como interrupción!

2.10.2 Aproximación y retirada del contorno (NORM, KONT, KONTC, KONTT)

Requisitos

Los comandos KONTC y KONTT sólo están disponibles si el control tiene habilitada la opción "Interpolación de polinomios".

Función

Con los comandos NORM, KONT, KONTC o KONTT, si la corrección del radio de herramienta está activada (G41/G42), el trayecto de aproximación/retirada de la herramienta se puede adaptar al contorno deseado o a la forma de la pieza en bruto.

Con KONTC o KONTT se cumplen las condiciones de continuidad en los tres ejes. De este modo, es admisible programar al mismo tiempo un componente de trayectoria perpendicular al plano de corrección.

Sintaxis

G41/G42 **NORM**/**KONT**/**KONTC**/**KONTT** X... Y... Z...	
...	
G40 X... Y... Z...	

Descripción

NORM:	Activar aproximación/retirada directa en una recta La herramienta se alinea perpendicular al punto de contorno.
KONT:	Activar aproximación/retirada rodeando el punto inicial/final según el comportamiento en esquinas programado G450 o G451
KONTC:	Activar aproximación/retirada de curvatura continua
KONTT:	Activar aproximación/retirada de tangente continua

Nota

Como secuencias de aproximación/retirada originales para KONTC y KONTT sólo se admiten secuencias G1. El control las sustituye por polinomios para la correspondiente trayectoria de aproximación/retirada.

Limitaciones

KONTT y KONTC no están disponibles en las variantes 3D de la corrección del radio de herramienta (CUT3DC, CUT3DCC, CUT3DF). Si a pesar de todo se programan, se conmuta a nivel interno del control a NORM sin aviso de error.

Ejemplo

KONTC

Empezando por el centro del círculo, se efectúa la aproximación a la circunferencia. En el punto final de la secuencia de aproximación, la dirección y el radio de curvatura son idénticos a los valores del siguiente círculo. En las dos secuencias de aproximación/retirada se efectúa simultáneamente la aproximación en dirección Z. En la siguiente figura se muestra la proyección perpendicular de la trayectoria de herramienta.

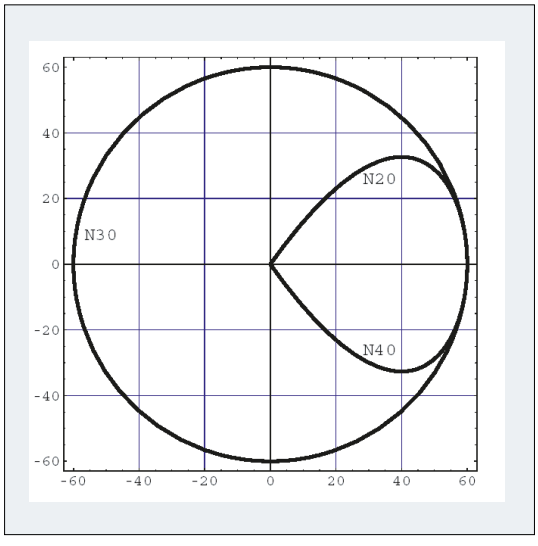


Figura 2-14 Proyección perpendicular

El correspondiente segmento de programa CN se presenta como sigue:

Código de programa	**Comentarios**
`$TC_DP1[1,1]=121`	`; Fresa`
`$TC_DP6 [1,1] = 10`	`; Radio 10 mm`
`N10 G1 X0 Y0 Z60 G64 T1 D1 F10000`	
`N20 G41 KONTC X70 Y0 Z0`	`; Aproximación`
`N30 G2 I-70`	`; Circunferencia`
`N40 G40 G1 X0 Y0 Z60`	`; Retirada`
`N50 M30`	

Simultáneamente a la adaptación de la curvatura a la trayectoria circular de la circunferencia se efectúa el desplazamiento de Z60 al plano del círculo Z0:

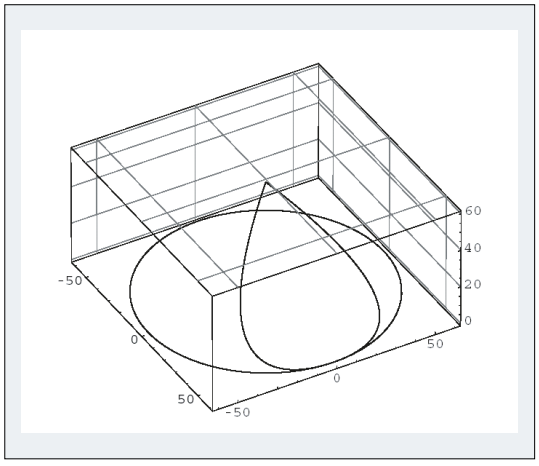
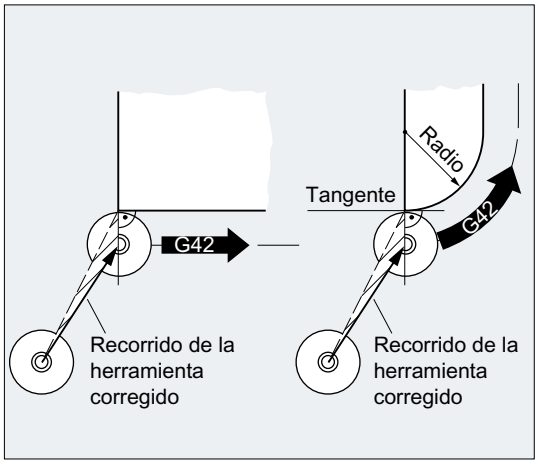


Figura 2-15 Representación en el espacio

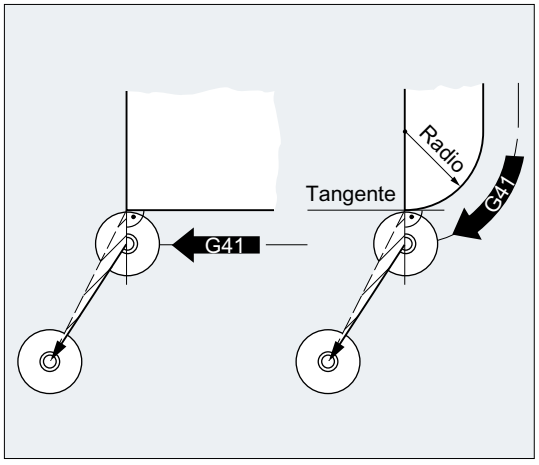
Información adicional

Aproximación/retirada con NORM

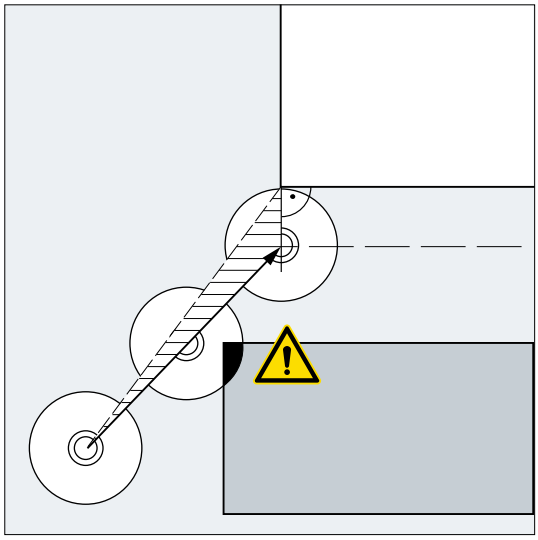
1. Aproximación:
   Cuando se activa `NORM`, la herramienta se desplaza directamente a lo largo de una línea recta hasta la posición inicial corregida (independientemente del ángulo de aproximación programado para este desplazamiento) y se posiciona perpendicular a la tangente de trayectoria en el punto inicial:



2. Retirada:
   La herramienta se posiciona perpendicular al último punto final corregido de la trayectoria y se desplaza (independientemente del ángulo de aproximación programado para este desplazamiento) directamente a lo largo de una línea recta hasta la siguiente posición no corregida, p. ej., al punto de cambio de herramienta:



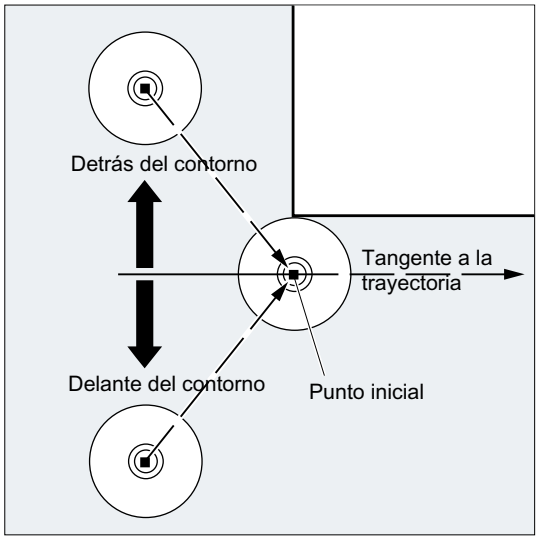
La modificación de los ángulos de aproximación/retirada implica peligro de colisión:



ATENCIÓN
Peligro de colisión A fin de evitar posibles colisiones, los ángulos de aproximación/retirada modificados deben tenerse en cuenta en la programación.

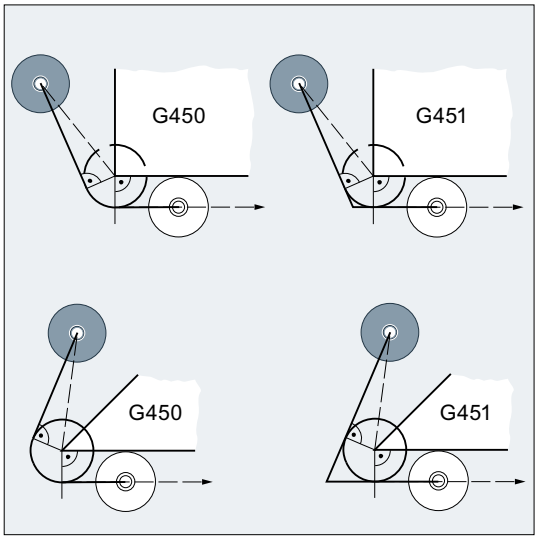
Aproximación/retirada con KONT

Antes de la aproximación, la herramienta puede encontrarse **delante** o **detrás** del contorno. La tangente de trayectoria sirve de línea separadora en el punto inicial:

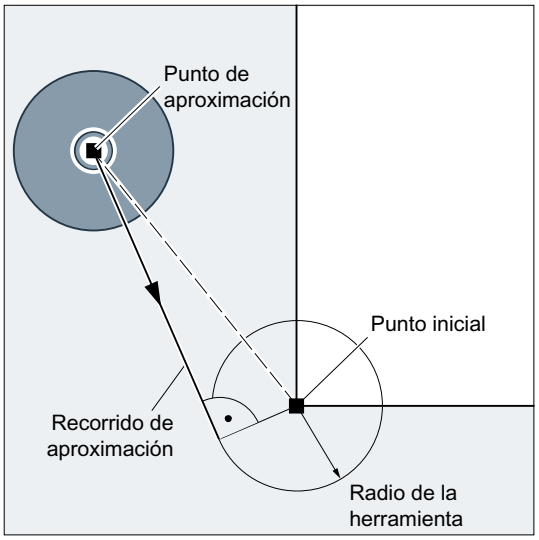


Conforme a ello, en la aproximación/retirada con `KONT` se pueden presentar dos casos:

1. La herramienta se encuentra delante del contorno.
   → La estrategia de aproximación/retirada es igual a la descrita para NORM.
2. La herramienta se encuentra detrás del contorno.
   – Aproximación:
   La herramienta se desplaza rodeando el punto inicial, teniendo en cuenta las opciones programadas para el comportamiento en esquinas (`G450`/`G451`), a lo largo de una trayectoria circular o bien siguiendo la intersección de las equidistantes.
   Los comandos `G450`/`G451` también definen cómo se va a realizar el cambio de la secuencia actual a la siguiente:



En ambos casos (`G450`/`G451`) se realiza el siguiente trayecto de posicionado:



Se traza una línea recta desde el punto de aproximación sin corregir que también sea tangente a un círculo de radio igual al de la herramienta. Esta línea describe la trayectoria de aproximación al punto inicial.

– Retirada:
  Para la retirada se aplican las mismas pautas que para la aproximación, pero en orden inverso.

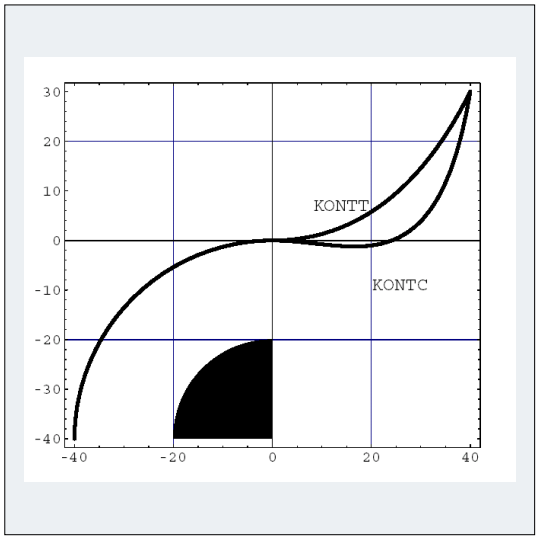
Aproximación/retirada con KONTC

La aproximación/retirada del contorno se efectúa con curvatura constante. En el contorno no se produce ningún salto de aceleración. La trayectoria del punto inicial al punto de contorno se interpola como polinomio.

Aproximación/retirada con KONTT

La aproximación/retirada del contorno se efectúa con tangente constante. En el punto de contorno se puede producir un salto de aceleración. La trayectoria del punto inicial al punto de contorno se interpola como polinomio.

Diferencia entre KONTC y KONTT

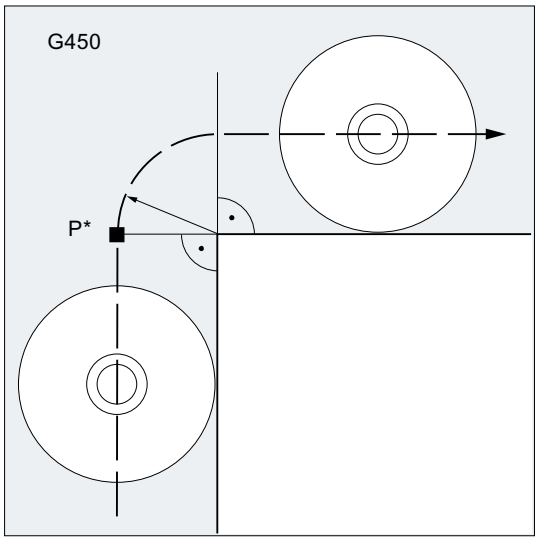


En esta figura se representan los distintos comportamientos de aproximación y retirada con `KONTT` y `KONTC`. Un círculo con un radio de 20 mm alrededor del punto central en X0 Y-40 se corrige con una herramienta con un radio de 20 mm en el lado exterior. Por esta razón, se obtiene un movimiento circular del centro de la herramienta con un radio de 40 mm. El punto final de la secuencia de retirada se sitúa en X40 Y30. La transición entre la secuencia circular y la secuencia de retirada se sitúa en el punto cero. Debido a la curvatura continua exigida con `KONTC`, la secuencia de retirada ejecuta primero un movimiento con una componente Y negativa. Frecuentemente, no se deseará este hecho. La secuencia de retirada con `KONTT` no muestra este comportamiento. Sin embargo, se produce, en este caso, un salto de aceleración en la transición de secuencia.

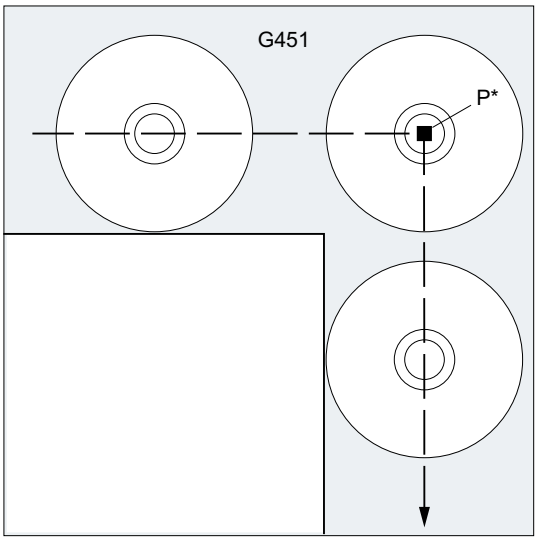
Si la secuencia `KONTT` o `KONTC` no es la secuencia de retirada, sino la secuencia de aproximación, se obtiene exactamente el mismo contorno, con la única diferencia que se ejecuta en el sentido inverso.

2.10.3 Corrección en las esquinas exteriores (G450, G451, DISC)

Con el comando G450 o G451, si la corrección del radio de herramienta está activada (G41/G42), se define el desarrollo de la trayectoria de herramienta corregida al rodear esquinas exteriores:



Con G450, el centro de la herramienta se desplaza rodeando la esquina exterior de la pieza describiendo un arco de circunferencia cuyo radio coincide con el radio de la herramienta.



Con G451, el centro de la herramienta se desplaza hasta la intersección de las dos equidistantes que se encuentran a una distancia del contorno programado igual al radio de la herramienta. G451 solo se puede aplicar a líneas rectas y arcos de circunferencia.

Nota

Con G450/G451, también se define el recorrido de aproximación con la función KONT activa y el punto inicial por detrás del contorno (ver "Aproximación y retirada del contorno (NORM, KONT, KONTC, KONTT) (Página 266)").

Con el comando DISC se pueden deformar arcos de transición con G450 y crear así esquinas de contorno agudas.

Sintaxis

```
G450 [DISC=<Valor>]
G451
```

Descripción

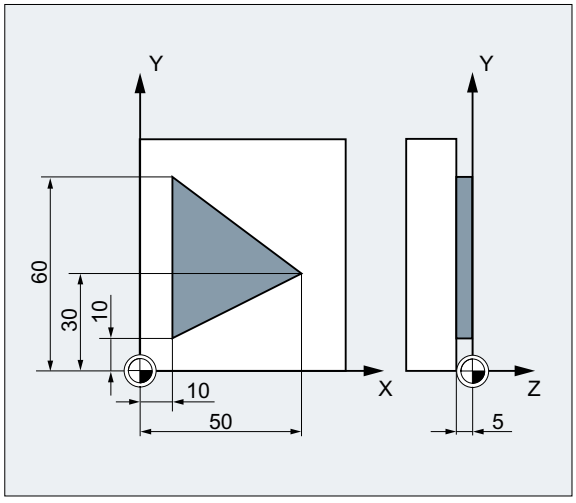
G450:	Con G450 se pueden rodear las esquinas de la pieza siguiendo una trayectoria circular.			
DISC:	Programación flexible de la trayectoria circular con G450 (opcional)			
	<Valor>:	Tipo:	INT	
		Rango de valores:	0, 1, 2 ... 100	
		Significado:	0	Circunferencia de transición
			100	Intersección de las equidistantes (valor teórico)
G451:	En caso de existir esquinas en la pieza, el desplazamiento a la intersección de ambas equidistantes se efectúa con G451. La herramienta sale en el ángulo de la pieza.			

Nota

DISC solamente actúa cuando se ha llamado al comando G450, pero también se puede programar en una secuencia previa sin G450. Ambos comandos son de tipo modal.

Ejemplo

En este ejemplo se inserta un redondeo de transición en todas las esquinas exteriores (conforme a la programación del comportamiento en esquinas en la secuencia N30). Esto evita que la herramienta se tenga que parar en el cambio de sentido y haga un corte no guiado.



Código de programa	**Comentarios**
N10 G17 T1 G0 X35 Y0 Z0 F500	; Condiciones iniciales
N20 G1 Z-5	; Penetración de herramienta.
N30 G41 KONT **G450** X10 Y10	; **Activación** de corrección de radio de herramienta con modo de aproximación/retirada KONT y **comportamiento en esquinas G450.**
N40 Y60	; Fresado del contorno.
N50 X50 Y30	
N60 X10 Y10	

Código de programa	Comentarios
N80 G40 X-20 Y50	; Desactivación del modo de corrección, retirada por circunferencia de transición.
N90 G0 Y100	
N100 X200 M30	

Información adicional

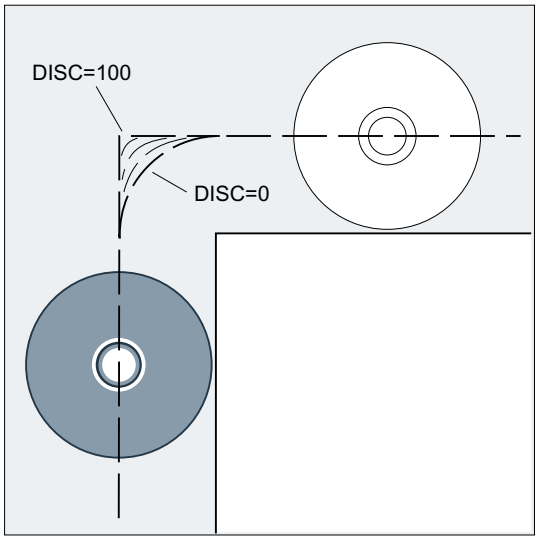
G450/G451

En el punto intermedio P*, el control ejecuta instrucciones tales como desplazamientos de penetración o funciones de conmutación. Estas instrucciones se programan en secuencias insertadas entre las dos secuencias que conforman la esquina.

Desde el punto de vista del procesamiento de datos, la circunferencia de transición con `G450` pertenece a la siguiente secuencia de desplazamiento.

DISC

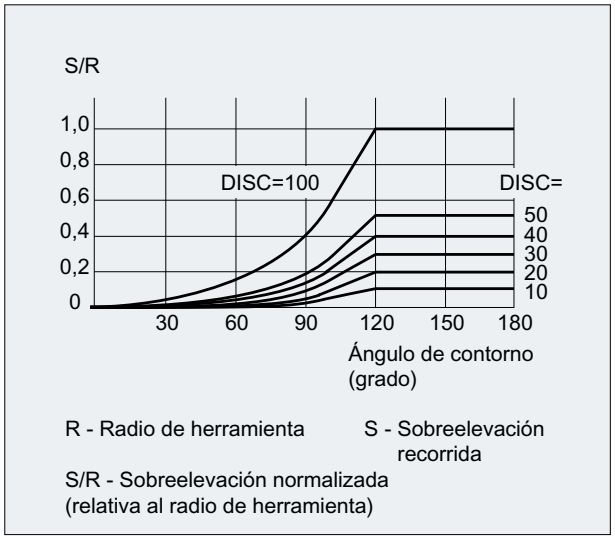
Cuando se introducen valores de `DISC` mayores que 0, los arcos de la circunferencia de transición se deforman, transformándose en elipses, parábolas o hipérbolas de transición.



Mediante datos de máquina se puede definir un valor límite superior (generalmente DISC=50).

Comportamiento de desplazamiento

Si `G450` está activado, la herramienta se aparta del contorno en las esquinas cuando los ángulos del contorno son agudos y los valores `DISC` elevados. Si el ángulo del contorno es superior a 120°, el contorno se rodea de modo uniforme:

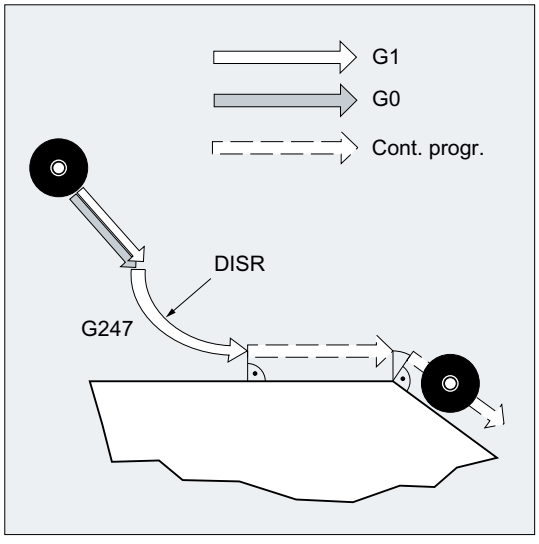


Si `G451` está activado, los movimientos de retirada en esquinas agudas pueden hacer que la herramienta siga una trayectoria con mucho espacio muerto. Para evitarlo, en estos casos se puede definir por datos de máquina el cambio automático al desplazamiento siguiendo circunferencias de transición.

2.10.4 Aproximación y retirada suaves

2.10.4.1 Aproximación y retirada (G140 a G143, G147, G148, G247, G248, G347, G348, G340, G341, DISR, DISCL, DISRP, FAD, PM, PR)

La función "Aproximación y retirada suaves (WAB)" se utiliza para aproximarse tangencialmente al punto inicial de un contorno independientemente de la posición del punto de partida para dicho desplazamiento.



La función se utiliza principalmente junto con la corrección del radio de la herramienta.

Al activarse la función, el control asume la tarea de calcular los puntos intermedios de modo que el paso a la secuencia siguiente (o el paso desde la secuencia anterior en el caso de la retirada) se produzca de conformidad con los parámetros especificados.

El movimiento de aproximación consta de un máximo de 4 desplazamientos parciales. El punto inicial del movimiento se identifica en lo sucesivo mediante P_0, y el punto final mediante P_4. Entre ambos pueden existir un máximo de tres puntos intermedios P_1, P_2 y P_3. Los puntos P_0, P_3 y P_4 siempre están definidos. Los puntos intermedios P_1 y P_2 se pueden suprimir según la parametrización y las condiciones geométricas. En la retirada, los puntos se recorren en orden inverso, es decir, empezando por P_4 y acabando con P_0.

Sintaxis

Aproximación suave:

- siguiendo una recta:
  `G147 G340/G341 ... DISR=..., DISCL=..., DISRP=... FAD=...`
- siguiendo un cuadrante o una semicircunferencia:
  `G247/G347 G340/G341 G140/G141/G142/G143 ... DISR=... DISCL=... DISRP=... FAD=...`

Retirada suave:

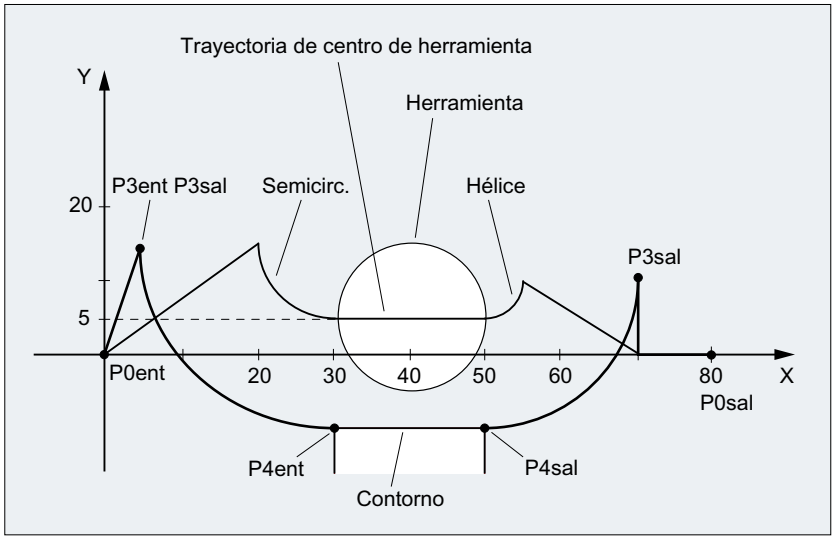
- siguiendo una recta:
  `G148 G340/G341 ... DISR=..., DISCL=..., DISRP=... FAD=...`
- siguiendo un cuadrante o una semicircunferencia:
  `G248/G348 G340/G341 G140/G141/G142/G143 ... DISR=... DISCL=... DISRP=... FAD=...`

Descripción

`G147`:	Aproximación siguiendo una recta
`G148`:	Retirada siguiendo una recta
`G247`:	Aproximación siguiendo un cuarto de circunferencia
`G248`:	Retirada siguiendo un cuarto de circunferencia
`G347`:	Aproximación siguiendo un semicírculo
`G348`:	Retirada siguiendo un semicírculo
`G340`:	Aproximación y retirada en el espacio (ajuste básico)
`G341`:	Aproximación y retirada en el plano
`G140`:	Aproximación y retirada dependiente del lado de corrección actual (ajuste básico)
`G141`:	Aproximación por la izquierda o bien retirada hacia la izquierda
`G142`:	Aproximación por la derecha o bien retirada hacia la derecha
`G143`:	Aproximación o retirada dependiente de la posición relativa del punto inicial o final respecto a la dirección de la tangente

`DISR=...`:	1. En la aproximación y retirada siguiendo rectas (G147/G148): Distancia entre el borde de la fresa y el punto inicial del contorno 2. En la aproximación y retirada siguiendo un arco (G247, G347/G248, G348): Radio de la trayectoria del centro de la herramienta. **Atención:** En REPOS siguiendo un semicírculo, DISR corresponde al diámetro del círculo.
`DISCL=...`:	Distancia entre el punto final del movimiento de aproximación rápido y el plano de mecanizado DISCL=AC(...) Indicación de la posición absoluta del punto final de la penetración rápida a profundidad de trabajo
`DISCL=AC(...)`:	Indicación de la posición absoluta del punto final de la penetración rápida a profundidad de trabajo
`DISRP`:	Distancia entre el punto P1 (plano de retirada) y el plano de mecanizado
`DISRP=AC(...)`:	Indicación de la posición absoluta del punto P1
`FAD=...`:	Velocidad del desplazamiento de penetración lento El valor programado se aplica conforme al tipo de avance activo (grupo G 15).
`FAD=PM(...)`:	El valor programado se interpreta como avance lineal (igual que G94) sea cual sea el tipo de avance activo.
`FAD=PR(...)`:	El valor programado se interpreta como avance por vuelta (igual que G95) sea cual sea el tipo de avance activo.

Ejemplo



- Aproximación suave (activada en la secuencia N20)
- Desplazamiento de aproximación siguiendo un cuadrante de circunferencia (G247)
- Dirección de aproximación no programada, es válido G140, es decir la corrección del radio de la herramienta está activo (G41)

- Decalaje de contorno OFFN=5 (N10)
- Corrección del radio de la herramienta actual=10, entonces la corrección del radio efectiva para corrección del radio de la herramienta=15, el radio de la aproximación/retirada suaves del contorno=25, de manera que el radio de la trayectoria del centro de la herramienta es DISR=10
- El punto final del arco se obtiene de la N30, ya que en la N20 solamente se ha programado la posición Z.
- Penetración a profundidad de trabajo
  - Desde Z20 a Z7 (DISCL=AC(7)) en rápido.
  - A continuación a Z0 con FAD=200.
  - Círculo de aproximación en el plano X-Y y en las siguientes secuencias con F1500 (para que esta velocidad sea válida en las siguientes secuencias, la G0 activada en N30 se deberá de sobrescribir con G1, en caso contrario se seguirá mecanizando el contorno con G0).
- Retirada suave (activado en la secuencia N60)
- Desplazamiento de retirada siguiendo un cuadrante de circunferencia (G248) y una hélice (G340)
- No se ha prog. FAD, ya que carece de significado con G340
- Z=2 en punto inicial; Z=8 en punto final, ya que DISCL=6
- Con DISR=5 el radio de la aprox./retir.. suaves del cont.=20, el radio de la trayectoria del centro de la herramienta=5

Desplazamiento de retirada desde Z8 hasta Z20 y desplazamiento paralelo al plano X-Y hasta X70 Y0.

Código de programa	**Comentarios**
`$TC_DP1[1,1]=120`	`; Definición de la herramienta T1/D1`
`$TC_DP6[1,1] = 10`	`; Radio`
`N10 G0 X0 Y0 Z20 G64 D1 T1 OFFN=5`	`; (P0 aproximación)`
`N20 G41 G247 G341 Z0 DISCL=AC(7) DISR=10 F1500 FAD=200`	`; Aproximación (P3aprox)`
`N30 G1 X30 Y-10`	`; (P4 aproximación)`
`N40 X40 Z2`	
`N50 X50`	`; (P4 retirada)`
`N60 G248 G340 X70 Y0 Z20 DISCL=6 DISR=5 G40 F10000`	`; Retirada (P3retir)`
`N70 X80 Y0`	`; (P0 retirada)`
`N80 M30`	

Información adicional

Selección del contorno de aprox. o retirada

La selección del contorno de aproximación y retirada se efectúa con el correspondiente comando G del 2.º Grupo G:

G147:	Aproximación siguiendo una recta
G247:	Aproximación siguiendo un cuarto de circunferencia
G347:	Aproximación siguiendo un semicírculo
G148:	Retirada siguiendo una recta
G248:	Retirada siguiendo un cuarto de circunferencia
G348:	Retirada siguiendo un semicírculo

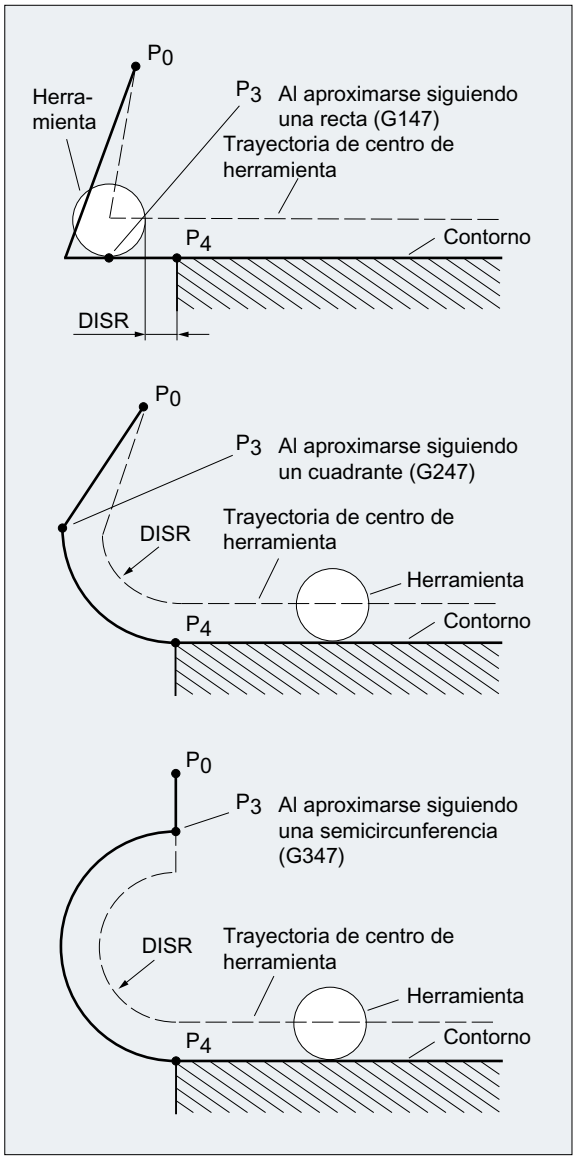


Figura 2-16 Aproximaciones con activación simultánea de la corrección de radio de herramienta

Selección de la dirección de aprox. o retirada

Definición de la dirección de aproximación o retirada con ayuda de la corrección del radio de herramienta (G140, ajuste básico) con un radio de herramienta positivo:

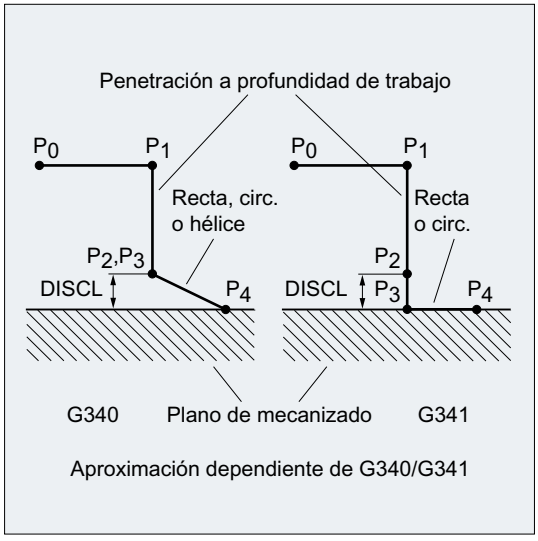
- G41 activo → aproximación desde la izquierda
- G42 activo → aproximación desde la derecha

Las otras posibilidades de aproximación se definen con G141, G142 y G143.

Estos comandos G solo tienen sentido cuando el contorno de aproximación es un cuadrante o un semicírculo.

División del desplazamiento desde el punto inicial hasta el punto final (G340 y G341)

En todos los casos, los movimientos se componen de una o varias rectas, así como, dependiendo del comando G para el cálculo del contorno de aproximación, de otra recta o un cuadrante o semicírculo. Las 2 variantes de división de la trayectoria se muestran en la siguiente figura:



G340:	Aproximación siguiendo una recta desde el punto P_0 al punto P_1. Si no se ha programado el parámetro DISRP, esta recta es paralela al plano de mecanizado. Penetración perpendicular al plano de mecanizado desde el punto P_1 al punto P_3 a la distancia de seguridad respecto al plano de mecanizado definida por el parámetro DISCL. Aproximación al punto final P4 siguiendo la curva definida mediante el comando G del segundo grupo (recta, arco, hélice). Si están activos G247 o G347 (cuadrante o semicírculo) y el punto inicial P_3 no se encuentra en el plano de mecanizado definido por el punto final P_4, en lugar de un círculo se inserta una hélice. El punto P_2 no está definido o coincide con P_3. El plano del arco o el eje de la hélice se define mediante el plano activo en la secuencia de aproximación/retirada suaves (G17/G18/G19), es decir, la secuencia siguiente no utiliza la propia tangente inicial para definir el círculo, sino la proyección de la tangente hacia el plano activo. El movimiento del punto P_0 al punto P_3 se produce en dos rectas a la velocidad que estaba activa antes de la secuencia de aproximación/retirada suaves.
G341:	Aproximación siguiendo una recta desde el punto P_0 al punto P_1. Si no se ha programado el parámetro DISRP, esta recta es paralela al plano de mecanizado. Penetración perpendicular al plano de mecanizado desde el punto P_1 hasta la distancia de seguridad respecto al plano de mecanizado definida por el parámetro DISCL en el punto P_2. Penetración perpendicular al plano de mecanizado desde el punto P_2 hasta el punto P_3. Aproximación al punto final siguiendo la curva definida mediante el comando G del segundo grupo. P_3 y P_4 se encuentran en el plano de mecanizado, por lo que en G247 y G347 nunca se insertará una hélice, sino siempre un arco.

En todos los casos que incluyan la posición del plano activo G17/G18/G19 (plano del arco, eje de la hélice, desplazamientos de penetración perpendicular al plano activo), se tendrá en cuenta cualquier frame activo girado.

Longitud de la recta de aproximación o radio del círculo de aproximación (DISR)

- Aproximación/retirada siguiendo una recta
  DISR indica la distancia de la esquina de la fresa al punto inicial del contorno; es decir, la longitud de la recta se obtiene cuando la corrección del radio de la herramienta está activa como la suma del radio de la herramienta y el valor programado para DISR. El radio de la herramienta solo se tiene en cuenta si es positivo.
  La longitud de recta resultante debe ser positiva, es decir, se admiten valores negativos para DISR solo en tanto el valor de DISR sea inferior al radio de la herramienta.
- Aproximación/retirada siguiendo un arco
  DISR indica el radio para la trayectoria del centro de la herramienta. Si está activada la corrección del radio de la herramienta, entonces se realiza un arco cuyo radio en este caso resulta ser el programado para la trayectoria del centro de la herramienta.

Distancia del punto P2 al plano de mecanizado (DISCL)

Si se desea indicar la posición del punto P_2 como valor absoluto en el eje perpendicular al plano del círculo, entonces el valor se deberá programar en la forma `DISCL=AC(...)`.

Con DISCL=0 es válido lo siguiente:

- Con G340: Todo el movimiento de aproximación ya solo se compone de dos secuencias (P_1, P_2 y P_3 coinciden). El contorno de aproximación es formado por P_1 a P_4.
- Con G341: Todo el movimiento de aproximación se compone de tres secuencias (P_2 y P_3 coinciden). Si P_0 y P_4 se sitúan en el mismo plano, sólo se forman dos secuencias (el movimiento de aproximación de P_1 a P_3 se suprime).
- Se vigila que el punto definido por DISCL se encuentre entre P_1 y P_3; es decir, que para todos los movimientos con una componente perpendicular al plano de mecanizado, dicha componente deberá tener el mismo signo.
- Cuando se detecte un cambio de sentido, se permitirá una determinada tolerancia definida en el DM20204 $MC_WAB_CLEARANCE_TOLERANCE.

Distancia entre el punto P1 (plano de retirada) y el plano de mecanizado (DISRP)

Si se desea indicar la posición del punto P_1 como valor absoluto en el eje perpendicular al plano de mecanizado, entonces el valor se deberá programar en la forma `DISRP=AC(...)`.

Si este parámetro no está programado, el punto P_1 tiene la misma distancia respecto al plano de mecanizado que el punto P_0, es decir, la recta de aproximación $P_0 \rightarrow P_1$ es paralela al plano de mecanizado.

Se vigila que el punto definido por DISRP se encuentre entre P_0 y P_2; es decir: para todos los movimientos con una componente perpendicular al plano de mecanizado (penetraciones, aproximaciones de P_3 a P_4), dicha componente deberá tener el mismo signo. No se permite la inversión del sentido. Dado el caso, se emite una alarma.

Cuando se detecte un cambio de sentido, se permitirá una determinada tolerancia definida en el DM20204 $MC_WAB_CLEARANCE_TOLERANCE. Si P_1 se encuentra fuera del rango definido por P_0 y P_2, pero la desviación es menor o igual que esa tolerancia, se asume que P_1 se encuentra en el plano definido por P_0 o P_2.

Programación del punto final

Generalmente se programa el punto final con
X... Y... Z...

La programación del punto final del contorno para la aproximación se distingue claramente de la utilizada para la retirada. Por eso tratamos aquí ambos casos por separado.

Programación del punto final P4 para la aproximación

El punto final P_4 puede estar programado en la propia secuencia de aproximación/retirada suaves. Alternativamente es posible determinar P_4 mediante el punto final de la siguiente secuencia de desplazamiento. Entre la secuencia de aproximación y retirada suaves del contorno y la siguiente secuencia de desplazamiento se pueden insertar secuencias adicionales sin desplazamiento de los ejes geométricos.

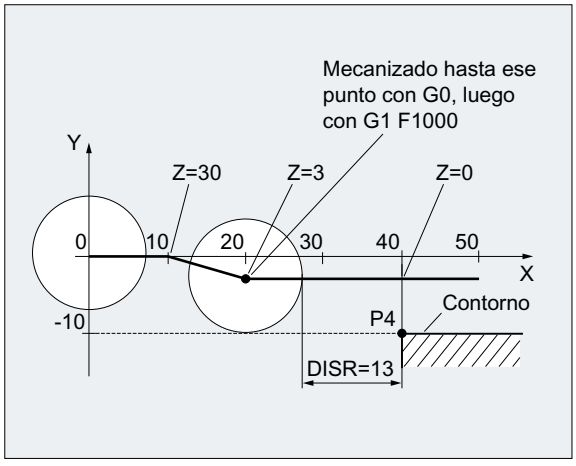
Ejemplo:

Código de programa	**Comentarios**
`$TC_DP1[1,1]=120`	`; Fresa T1/D1`
`$TC_DP6[1,1]=7`	`; Herramienta con un radio de 7 mm`
`N10 G90 G0 X0 Y0 Z30 D1 T1`	
`N20 X10`	

Código de programa	**Comentarios**
N30 G41 G147 DISCL=3 DISR=13 Z=0 F1000	
N40 G1 X40 Y-10	
N50 G1 X50	
...	

N30/N40 se puede sustituir por:
N30 G41 G147 DISCL=3 DISR=13 X40 Y-10 Z0 F1000

o bien
N30 G41 G147 DISCL=3 DISR=13 F1000
N40 G1 X40 Y-10 Z0



Programación del punto final P0 para la aproximación

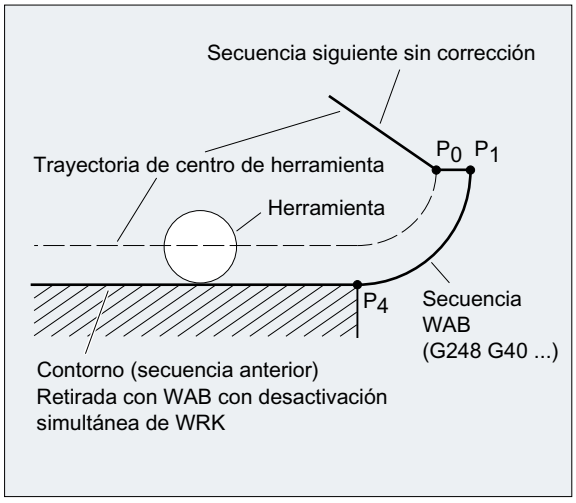
En la retirada, no está prevista la programación del punto final del contorno de aproximación/ retirada suaves en una secuencia siguiente, es decir, la posición final se obtiene siempre de la

secuencia WAB, sin importar cuántos ejes se hayan programado. Para determinar el punto final debe distinguirse, sin embargo, entre tres casos:

1. En la secuencia WAB (aproximación/retirada suaves) no hay ningún eje geométrico programado. En este caso, el contorno finaliza en el punto P_1 (si está programado DISRP), en el punto P_2 (si está programado DISCL pero no DISRP) o en el punto P_3 (si no están programados ni DICLS ni DISRP).
   La posición en los ejes que forman el plano de mecanizado resulta del contorno de retirada (punto final de la recta o del arco). La componente de eje perpendicular se define con DISCL o DISPR. Si en este caso DISCL=0 y DISRP=0, el movimiento se realiza por completo en el plano, es decir, los puntos P_0 a P_3 coinciden.
2. En la secuencia de aproximación/retirada suaves solo está programado el eje perpendicular al plano de mecanizado. En este caso, el contorno finaliza en el punto P_0. En caso de que DISRP esté programado (es decir, que los dos puntos P_0 y P_1 no coincidan), la recta discurre de $P_1 \rightarrow P_0$ perpendicular al plano de mecanizado. Las posiciones de los dos ejes restantes se obtienen como en 1.
3. Está programado al menos un eje del plano de mecanizado. El segundo eje eventualmente ausente del plano de mecanizado se completa de forma modal a partir de su última posición en la secuencia anterior.

La posición del eje perpendicular al plano de mecanizado se forma como en 1. o 2., en función de si dicho eje está programado o no. La posición obtenida de este modo define el punto final P_0. Si la secuencia de retirada WAB es, al mismo tiempo, la secuencia de desactivación de la corrección de radio de herramienta, en los dos primeros casos se inserta en el plano de mecanizado un componente adicional de trayectoria de P_1 a P_0 de tal modo que no se produce ningún movimiento al desactivarse la corrección de radio de herramienta al final del contorno de retirada, es decir, dicho punto ya no define entonces una posición en un contorno que se debe corregir, sino el centro de la herramienta. En el caso tres, la cancelación de la corrección del radio de herramienta no debe tratarse por separado, ya que el punto programado P_0 ya define de manera inmediata la posición del centro de la herramienta al final del contorno completo.

El comportamiento en los casos 1 y 2, es decir, sin un punto final explícitamente programado en el plano de mecanizado con cancelación simultánea de la corrección de radio de herramienta, se muestra en la siguiente figura:

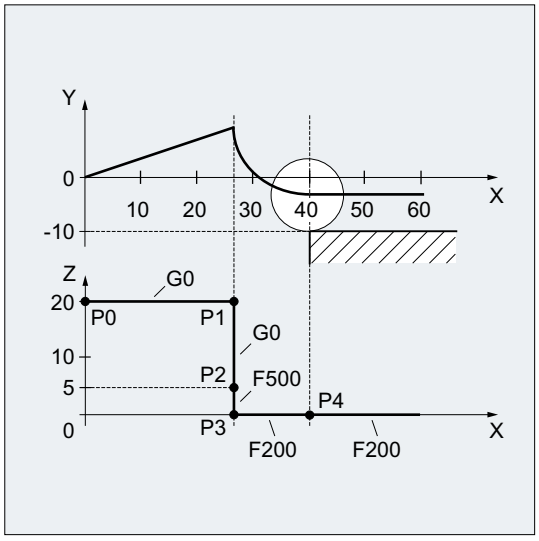


Velocidad de aproximación o retirada del contorno

- Velocidad de la secuencia anterior (G0)
  Se realizan todos los desplazamientos desde P_0 a P_2 con esta velocidad; es decir, el desplazamiento paralelo al plano de mecanizado y la parte de la penetración hasta la distancia de seguridad.
- Programación con FAD
  Indicación de la velocidad de avance con
  - G341: Movimiento de aproximación perpendicular al plano de trabajo P_2 a P_3
  - G340: del punto P_2 o P_3 al punto P_4
    Si no se programa FAD, esta parte del contorno se realiza con la velocidad modalmente activa de la secuencia anterior, en el caso de que no se haya programado una palabra F en la secuencia de aproximación/retirada suaves.
- Avance programado F
  Este valor de avance es válido a partir de P_3 o bien P_2, en el caso de que no se haya programado FAD. Si no se ha programado un valor F en la secuencia de aproximación/ retirada suaves del contorno, entonces será válida la velocidad programada en la secuencia anterior.

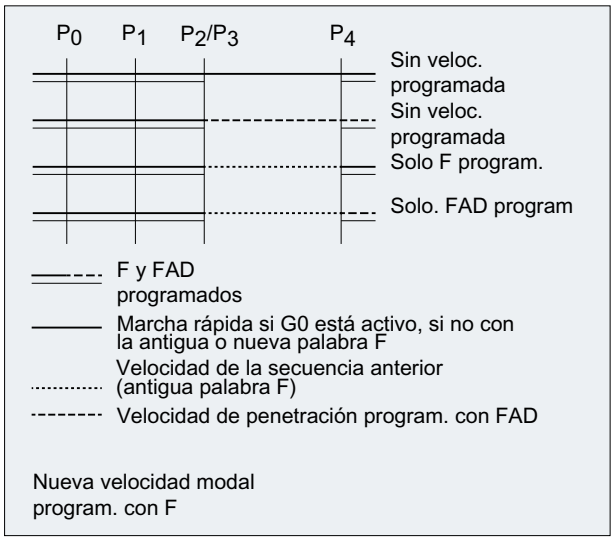
Ejemplo:

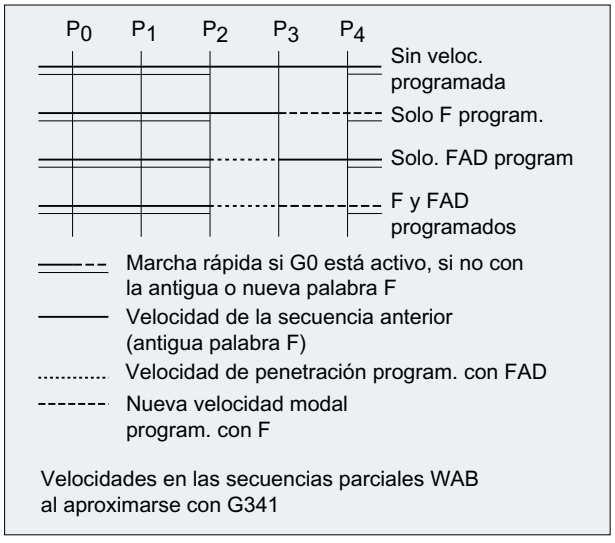
Código de programa	**Comentarios**
`$TC_DP1[1,1]=120`	`; Fresa T1/D1`
`$TC_DP6[1,1]=7`	`; Herramienta con un radio de 7 mm`
`N10 G90 G0 X0 Y0 Z20 D1 T1`	
`N20 G41 G341 G247 DISCL=AC(5) DISR=13 FAD 500 X40 Y-10 Z=0 F200`	
`N30 X50`	
`N40 X60`	
`...`	

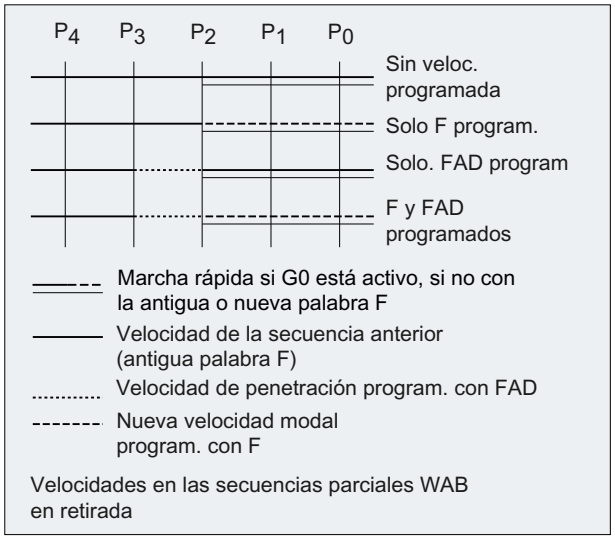


En la retirada, se invierten los avances activos modalmente de la secuencia anterior y del avance programado en la secuencia de aproximación y retirada suaves del contorno; es decir,

el propio contorno de retirada se realiza con el avance antiguo y el valor F programado servirá para la velocidad entre los puntos P_2 a P_0.







Lectura de posiciones

Los puntos P_3 y P_4 se pueden leer en la aproximación como variables de sistema en el WKS.

- $P_APR: Lectura de P
- $_3$ (punto inicial)
- $P_AEP: Lectura de P
- $_4$ (punto inicial del contorno)
- $P_APDV: Lectura para comprobar si $P_APR y $P_AEP contienen valores válidos

2.10.4.2 Aproximación y retirada con estrategias de retirada ampliada (G460, G461, G462)

En algunos casos geométricos especiales, al activar o desactivar la corrección del radio de la herramienta es necesario utilizar estrategias para la aproximación y retirada del contorno ampliadas frente a la realización anterior con vigilancia de colisión activada. Así, por ejemplo, una vigilancia de colisión puede tener el efecto de que un segmento del contorno no sea mecanizado por completo; ver la siguiente figura:

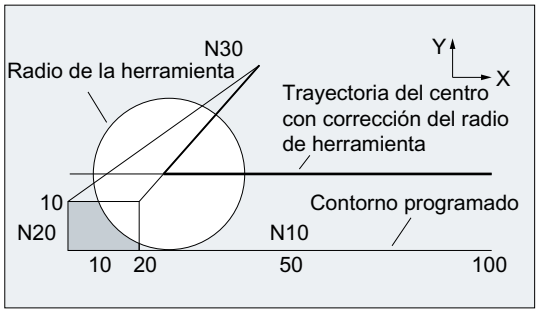


Figura 2-17 Comportamiento de retirada con G460

Sintaxis

```
G460
```

```
G461
G462
```

Descripción

G460:	Como hasta ahora (activación de la vigilancia de colisión para la secuencia de aproximación y retirada)
G461:	Insertar un círculo en secuencia con corrección del radio de corte/herramienta, cuando no sea posible punto de intersección, cuyo centro se encuentre en el punto final de la secuencia no corregida y cuyo radio sea igual al de la herramienta. Hasta el punto de intersección se mecaniza con **círculo auxiliar** alrededor del punto final del contorno (es decir, hasta el final del contorno).
G462:	Insertar una recta en una secuencia con corrección del radio de corte/herramienta cuando no sea posible un punto de intersección; la secuencia se prolonga mediante su tangente en el punto final (ajuste por defecto). Se mecaniza hasta la **prolongación** del último elemento de contorno (es decir, hasta poco antes del fin del contorno).

Nota

El comportamiento en la aproximación es simétrico al de retirada.

El comportamiento en la aproximación o retirada queda determinado por el estado del comando G en las secuencias de aproximación y retirada del contorno. El comportamiento en la aproximación se puede definir independientemente del comportamiento en la retirada.

Ejemplos

Ejemplo 1: comportamiento de retirada con G460

A continuación se representará siempre la situación correspondiente a la desactivación de la corrección del radio de herramienta. El comportamiento para la aproximación es totalmente análogo.

Código de programa	Comentarios
G42 D1 T1	; Radio de herramienta 20 mm
...	
G1 X110 Y0	
N10 X0	
N20 Y10	
N30 G40 X50 Y50	

Ejemplo 2: aproximación con G461

Código de programa	Comentarios
N10 $TC_DP1[1,1]=120	; Tipo de herramienta fresa
N20 $TC_DP6[1,1]=10	; Radio de la herramienta

Código de programa	Comentarios
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	

Información adicional

G461

Cuando no sea posible encontrar un punto de intersección entre la última secuencia con corrección del radio de corte/herramienta y la secuencia anterior, se prolonga la curva desplazada de ésa mediante una circunferencia cuyo centro coincida con el punto final de la secuencia no corregida y cuyo radio sea igual al radio de la herramienta.

El control intenta buscar un punto de intersección entre la circunferencia y la secuencia anterior.

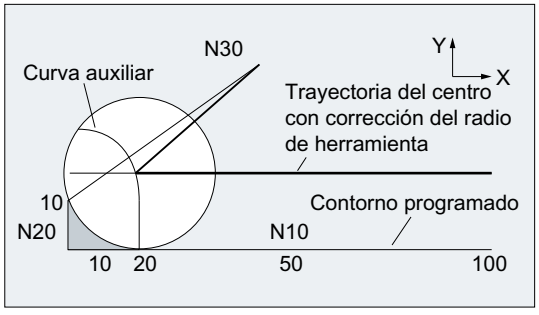


Figura 2-18 Comportamiento de retirada con G461

Vigilancia de colisión CDON, CDOF

Cuando no se encuentra punto de intersección estando activo CDOF (ver apartado: Vigilancia de colisión, CDON, CDOF), se interrumpe la búsqueda; es decir, no se comprueba que puedan existir otros puntos de intersección con otras secuencias anteriores.

Cuando está activo CDON se continúa con la búsqueda de otros puntos de intersección incluso cuando ya se haya encontrado uno.

Cuando se haya encontrado uno de estos puntos de intersección, éste será el nuevo punto final de la secuencia precedente y el punto inicial de la secuencia de desactivación. El círculo insertado solo sirve para calcular el punto de corte y no genera ningún desplazamiento.

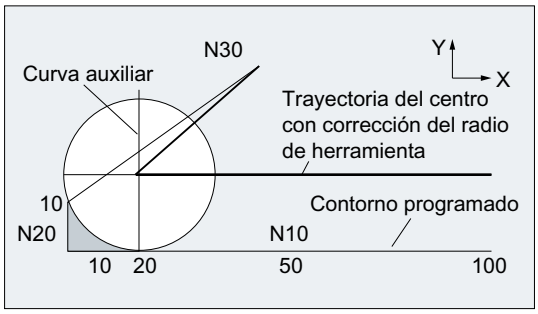
Nota

Si no se encuentra un punto de intersección, se genera la alarma 10751 (peligro de colisión).

G462

Cuando no sea posible encontrar un punto de intersección entre la última secuencia con corrección del radio de corte/herramienta y una secuencia anterior, al realizar la retirada con G462 (ajuste básico), se inserta una recta en el punto final de la última secuencia con corrección del radio de corte/herramienta (la secuencia se prolonga mediante su tangente en el punto final).

La búsqueda del punto de intersección transcurre igual que con G461.



Comportamiento de retirada con G462 (ver ejemplo)

Con G462 no se mecaniza el ejemplo del programa desde N10 y N20 en la esquina conformada de la manera que sería posible con la herramienta utilizada. Este comportamiento puede ser necesario cuando el contorno parcial (desviado del contorno programado), no deba de ser violado (en el ejemplo a la izquierda de N20) incluso para valores de y mayores que 10 mm.

Comportamiento en esquinas con KONT

Si está activo KONT (bordear contorno en el punto inicial o final), se distingue si el punto final está por delante o por detrás del contorno.

- **Punto final delante del contorno**
  Si el punto final se sitúa delante del contorno, el comportamiento de retirada es igual que con NORM. Esta característica tampoco cambia si la última secuencia de contorno con G451 se prolonga con una línea recta o un arco. No son necesarias estrategias adicionales de rodeo para evitar violaciones del contorno en las proximidades del punto final del contorno.
- **Punto final detrás del contorno**
  Si el punto final se encuentra detrás del contorno, dependiendo de G450/G451, se inserta una recta o un arco de circunferencia. Entonces G460 - G462 carecen de significado. Si, en esta situación, la última secuencia de desplazamiento no muestra ningún punto de intersección con una secuencia anterior, se puede producir entonces un punto de intersección con el elemento de contorno insertado o con el tramo recto del punto final de la circunferencia de desvío al punto final programado.
  Si el elemento de contorno insertado es un círculo (G450) y forma un punto de intersección con la secuencia anterior, éste es idéntico al punto de intersección que se produciría con NORM y G461. Generalmente queda por desplazar una porción adicional de la circunferencia. Para la parte lineal de la secuencia de desplazamiento no es necesario el cálculo del punto de intersección.
  En el segundo caso (cuando no se encuentra punto de intersección del elemento de contorno insertado con la secuencia precedente) se realiza el desplazamiento al punto de intersección entre la recta de retirada y una secuencia precedente.
  Por lo tanto, con G461 ó G462 activos, sólo se puede producir un comportamiento modificado frente a G460 si NORM está activo o si, por razones geométricas, el comportamiento con KONT es idéntico al de NORM.

2.10.5 Activación/desactivación de la vigilancia de colisión ("detección de cuellos de botella") (CDON, CDOF, CDOF2)

La vigilancia de colisión ("detección de cuellos de botella") con la corrección del radio de herramienta (WRK) activa se activa/desactiva en el programa CN con los comandos del grupo G 23.

Sintaxis

```
G41/G42 CDON
...
CDOF/CDOF2
```

Descripción

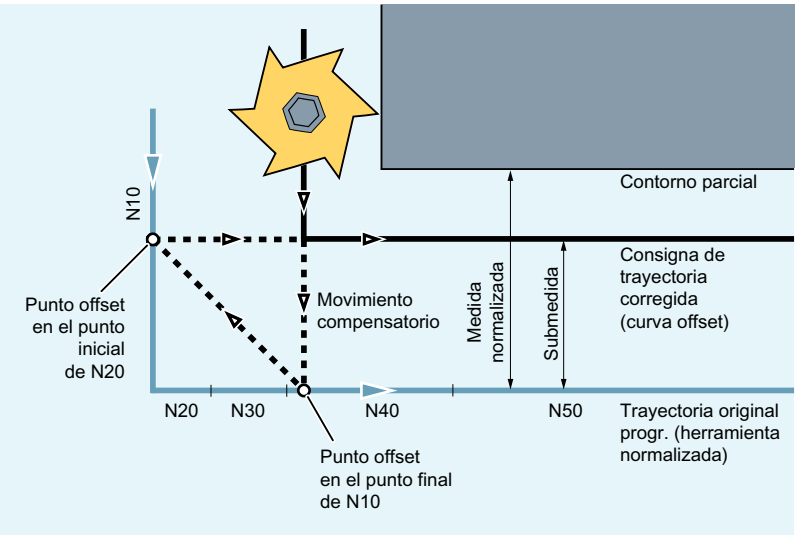
CDON:	Activar la vigilancia de colisión ("detección de cuellos de botella") Con CDON se comprueba en un número ajustable (MD20240) de secuencias si las trayectorias de herramienta de secuencias **no contiguas** se solapan. De esta forma se detectan por adelantado posibles colisiones que el control numérico puede evitar activamente.
CDOF:	Desactivar la vigilancia de colisión ("detección de cuellos de botella") Con CDOF, se busca un punto de intersección común para la secuencia actual en la secuencia de desplazamiento **anterior** (en esquinas interiores); si fuese necesario, se busca también en las secuencias que preceden a esta. Cuando se encuentra un punto de intersección, ya no se examinan más secuencias. En el caso de las esquinas exteriores, siempre se encuentra un punto de intersección entre dos secuencias consecutivas. **Nota:** CDOF ayuda a prevenir la detección incorrecta de "cuellos de botella" debidos, p. ej., a la falta de información en el programa CN.
CDOF2:	Desactivar la vigilancia de colisión para el fresado periférico 3D CDOF2 permite calcular la dirección de corrección de la herramienta a partir de segmentos de secuencia contiguos. CDOF2 solo actúa en el fresado periférico 3D y tiene el mismo significado que CDOF en todos los demás tipos de mecanizado (p. ej., el fresado frontal 3D).

Efecto de la vigilancia de colisión por medio de un ejemplo

El programa CN describe la trayectoria del centro de una herramienta normalizada. El contorno para una herramienta utilizada actualmente produce una tolerancia rebasada por defecto que, en la figura siguiente, se representa con una magnitud poco realista para ilustrar las condiciones geométricas.

Además, en el ejemplo se da por supuesto que el control abarca únicamente tres secuencias:

MD20240 $MC_CUTCOM_MAXNUM_CHECK_BLOCKS = 3



Dado que un punto de intersección existe únicamente entre las curvas offset de las dos secuencias N10 y N40, se deberían omitir las dos secuencias N20 y N30. En el ejemplo, el

control aún no conoce la secuencia N40 si N10 se tiene que ejecutar al final. De este modo se puede omitir una secuencia única.

Con CDOF2 activo, el movimiento compensatorio representado en la figura se ejecuta y no se detiene. En esta situación, un CDOF o CDON activo provocaría una alarma.

2.10.6 Corrección de herramienta en 2 1/2 D (CUT2D, CUT2DD, CUT2DF, CUT2DFD)

La corrección del radio de herramienta en 2½ D se utiliza para el mecanizado de planos inclinados cuando no se gira la orientación de la herramienta sino la **pieza**. La activación se realiza a través de los comandos `CUT2D`, `CUT2DD`, `CUT2DF` o `CUT2DFD`.

Corrección longitudinal de herramienta

La corrección longitudinal de la herramienta se incluye siempre en el cálculo en relación al plano de trabajo fijo, no rotado.

Corrección del radio de herramienta en 2½ D para herramientas de contornos

La corrección del radio de herramienta en 2½ D para herramientas de contornos se activa cuando se programa con `CUT2D`, `CUT2DD`, `CUT2DF` o `CUT2DFD` uno de los dos comandos `G41` (corrección del radio de herramienta a la izquierda del contorno) o `G42` (corrección del radio de herramienta a la derecha del contorno). Sirve para la selección automática de filos para herramientas con rotación no simétrica con las cuales se pueden mecanizar pieza a pieza segmentos de contorno individuales.

Nota

Cuando no está activa la corrección del radio de herramienta en 2½ D, una herramienta de contornos se comporta como una herramienta normal que consta únicamente del primer filo.

Corrección del radio de herramienta en 2½ D relativa a una herramienta diferencial

La corrección del radio de herramienta en 2½ D relativa a una herramienta diferencial se activa a través de los comandos `CUT2DD` o `CUT2DFD`. Debe utilizarse cuando el contorno programado se refiere a la trayectoria del centro de una herramienta diferencial y el mecanizado se realiza con otra herramienta distinta. En tal caso, en el cálculo de la corrección del radio de herramienta en 2½ D se incluirán solo el valor de desgaste del radio de la herramienta activa ($TC_DP_15) y el offset de corrección de herramienta OFFN (Página 257) y TOFFR (Página 85) en caso de haberse programado. El radio base ($TC_DP6) de la herramienta activa **no** se incluye en el cálculo.

Sintaxis

```
CUT2D
CUT2DD
CUT2DF
CUT2DFD
```

Descripción

CUT2D:	Activación de la corrección del radio en 2½ D
CUT2DD:	Activación de la corrección del radio en 2½ D relativa a una herramienta diferencial
CUT2DF:	Activación de la corrección del radio en 2½ D, corrección del radio de herramienta relativa al frame actual o al plano inclinado
CUT2DFD:	Activación de la corrección del radio en 2½ D relativa a una herramienta diferencial en cuanto al frame actual o al plano inclinado

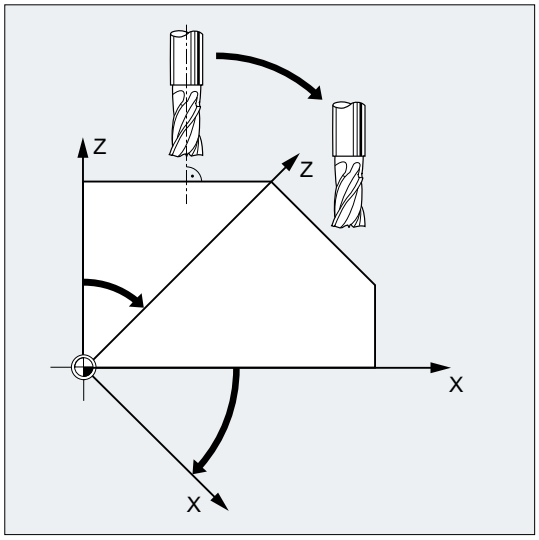
Información adicional

Herramientas de contorno

- Habilitación
  La habilitación de la corrección del radio de herramienta para herramientas de contorno se realiza de forma específica para cada canal a través de:
  MD28290 $MC_MM_SHAPED_TOOLS_ENABLE
- Tipo de herramienta
  Los tipos de herramientas de contorno se definen de forma específica para cada canal a través de:
  MD20370 $MC_SHAPED_TOOL_TYPE_NO
- Filos
  A cada herramienta de contornos se le puede asignar, en cualquier orden, un número de filos (números D). El número máximo de filos por herramienta se parametriza a través de:
  MD18106 $MN_MM_MAX_CUTTING_EDGE_PERTOOL
  El primer filo de una herramienta de contornos es el filo que se selecciona al activar la herramienta. Si, por ejemplo, en un programa se activa el quinto filo (D5) de la tercera herramienta (T3) a través de los comandos `T3 D5`, entonces D5 y los filos posteriores definen la herramienta de contornos en una parte o en su conjunto. Los filos anteriores a D5 se ignoran.

Corrección del radio de herramienta en 2½ D sin rotación del plano de corrección (CUT2D, CUT2DD)

Si se programa un frame con una rotación, en `CUT2D` o `CUT2DD`, el plano en el que tiene lugar la corrección del radio de herramienta (plano de corrección) **no acompaña el giro**. La corrección del radio de herramienta se incluye en el cálculo en relación con el plano de trabajo **no girado** (`G17`, `G18`, `G19`). La corrección longitudinal de herramienta es relativa al plano de corrección.

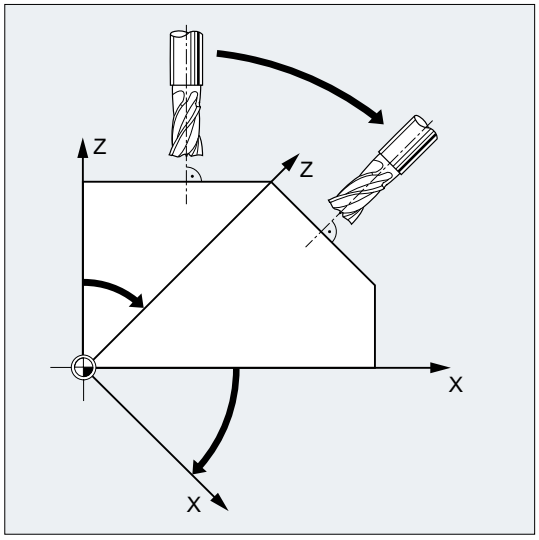


Para el mecanizado de planos inclinados deben definirse las correcciones de herramienta de acuerdo con dicho plano, o bien calcularse utilizando la función para "Corrección longitudinal de herramientas orientables".

Corrección del radio de herramienta en 2½ D con rotación del plano de corrección (CUT2DF, CUT2DFD)

Si se programa un frame con una rotación, en `CUT2DF` o `CUT2DFD`, el plano en el que tiene lugar la corrección del radio de herramienta (plano de corrección) **acompaña el giro**. La corrección del radio de herramienta se incluye en el cálculo en relación con el plano de trabajo **girado** (`G17`, `G18`, `G19`). Sin embargo, la corrección longitudinal de herramienta continúa siendo relativa al plano de trabajo **no girado**.

Requisitos: En la máquina, la herramienta debe poder ajustarse en perpendicular al plano de trabajo girado y debe ajustarse para el mecanizado.



Nota

La corrección longitudinal de herramienta continúa siendo activa con relación al plano de trabajo no girado.

Encontrará más información en el manual de funciones "Herramientas".

2.10.7 Mantener constante la corrección del radio de herramienta (CUTCONON, CUTCONOF)

La función "Mantener constante la corrección del radio de herramienta" sirve para suprimir la corrección del radio de herramienta para un cierto número de secuencias, manteniendo sin embargo como decalaje una diferencia, producida por la corrección del radio de herramienta en secuencias precedentes, entre la trayectoria programada y la trayectoria real recorrida del centro de la herramienta. Esa diferencia puede resultar ventajosa, p. ej., cuando se necesitan varias secuencias de desplazamiento en los puntos de inversión durante el fresado lineal, pero no se desean los contornos creados por la corrección del radio de herramienta (estrategias de rodeo). Se puede utilizar independientemente del tipo de corrección del radio de la herramienta ($2^1/_2$D, fresado frontal en 3D, fresado periférico en 3D).

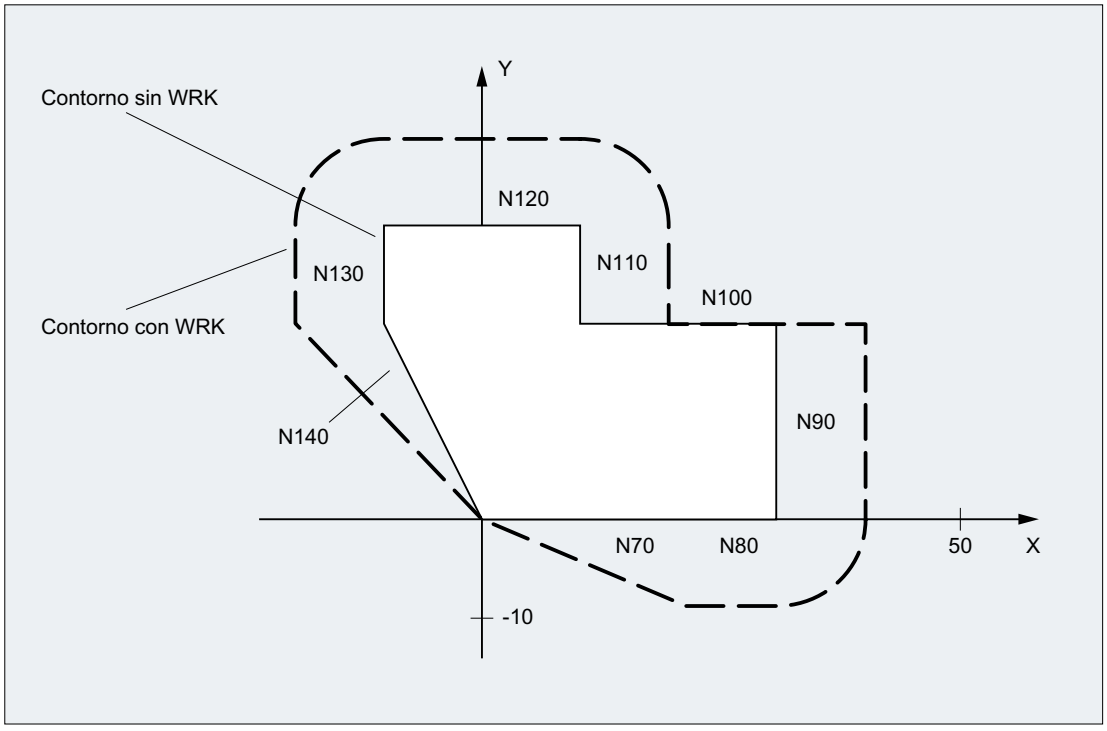
Sintaxis

```
CUTCONON
CUTCONOF
```

Descripción

`CUTCONON`:	Comando para la activación de la función "Mantener constante la corrección del radio de herramienta"
`CUTCONOF`:	Comando para la desactivación de la función "Mantener constante la corrección del radio de herramienta"

Ejemplo



Código de programa	Comentarios
N10	; Definición de la herramienta d1.
N20 $TC_DP1[1,1]=110	; Tipo
N30 $TC_DP6[1,1]= 10.	; Radio
N40	
N50 X0 Y0 Z0 G1 G17 T1 D1 F10000	
N60	
N70 X20 G42 NORM	
N80 X30	
N90 Y20	
N100 X10 CUTCONON	; Activación de la supresión de corrección.
N110 Y30 KONT	; Al desactivar la supresión de corrección, si es necesario, insertar circunferencia de rodeo.
N120 X-10 CUTCONOF	
N130 Y20 NORM	; Sin circunferencia de rodeo al desactivar la corrección del radio de herramienta.
N140 X0 Y0 G40	
N150 M30	

Informaciones adicionales

En caso normal, antes de la activación de la supresión de corrección se encuentra ya activada la corrección del radio de la herramienta, y sigue estando activa cuando se vuelve a desactivar la supresión de corrección. En la última secuencia de desplazamiento antes de `CUTCONON` se produce el desplazamiento al punto de offset en el punto final de la secuencia. Todas las secuencias siguientes en las que esté activa la supresión de corrección se desplazan sin corrección. Sin embargo, durante esta operación son desplazadas en torno al vector que va desde el punto final de la última secuencia de corrección hasta su punto de offset. El tipo de interpolación de estas secuencias (lineal, circular, polinómico) puede ser cualquiera.

La secuencia de desactivación de la supresión de corrección, es decir, la secuencia que contiene la función `CUTCONOF`, se corrige de forma normal. Empieza en el punto de offset del punto inicial. Entre el punto final de la secuencia precedente, es decir, de la última secuencia de desplazamiento programada con la función `CUTCONON` activa, y este punto se inserta una secuencia lineal.

Las secuencias circulares en las que el plano circular es perpendicular al plano de corrección (círculos verticales) se tratan como si en ellas estuviera programada la función `CUTCONON`. Esta activación implícita de la supresión de corrección se anula automáticamente en la primera secuencia de desplazamiento que contenga un movimiento de desplazamiento en el plano de corrección y que no sea un círculo de este tipo. Los círculos verticales de este tipo pueden aparecer sólo en el fresado circunferencial.

2.10.8 Herramientas con posición fija de filo

En herramientas con posición específica del filo (herramientas de torneado y rectificado, tipos de herramienta 400–599; ver el capítulo "Evaluación de signos de desgaste", se realiza el cambio de G40 a G41/G42 y viceversa como si se tratase de un cambio de herramienta. Esto conduce, si está activa una transformada (p. ej., TRANSMIT), a una parada de pretratamiento (parada de decodificación) y con esto eventualmente a desviaciones del contorno de pieza previsto.

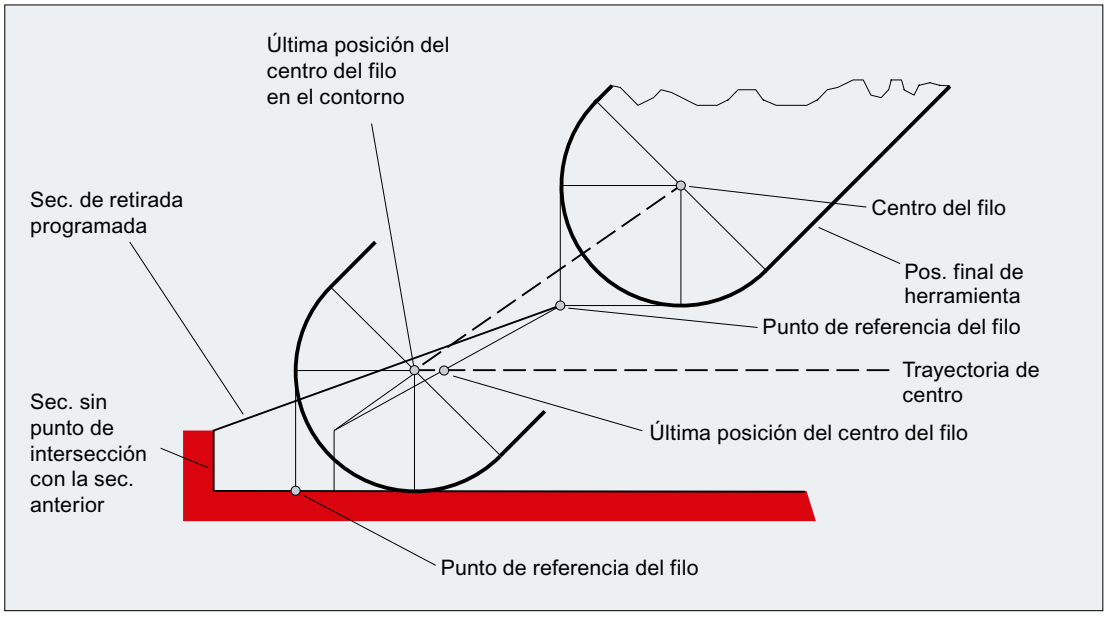
Esta funcionalidad original cambia con respecto a:

1. Parada de pretratamiento con TRANSMIT
2. Cálculo de puntos de intersección en la aproximación y retirada con KONT
3. Cambio de una herramienta con corrección del radio activada
4. Corrección del radio de corte/herramienta con orientación variable de la herramienta en la transformación

Información adicional

La función original ha cambiado de la siguiente manera:

- El cambio de G40 a G41/G42 y viceversa no se trata en adelante como un cambio de herramienta. Con TRANSMIT ya no se produce una parada de pretratamiento.
- Para el cálculo de los puntos de intersección con la secuencia de aproximación/retirada se utiliza la recta que une los puntos de definen el centro de corte al principio y al final de la secuencia. La diferencia entre el punto de referencia del corte y el punto medio del corte, se superpone en ese desplazamiento.
  En la aproximación y retirada con KONT (la herramienta recorre el punto de contorno; ver el apartado anterior "Aproximación y retirada del contorno"), la superposición se produce en la parte lineal del movimiento de aproximación o retirada. Por esta razón, las condiciones geométricas son idénticas en herramientas con o sin posición específica de filo. La diferencia entre el comportamiento anterior y el nuevo estriba en aquellos casos esporádicos en los que se genera un punto de intersección entre la secuencia de aproximación o retirada del contorno con una secuencia no consecutiva; ver figura siguiente:



- No se permite el cambio de una herramienta con corrección del radio activada, para la que se modifica la distancia entre centro del radio de corte/herramienta y el punto de referencia del filo, cuando se utilicen secuencias circulares y polinomios racionales de orden > 4 para secuencias de desplazamiento. Al contrario que en comportamientos anteriores, se puede realizar el cambio de herramienta al utilizar otros tipos de interpolación, incluso con transformaciones activas (p. ej., TRANSMIT).
- Cuando se utilice la corrección del radio de corte/herramienta con orientación variable de la herramienta ya no se podrá realizar la transformada del punto de referencia del filo al centro del radio de corte/herramienta mediante un sencillo decalaje de origen. Las herramientas con posición específica del filo de corte no se pueden utilizar para fresados periféricos 3D (alarma).

Nota

Para fresados frontales esto no es relevante, ya que en este caso sólo es posible utilizar tipos de herramienta sin posición específica del filo. (Las herramientas que no se puedan describir con uno de los tipos ya existentes se tratarán como herramientas de tipo fresa esférica con el radio indicado. Se ignora la posición del filo introducida).

2.11 Influencia sobre la trayectoria

2.11.1 Parada precisa (G60, G9, G601, G602, G603)

La parada precisa es un modo de desplazamiento en el que, al final de cada secuencia de desplazamiento, todos los ejes de contorneado y ejes adicionales que intervienen en el movimiento y que no se desplazan abarcando todas las secuencias se frenan hasta detenerse.

La parada precisa se utiliza para mecanizar esquinas exteriores agudas o para acabar esquinas interiores a la medida.

Con el criterio de parada precisa de establece con qué precisión debe aproximarse la esquina y cuándo debe conmutarse a la siguiente secuencia:

- "Parada precisa fina"
  El cambio de secuencia se realiza en cuanto se han alcanzado los límites de tolerancia para "parada precisa fina" específicos de todos los ejes que intervienen en el movimiento.
  "Parada precisa fina" se ajusta mediante: MD36010 $MA_STOP_LIMIT_FINE[<Eje>]
- "Parada precisa basta"
  El cambio de secuencia se realiza en cuanto se han alcanzado los límites de tolerancia para "parada precisa basta" específicos de todos los ejes que intervienen en el movimiento.
  "Parada precisa basta" se ajusta mediante: MD36000 $MA_STOP_LIMIT_COARSE[<Eje>]
- "Final de interpolador"
  El cambio de secuencia se realiza en cuanto el control ha calculado la velocidad de consigna cero para todos los ejes que intervienen en el movimiento. No se tienen en cuenta ni la posición real ni el error de seguimiento de los ejes implicados.

Sintaxis

```
G60 ...
G9 ...
G601/G602/G603 ...
```

Descripción

G60:	Comando para la activación de la parada precisa **modalmente** activa
G9:	Comando para la activación de la parada precisa válida **secuencia a secuencia**
G601:	Comando para la activación del criterio de parada precisa "**Parada precisa fina**"
G602:	Comando para la activación del criterio de parada precisa "**Parada precisa basta**"
G603:	Comando para la activación del criterio de parada precisa "**Final de interpolador**"

Nota

Los comandos para la activación de los criterios de parada precisa (G601/G602/G603) sólo actúan si G60 o G9 están activos.

Ejemplo

Código de programa	Comentarios
N5 G602	; Criterio "Parada precisa basta" seleccionado.
N10 G0 G60 Z...	; Parada precisa modalmente activa.
N20 X... Z...	; G60 sigue actuando.
...	
N50 G1 G601	; Criterio "Parada precisa fina" seleccionado.
N80 G64 Z...	; Conmutación al modo de contorneado.
...	
N100 G0 G9	; Parada precisa sólo activa para esta secuencia.
N110 ...	; Modo de contorneado nuevamente activo.

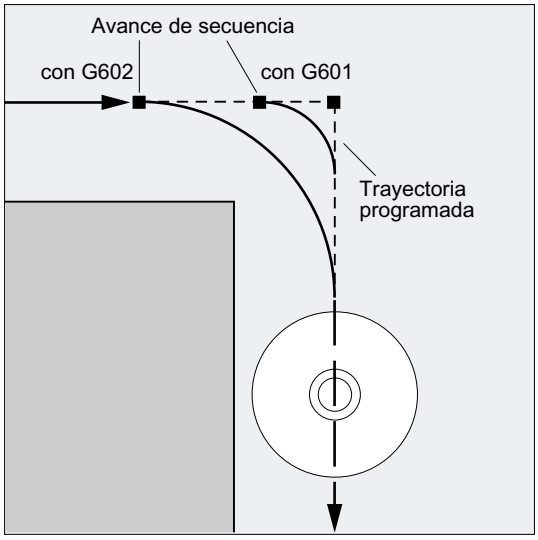
Información adicional

G60, G9

G9 origina la parada precisa en la secuencia actual, G60 origina la parada precisa en la secuencia actual y en todas las siguientes.

Con los comandos del modo de contorneado G64 o G641 - G645, se desactiva G60.

G601, G602



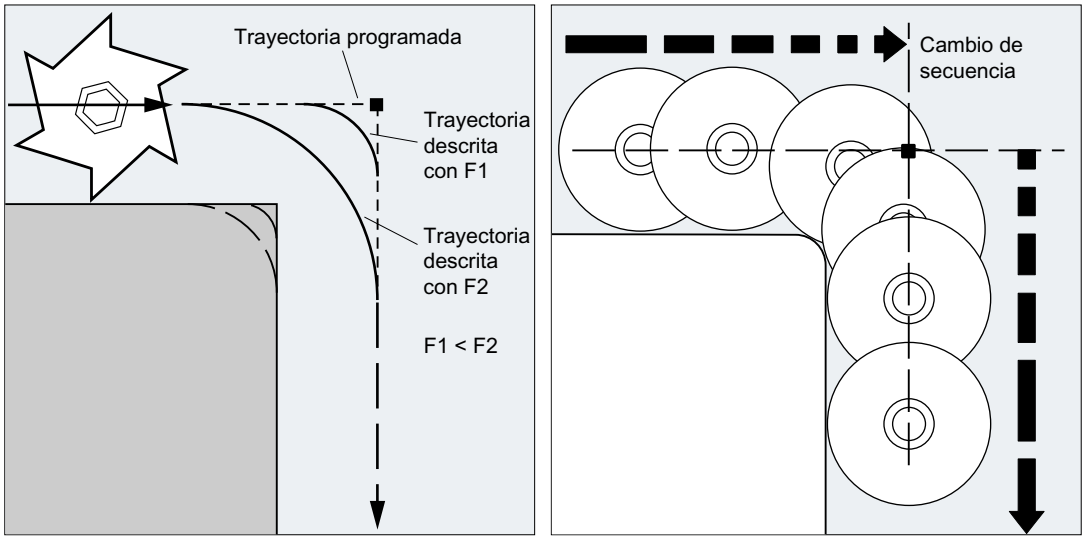
El desplazamiento va perdiendo velocidad y se detiene muy brevemente en la esquina.

Nota

Los límites para los criterios de parada precisa deben reducirse únicamente lo necesario. Cuánto más exiguos sean los límites, más durarán la compensación de posición y la aproximación a la posición de destino.

G603

El cambio de secuencia se realiza cuando el control ha calculado velocidad de consigna cero para los ejes implicados. En dicho instante la posición real de los ejes está retrasada en el valor de seguimiento (depende de la dinámica de la máquina y de la velocidad programada). Esto permite matar las esquinas exteriores.



Criterio de parada precisa configurado

Para G0 y los demás comandos del 1.er grupo G puede definirse de forma específica para el canal que, a diferencia del criterio de parada precisa programado, se utilice automáticamente un criterio predefinido (ver las indicaciones del fabricante de la máquina).

2.11.2 Modo de contorneado (G64, G641, G642, G643, G644, G645, ADIS, ADISPOS)

En este modo, la velocidad de contorneado no se reduce al final de la secuencia para cambiar la secuencia hasta una velocidad que permita alcanzar el criterio de parada precisa. El objetivo es evitar un frenado mayor de los ejes de contorneado en el punto de cambio de secuencia para, de este modo, cambiar a la siguiente secuencia con la mínima variación de velocidad posible. Para lograr este objetivo, cuando se selecciona el modo de contorneado se activa además la función "Control anticipativo de la velocidad (LookAhead)".

El modo de contorneado con matado de esquinas significa que se suavizan o se diseñan tangencialmente transiciones de secuencia con forma acodada mediante cambios locales del desarrollo programado.

El modo de contorneado produce:

- Un redondeo del contorno.
- Una reducción del tiempo de mecanizado gracias a la eliminación de los procesos de frenado y aceleración que se precisan para alcanzar el criterio de parada precisa.
- Mejores condiciones de corte gracias a una curva de velocidad más uniforme.

El modo de contorneado tiene sentido cuando:

- Se debe recorrer un contorno con la máxima rapidez posible (p. ej. con desplazamiento rápido).
- El desarrollo preciso en el marco de un criterio de error puede diferir del programado para generar un desarrollo continuo.

El modo de contorneado no tiene sentido cuando:

- Se debe recorrer un contorno con precisión.
- Se requiere una constancia absoluta de la velocidad.

Nota

El modo de contorneado se interrumpe con secuencias que provocan implícitamente una parada de decodificación previa, p. ej. mediante:

- Acceso a los datos de estado determinados de la máquina ($A...)
- Emisión de funciones auxiliares

Sintaxis

```
G64 ...
G641 ADIS=…
G641 ADISPOS=…
G642 ...
G643 ...
G644 ...
G645 ...
```

Descripción

`G64`:	Modo de contorneado con reducción de la velocidad según el factor de sobrecarga
`G641`:	Modo de contorneado con matado de esquinas según el criterio de recorrido
`ADIS=...`:	Criterio de recorrido con `G641` para las funciones de contorneado `G1`, `G2`, `G3`, …
`ADISPOS=...`:	Criterio de recorrido con `G641` para el desplazamiento rápido `G0`
	El criterio de recorrido (= distancia de matado de esquinas) `ADIS` o `ADISPOS` describe el recorrido que la secuencia de matado de esquinas debe comenzar antes del final de la secuencia, o bien el recorrido que la secuencia de matado de esquinas debe haber finalizado después del final de la secuencia. **Nota:** Si no se programa ningún `ADIS/ADISPOS`, se aplica el valor "cero" y, en consecuencia, el comportamiento de desplazamiento es igual que con `G64`. En recorridos cortos, la distancia de matado de esquinas se reduce automáticamente (hasta máx. un 36%).

G642:	Modo de contorneado con matado de esquinas respetando las tolerancias definidas En este modo, el matado de esquinas se realiza normalmente respetando la desviación de contorneado máxima permitida. No obstante, en lugar de la tolerancia específica de ejes, también puede optarse por una configuración que respete la desviación de contorno máxima (tolerancia de contorno) o la desviación angular máxima de la orientación de la herramienta (tolerancia de orientación). **Nota:** la ampliación de las tolerancias de contorno y de orientación existe sólo en sistemas con la opción "Interpolación polinómica".
G643:	Modo de contorneado con matado de esquinas respetando las tolerancias definidas (internas de la secuencia) Con G643, al contrario que para G642, no se forma una secuencia con matado de esquinas propia, sino que se insertan, específicamente para los ejes, movimientos de matado de esquinas internos de la secuencia. El recorrido de matado de esquinas para cada eje puede ser distinto.
G644:	Modo de contorneado con matado de esquinas con la máxima dinámica posible **Nota:** G644 no es posible con la transformación cinemática activa. Se conmuta a nivel interno a G642.
G645:	Modo de contorneado con matado de esquinas y transiciones de secuencia tangenciales respetando las tolerancias definidas G645 trabaja en esquinas del mismo modo que G642. En el caso de transiciones de secuencia tangenciales, con G645 sólo se forman secuencias con matado de esquinas si el desarrollo de la curvatura del contorno original presenta un salto en al menos un eje.

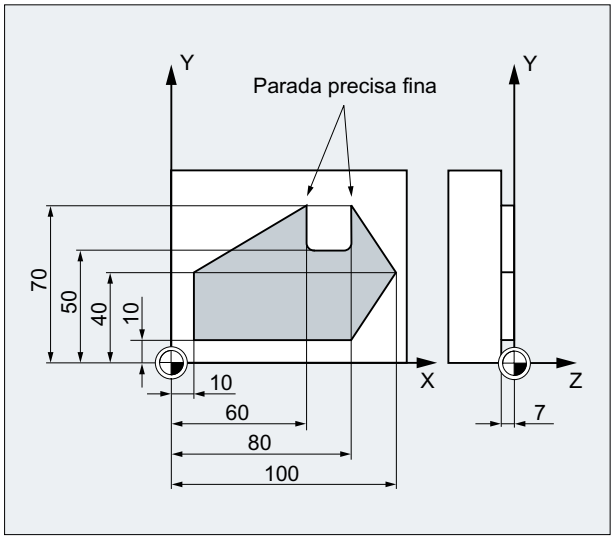
Nota

El matado de esquinas no sustituye al redondeo en esquinas (RND). El usuario no debe suponer el aspecto del contorno dentro de la zona de matado de esquinas. El tipo de matado de esquinas puede depender de aspectos dinámicos como, por ejemplo, la velocidad en la trayectoria. El redondeo en el contorno solo tiene sentido para pequeños valores de ADIS. Se tiene que utilizar RND cuando se precise un contorno definido en una esquina.

Nota

Si se interrumpe un movimiento de matado de esquinas generado por G641, G642, G643, G644 o G645, en el posterior reposicionado (REPOS) no se alcanza el punto de interrupción, sino el punto inicial o final de la secuencia de desplazamiento original (en función del modo REPOS).

Ejemplo



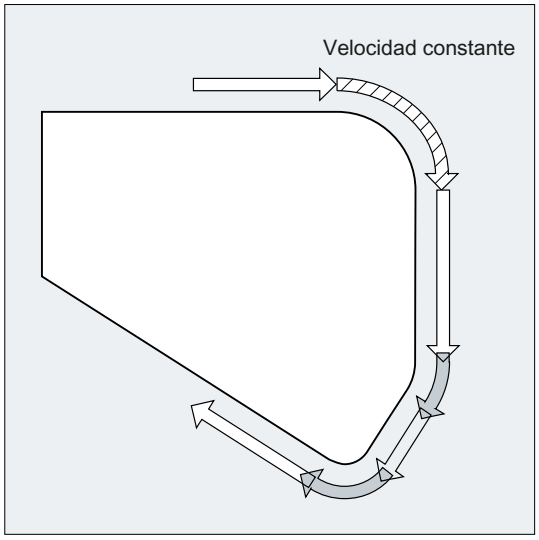
La aproximación a las dos esquinas exteriores en la ranura debe realizarse de forma exacta. De lo contrario, debe mecanizarse en el modo de contorneado.

Código de programa	**Comentarios**
N05 DIAMOF	; Acotado en radios.
N10 G17 T1 G41 G0 X10 Y10 Z2 S300 M3	; Desplazamiento a la posición inicial, activación del cabezal, corrección de la trayectoria.
N20 G1 Z-7 F8000	; Penetración de herramienta.
N30 G641 ADIS=0.5	; Matado de transiciones del contorno.
N40 Y40	
N50 X60 Y70 G60 G601	; Posicionado exacto con parada precisa fina.
N60 Y50	
N70 X80	
N80 Y70	
N90 G641 ADIS=0.5 X100 Y40	; Matado de transiciones del contorno.
N100 X80 Y10	
N110 X10	
N120 G40 G0 X-20	; Desactivar la corrección de trayectoria.
N130 Z10 M30	; Retirada de herramienta, fin del programa.

Información adicional

Modo de contorneado G64

En contorneado la herramienta se desplaza de forma tangencial a lo largo de transiciones de contorno con una velocidad lo más constante posible (sin frenados bruscos al final de las secuencias). Antes de esquinas y secuencias con parada precisa se frena de forma anticipativa (LookAhead).



Las esquinas también se rodean con velocidad constante. Para reducir los errores en las esquinas, disminuye la velocidad atendiendo a los límites de aceleración y a un factor de sobrecarga.

Nota

El redondeo de las aristas en contornos de transición depende del avance y del factor de sobrecarga. El factor de sobrecarga se puede ajustar en MD32310 $MA_MAX_ACCEL_OVL_FACTOR.

Definiendo MD20490 $MC_IGNORE_OVL_FACTOR_FOR_ADIS, las transiciones de secuencia se efectúan siempre independientemente del factor de sobrecarga ajustado.

Para evitar una parada no deseada en el movimiento interpolado (salida de la herramienta), se deberán tener en cuenta los siguientes puntos:

- Las funciones auxiliares que se activen después del final del desplazamiento o antes del siguiente desplazamiento interrumpen el contorneado (excepción: funciones auxiliares rápidas).
- Los ejes de posicionado se desplazan siempre intentando alcanzar la ventana de posicionado fino (como con G601). Si en una secuencia de programa de pieza hay que esperar a los ejes de posicionado, el modo de contorneado se interrumpe para los ejes de trayectoria.

Por el contrario, las secuencias intermedias sólo con comentarios, secuencias de cálculo o llamadas a subprogramas no repercuten en el contorneado.

Nota

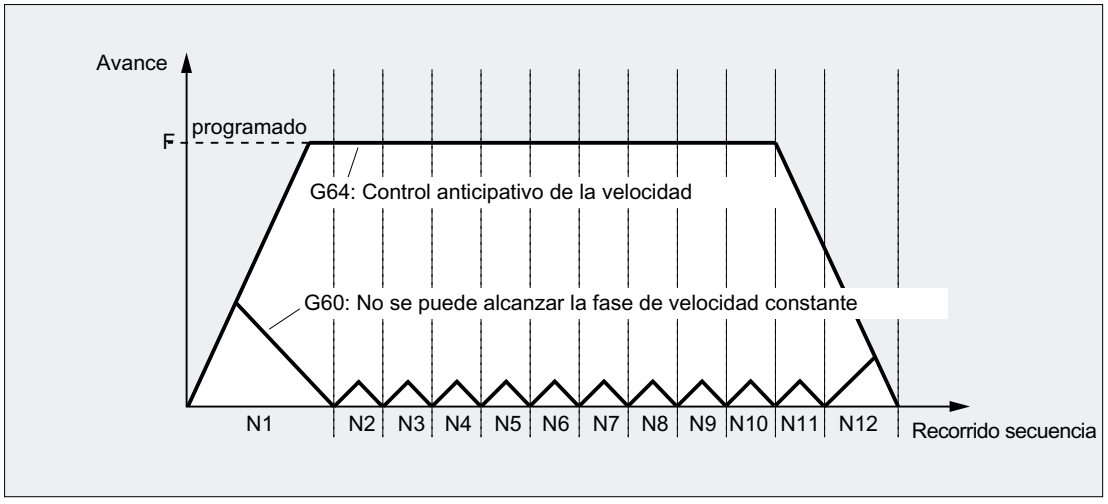
Si no todos los ejes de contorneado están contenidos en FGROUP, con frecuencia se producirá en transiciones de secuencia un salto de velocidad para los ejes no contenidos, el cual queda limitado por el control a través de la reducción de la velocidad en el cambio de secuencia al valor permitido por MD32300 $MA_MAX_AX_ACCEL y MD32310 $MA_MAX_ACCEL_OVL_FACTOR. Este frenado se puede evitar suavizando la relación de posición establecida de los ejes de contorneado mediante un matado de esquinas.

Control anticipativo de la velocidad LookAhead

En el modo de contorneado, el control determina automáticamente por anticipado la velocidad para varias secuencias CN. De este modo, se puede acelerar y frenar al concatenar varias secuencias si las transiciones son aproximadamente tangenciales.

Look Ahead resulta especialmente interesante para realizar programas con secuencias de desplazamiento cuyas trayectorias sean muy pequeñas y las velocidades de avance programadas sean elevadas.

El número de secuencias CN consideradas en el cálculo Look Ahead se puede definir mediante datos de máquina.



Modo de contorneado con matado de esquinas según el criterio de recorrido (G641)

El control inserta elementos de transición en transiciones de contorno cuando se utiliza G641. La distancia de matado de esquinas ADIS (o bien ADISPOS con G0) indica el grado máximo en que deben redondearse las esquinas. Dentro de la distancia de matado de esquinas, el control es libre de disolver la relación de contorneado y sustituirla por un recorrido dinámico óptimo.

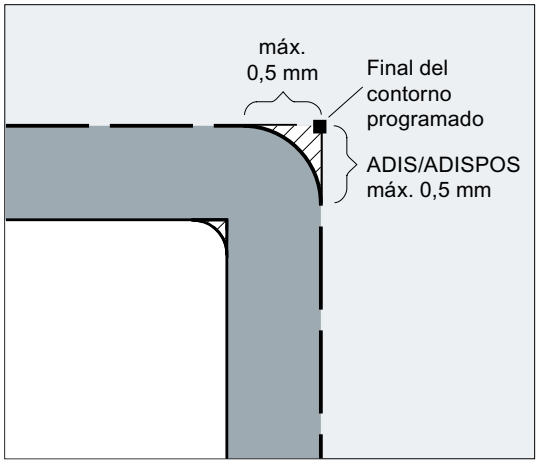
Desventaja: para todos los ejes se dispone solamente de un valor ADIS.

G641 actúa del mismo modo que RNDM, pero no está limitado a los ejes del plano de trabajo.

Al igual que G64, G641 trabaja con el control anticipativo de la velocidad LookAhead. Las secuencias con matado de esquinas de gran curvatura se realizan con velocidad reducida.

Ejemplo:

Código de programa	Comentarios
`N10 G641 ADIS=0.5 G1 X... Y...`	`; La secuencia con matado de esquinas no debe comenzar antes de 0,5 mm del final de la secuencia programada y debe haber finalizado 0,5 mm después del final de la secuencia programada. Este ajuste permanece activo de forma modal.`



Nota

El matado de esquinas no puede ni debe sustituir las funciones para el alisado definido (RND, RNDM, ASPLINE, BSPLINE, CSPLINE).

Matado de esquinas con precisión axial con G642

Con G642, el matado de esquinas no tiene lugar dentro del área ADIS definida, sino que se cumplen las tolerancias axiales definidas con MD33100 $MA_COMPRESS_POS_TOL. El recorrido de matado de esquinas viene determinado por el recorrido de matado de esquinas más corto de todos los ejes. Este valor se tiene en cuenta al generar una secuencia con matado de esquinas.

Matado de esquinas interno de la secuencia con G643

Las desviaciones máximas del contorno exacto en el matado de esquinas se definen con G643 a través del dato de máquina MD33100 $MA_COMPRESS_POS_TOL para cada eje.

Con G643 no se forma una secuencia con matado de esquinas propia, sino que se insertan, específicamente para los ejes, movimientos de matado de esquinas internos de la secuencia. Con G643, el recorrido de matado de esquinas de cada eje puede ser distinto.

Matado de esquinas con tolerancia de contorno y de orientación con G642/G643

Con MD20480 $MC_SMOOTHING_MODE, el matado de esquinas con G642 y G643 puede configurarse de modo que, en lugar de las tolerancias específicas de los ejes, estén activas una tolerancia de contorno y una tolerancia de orientación.

Las tolerancias de contorno y de orientación se ajustan en los datos de operador específicos del canal:

DO42465 $SC_SMOOTH_CONTUR_TOL (desviación máxima del contorno)

DO42466 $SC_SMOOTH_ORI_TOL (desviación angular máxima de la orientación de la herramienta)

Los datos de operador pueden programarse en el CN y, por tanto, definirse de forma distinta para cada transición de secuencia. Las especificaciones muy distintas para la tolerancia de contorno y la tolerancia de orientación sólo pueden actuar en G643.

Nota

La ampliación de las tolerancias de contorno y de orientación existe sólo en sistemas con la opción "Interpolación polinómica".

Nota

Para el matado de esquinas respetando la tolerancia de orientación debe estar activa una transformada de orientación.

Matado de esquinas con la máxima dinámica posible con G644

El matado de esquinas con la máxima dinámica posible se configura con MD20480 $MC_SMOOTHING_MODE en la cifra de millar:

Valor	Descripción
0	Especificación de las desviaciones axiales máximas con: MD33100 $MA_COMPRESS_POS_TOL
1	Especificación del recorrido máximo de matado de esquinas mediante la programación de: ADIS=... o ADISPOS=...
2	Especificación de las frecuencias máximas que se producen en cada eje en el área de matado con: MD32440 $MA_LOOKAH_FREQUENCY El área de matado se define de modo que, en el movimiento de matado, no se produzcan frecuencias que sobrepasen la frecuencia máxima establecida.
3	En el matado de esquinas con G644 no se vigilan la tolerancia ni la distancia de matado. Cada eje recorre una esquina con la máxima dinámica posible. Con SOFT se observan tanto la aceleración máxima como también el jerk máximo de cada eje. Con BRISK no se limita el jerk; cada eje se desplaza con la máxima aceleración posible.

Matado de esquinas de transiciones de secuencia tangenciales con G645

El movimiento de matado con G645 se define de modo que ningún eje implicado experimente un salto en la aceleración y no se sobrepasen las desviaciones máximas parametrizadas con respecto al contorno original (MD33120 $MA_PATH_TRANS_POS_TOL).

En el caso de transiciones de secuencia no tangenciales con forma acodada, el comportamiento de matado de esquinas es igual que G642.

Sin secuencias intermedias de matado de esquinas

En los siguientes casos no se inserta ninguna secuencia intermedia de matado de esquinas:

- Entre las dos secuencias se realiza una parada.
  Esto sucede cuando:
  - En la siguiente secuencia hay una emisión de función auxiliar antes del desplazamiento.
  - La siguiente secuencia no contiene desplazamiento.
  - En la siguiente secuencia se desplaza por primera vez un eje de contorneado que previamente haya sido un eje de posicionado.
  - En la siguiente secuencia se desplaza por primera vez un eje de posicionado, que previamente haya sido un eje de contorneado.
  - La secuencia anterior desplaza ejes geométricos y la secuencia siguiente no.
  - La secuencia siguiente desplaza ejes geométricos y la secuencia anterior no.
  - Antes del tallado de roscas, la siguiente secuencia tiene como condición G33 y la secuencia precedente, no.
  - Se conmuta entre BRISK y SOFT.
  - Los ejes implicados en la transformada no están asignados completamente al desplazamiento de la trayectoria (p. ej.: en vaivén, ejes de posicionado).
- La secuencia de matado de esquinas ralentizaría la ejecución del programa de pieza.
  Esto sucede:
  - Entre secuencias muy cortas.
    Dado que cada secuencia precisa por lo menos un ciclo de interpolador, la secuencia intermedia insertada multiplicaría por dos el tiempo de mecanizado.
  - Cuando se puede sobrepasar una transición de secuencia con G64 (modo de contorneado sin matado de esquinas) sin reducción de la velocidad.
    El redondeo provocaría un incremento del tiempo de ejecución. Es decir, el valor del factor de sobrecarga permitido (MD32310 $MA_MAX_ACCEL_OVL_FACTOR) influye en el hecho de si se efectúa o no un matado de esquinas en una transición de secuencias. El factor de sobrecarga solo se tiene en cuenta en el matado de esquinas con G641/G642. En el matado de esquinas con G643, el factor de sobrecarga no tiene ninguna influencia (este comportamiento también puede ajustarse para G641 y G642 seteando el MD20490 $MC_IGNORE_OVL_FACTOR_FOR_ADIS = TRUE).

- No se ha parametrizado el matado de esquina.
  Esto sucede cuando:
  - con G641 en secuencias G0, ADISPOS=0 (¡ajuste previo!).
  - con G641 en secuencias no G0, ADIS=0 (¡ajuste previo!).
  - con G641 en la transición entre G0 y no G0 o viceversa, prevalece el menor valor entre ADISPOS y ADIS.
  - con G642/G643, todas las tolerancias específicas de ejes son iguales a cero.
- La secuencia no contiene ningún movimiento de desplazamiento (secuencia cero).
  Esto sucede cuando:
  - Las acciones síncronas están activas.
    Normalmente, las secuencias cero son eliminadas por el intérprete. Sin embargo, si hay acciones síncronas activas, esta secuencia cero se encadena y se ejecuta. En este caso se produce una parada precisa conforme a la programación activa. De este modo, la acción síncrona recibe la posibilidad de conmutar si es necesario.
  - Se crean secuencias cero mediante saltos de programa.

Contorneado en rápido G0

Para el desplazamiento en rápido también se debe indicar una de las funciones G60/G9, G64 o G641 - G645 mencionadas. De lo contrario, son activos los ajustes estándar definidos mediante datos de máquina.

2.12 Transformaciones de coordenadas (frames)

2.12.1 Frames

Frame

Un frame es una regla de cálculo autocontenida que transforma un sistema de coordenadas cartesiano en otro sistema de coordenadas también cartesiano.

Frame básico (decalaje básico)

El frame básico describe la transformación de coordenadas del sistema de coordenadas básico (BKS) en el sistema de origen básico (BNS) y actúa como los frames ajustables.

Ver Sistema de coordenadas básico (BKS) (Página 38).

Frames ajustables

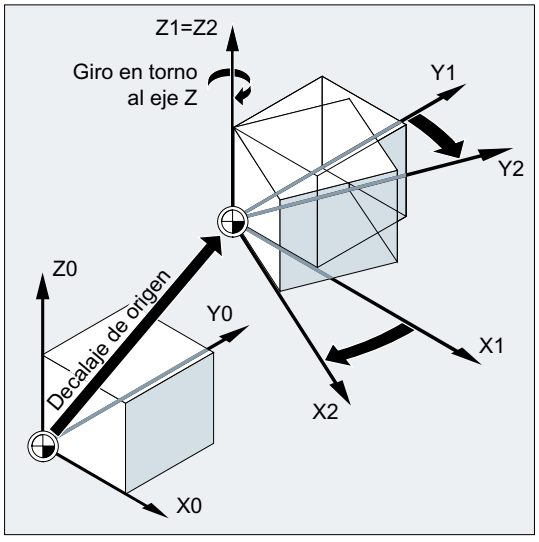
Los frames ajustables son los decalajes de origen ajustables que se pueden llamar desde cualquier programa CN con los comandos `G54` a `G57` y `G505` a `G599`. Los valores de decalaje están predefinidos por el usuario y se almacenan en la memoria de decalajes de origen del control numérico. Con ellos se define el sistema de origen ajustable (ENS).

Ver:

- Sistema de origen ajustable (ENS) (Página 40)
- Decalaje de origen ajustable (G54 ... G57, G505 ... G599, G53, G500, SUPA, G153) (Página 147)

Frames programables

A veces, dentro de un mismo programa CN es necesario o conveniente desplazar y, en caso necesario, girar, invertir en simetría especular o escalar el sistema de coordenadas de pieza seleccionado originalmente (o el "sistema de origen ajustable"). Esto se realiza mediante frames programables.



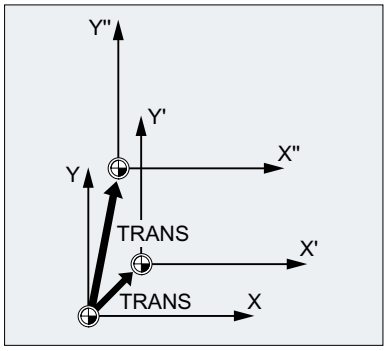
Ver Instrucciones frame (Página 316).

2.12.2 Instrucciones frame

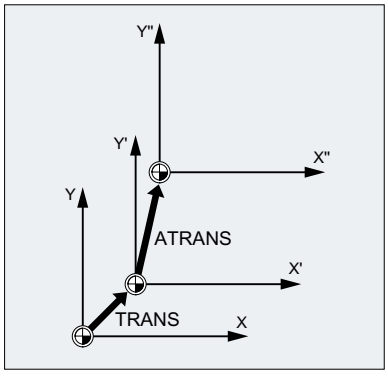
Función

Las instrucciones para los frames programables se refieren al programa CN actual. Actúan de forma aditiva o sustitutiva:

- Instrucción sustitutiva
  Borra todas las instrucciones frame anteriormente programadas. Como referencia se toma el último decalaje de origen ajustable (G54 ... G57, G505 ... G599) llamado.

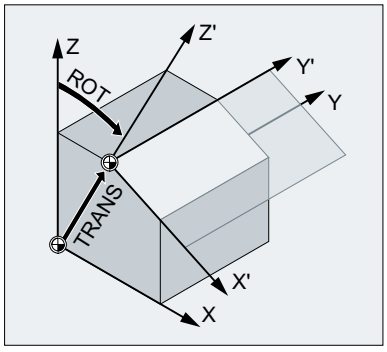


- Instrucción aditiva
  Se añade a frames existentes. Se toma como referencia el origen de pieza actualmente ajustado o bien el último origen de pieza programado con la instrucción frame.



Ejemplo de aplicación

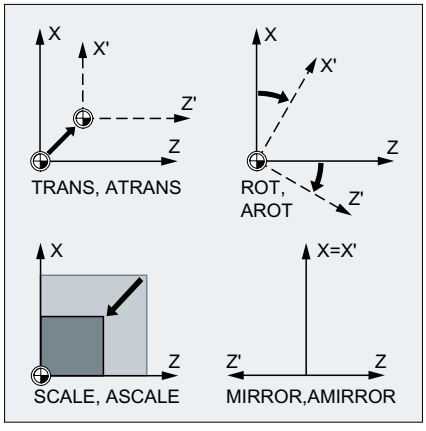
1. Decalaje de origen del sistema de coordenadas de pieza (WKS).
2. Rotación del sistema de coordenadas de pieza (WKS) para alinear un plano en paralelo con el plano de trabajo deseado.

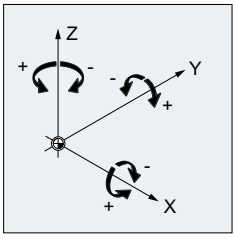


Sintaxis

Instrucciones sustitutivas	Instrucciones aditivas
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

Descripción



TRANS/ATRANS:	Decalaje del sistema de coordenadas de pieza en el sentido de los ejes geométricos indicados		
ROT/AROT:	Rotación del sistema de coordenadas de pieza: • mediante la concatenación de rotaciones individuales alrededor de los ejes geométricos indicados o bien • alrededor del ángulo RPL=... en el plano de trabajo actual (G17/G18/G19)		
	Sentido de giro:		
	Secuencia de giro:	con notación RPY:	Z, Y', X''
		con ángulo euleriano:	Z, X', Z''
	Rango de valores:	Los ángulos de giro sólo están definidos claramente en los siguientes rangos:	
		con notación RPY:	-180 ≤ x ≤ 180
			-90 < y < 90
			-180 ≤ z ≤ 180
		con ángulo euleriano:	0 ≤ x < 180
			-180 ≤ y ≤ 180
			-180 ≤ z ≤ 180
ROTS/AROTS:	Rotación del sistema de coordenadas de pieza definiendo ángulos espaciales La orientación de un plano en el espacio viene determinada de forma unívoca indicando un segundo ángulo espacial. Por este motivo, se puede programar un máximo de 2 ángulos espaciales: ROTS/AROTS X... Y... / Z... X... / Y... Z...		
CROTS:	CROTS actúa como ROTS, aunque se refiere al frame activo en el mantenimiento de datos.		
SCALE/ASCALE:	Escala en el sentido de los ejes geométricos indicados para aumentar/reducir un contorno		
MIRROR/AMIRROR:	Simetría del sistema de coordenadas de pieza mediante simetría (cambio de sentido) del eje geométrico indicado		
	Valor:	de libre elección (aquí: "0")	

Condiciones

- Las instrucciones frame deben programarse en una secuencia CN propia.
- Las instrucciones frame pueden ser utilizadas de forma individual o combinada.

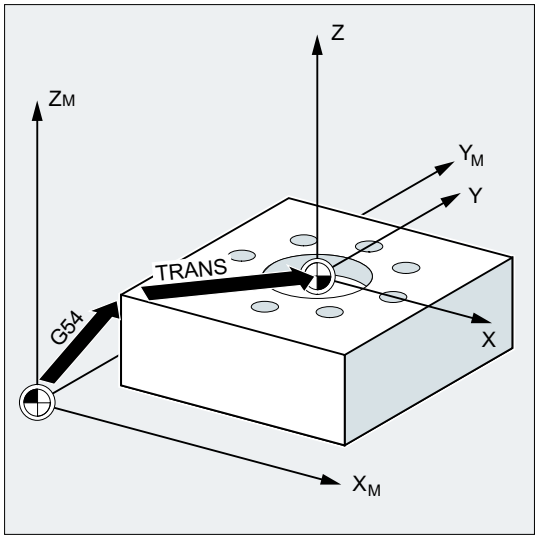
- Las instrucciones frame se ejecutan en el orden programado.
- en subprogramas se utilizan generalmente instrucciones aditivas. Si se programa la llamada al subprograma con el atributo SAVE entonces, al finalizarse la ejecución de dicho subprograma, se quedan activadas las instrucciones previamente definidas en el programa principal.

2.12.3 Decalaje de origen programable (TRANS, ATRANS)

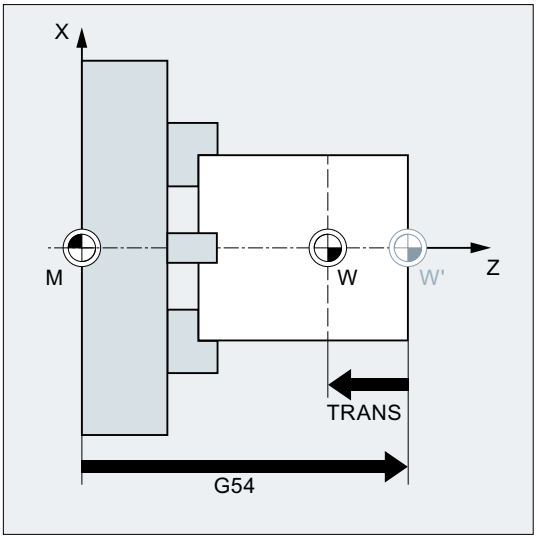
Con el comando TRANS se desplaza el WKS en modo absoluto, referido al ENS generado con un decalaje de origen ajustable (G54 ... G57, G505 ... G599).

Con el comando ATRANS se desplaza de modo aditivo el WKS generado con TRANS.

Fresado:



Torneado:



Sintaxis

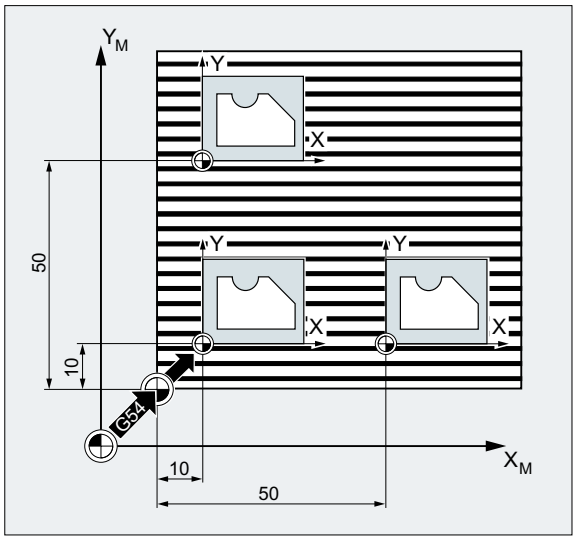
```
TRANS X… Y… Z…
ATRANS X… Y… Z…
```

Descripción

TRANS:	Decalaje absoluto del WKS, referido al origen de la pieza (ENS) ajustado con un decalaje de origen ajustable (G54 ... G57, G505 ... G599)	
	Único elemento de la secuencia:	sí
ATRANS:	Decalaje de origen aditivo del WKS, referido al origen de pieza ajustado con TRANS	
	Único elemento de la secuencia:	sí
X... Y... Z...:	Valores de decalaje en la dirección de los ejes geométricos indicados	

Ejemplos

Ejemplo 1: fresado



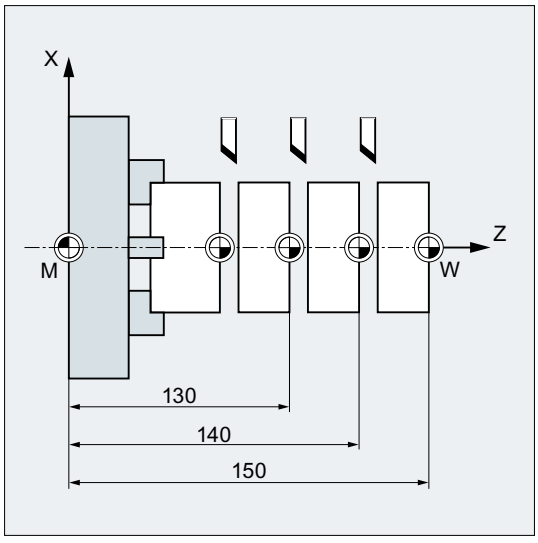
En esta pieza, los perfiles indicados aparecen varias veces en el mismo programa.

La secuencia de mecanizado para estos perfiles se programa en una subrutina.

El decalaje de origen se puede utilizar para definir los puntos de origen de la pieza necesarios y llamar al subprograma.

Código de programa	**Comentarios**
N10 G1 G54	; Plano de trabajo X/Y, origen de la pieza
N20 G0 X0 Y0 Z2	; Desplazar al punto inicial
N30 TRANS X10 Y10	; Decalaje absoluto
N40 L10	; Llamada de subprograma
N50 TRANS X50 Y10	; Decalaje absoluto
N60 L10	; Llamada de subprograma
N70 M30	; Fin del programa

Ejemplo 2: torneado



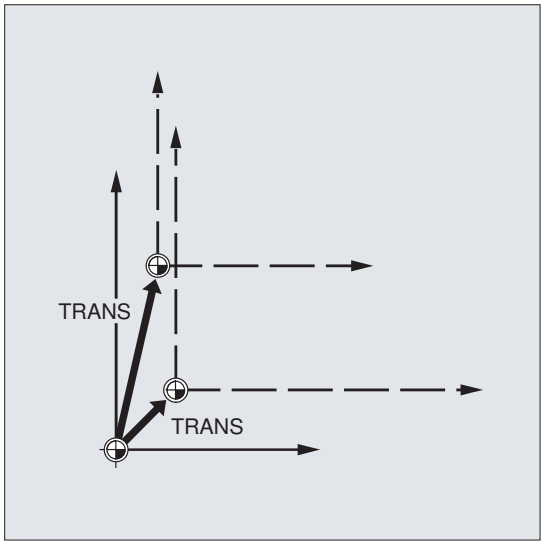
Código de programa	Comentarios
...	
N10 TRANS X0 Z150	; Decalaje absoluto
N15 L20	; Llamada de subprograma
N20 TRANS X0 Z140 (o ATRANS Z-10)	; Decalaje absoluto
N25 L20	; Llamada de subprograma
N30 TRANS X0 Z130 (o ATRANS Z-10)	; Decalaje absoluto
N35 L20	; Llamada de subprograma
...	

Información adicional

TRANS X... Y... Z...

Decalaje de origen a lo largo de los ejes indicados (ejes de trayectoria, síncronos y de posicionado). Como referencia se toma el último decalaje de origen ajustable (G54 ... G57, G505 ... G599) indicado.

ATENCIÓN
Sin frame de origen
El comando `TRANS` resetea todos los componentes del frame anteriormente programado.

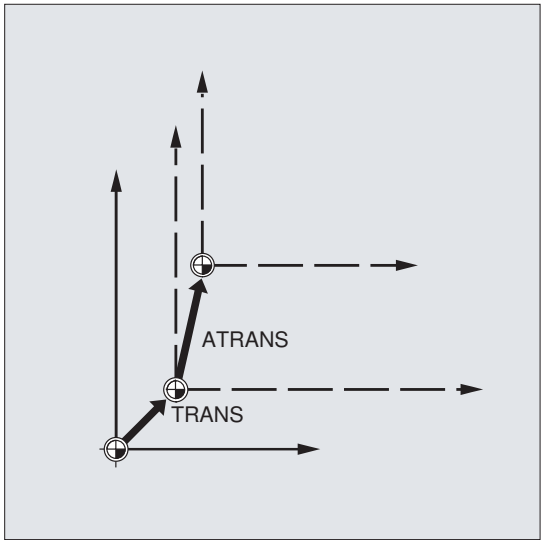


Nota

Para añadir un decalaje a los frames existentes, éste debe programarse con `ATRANS`.

ATRANS X... Y... Z...

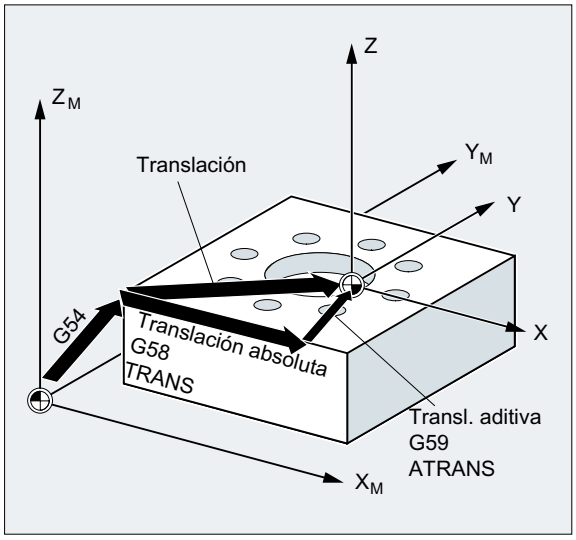
Decalaje de origen en la cantidad programada a lo largo de los ejes indicados. Se toma como referencia el origen actual o el último origen programado.



2.12.4 Decalaje de origen programable (G58, G59)

Con las funciones G58 y G59 se pueden sustituir por eje las partes de la transformación del decalaje de origen programable (TRANS/ATRANS) (Página 319):

- G58: parte absoluta de la transformación (decalaje basto)
- G59: parte aditiva de la transformación (decalaje fino)



Requisitos

Las funciones G58 y G59 sólo pueden utilizarse si está configurado el decalaje fino (MD24000 $MC_FRAME_ADD_COMPONENTS = 1).

Sintaxis

```
G58 <Eje_1><Valor_1> ... <Eje_3><Valor_3>
G59 <Eje_1><Valor_1> ... <Eje_3><Valor_3>
```

Descripción

G58:	G58 sustituye la parte absoluta de la transformación del decalaje de origen programable para los ejes indicados; se mantiene el decalaje programado aditivo. Como referencia se toma el último decalaje de origen ajustable (G54 ... G57, G505 ... G599) llamado.
	Único elemento de la secuencia: sí
G59:	G59 sustituye la parte aditiva de la transformación del decalaje de origen programable para los ejes indicados; se mantiene el decalaje programado absoluto.
	Único elemento de la secuencia: sí
<Eje_n>:	Eje geométrico en el canal
<Valor_n>:	Valores de decalaje en la dirección del eje geométrico indicado

Ejemplo

```
Código de programa          Comentarios
...
N50 TRANS X10 Y10 Z10       ; Parte absoluta de la transformación X10 Y10 Z10
N60 ATRANS X5 Y5            ; Parte aditiva de la transformación X5 Y5
                            → Decalaje total: X15 Y15 Z10
N70 G58 X20                 ; Parte absoluta de la transformación X20
                            → Decalaje total X25 Y15 Z10
N80 G59 X10 Y10             ; Parte aditiva de la transformación X10 Y10
                            → Decalaje total X30 Y20 Z10
...
```

Información adicional

La parte absoluta de la transformación (**decalaje basto**) se modifica mediante los siguientes comandos:

- TRANS
- G58
- CTRANS
- CFINE
- $P_PFRAME[X,TR]

La parte aditiva de la transformación (**decalaje fino**) se modifica mediante los siguientes comandos:

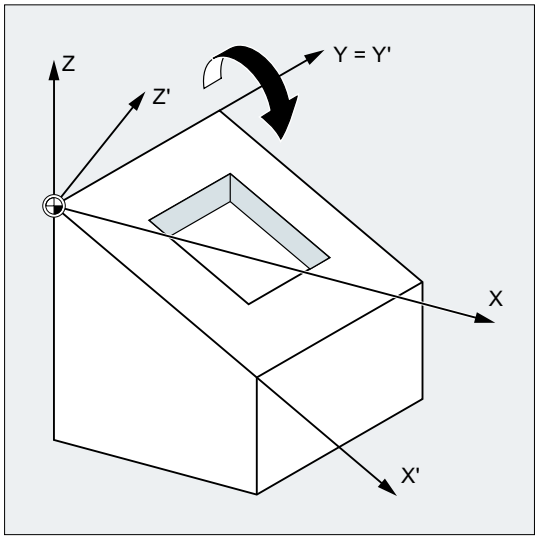
- ATRANS
- G59
- CTRANS
- CFINE
- $P_PFRAME[X,FI]

Ejemplos

Comando	Decalaje basto V_C	Decalaje fino V_F
TRANS X10	V_C = 10	Sin cambios
G58 X10	V_C = 10	Sin cambios
$P_PFRAME[X,TR]=10	V_C = 10	Sin cambios
ATRANS X10	Sin cambios	$V_F = V_F + 10$
G59 X10	Sin cambios	V_F = 10
$P_PFRAME[X,FI]=10	Sin cambios	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 Rotación programable (ROT, AROT, RPL)

Los comandos ROT/AROT permiten girar el sistema de coordenadas de pieza en el espacio. Los comandos se refieren exclusivamente al frame programable $P_PFRAME.



Sintaxis

```
ROT <1. GeoAx><Ángulo> <2.° GeoAx><Ángulo> <3.er GeoAx><Ángulo>
ROT RPL=<ángulo>
AROT <1. GeoAx><Ángulo> <2.° GeoAx><Ángulo> <3.er GeoAx><Ángulo>
AROT RPL=<ángulo>
```

Nota

Las rotaciones del sistema de coordenadas de pieza se efectúan mediante ángulos eulerianos.

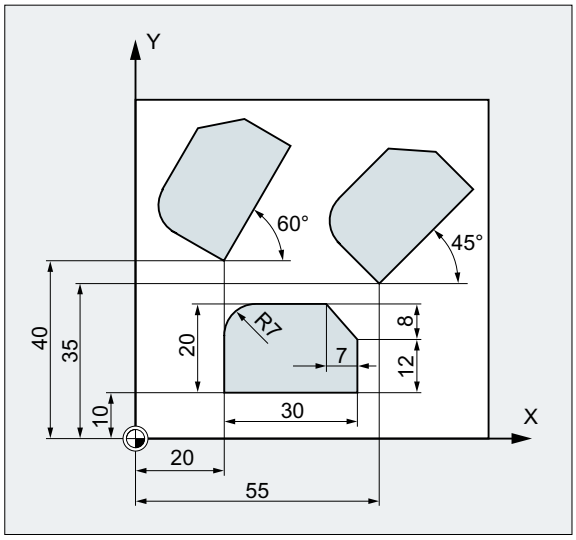
Descripción

ROT:	Rotación absoluta	
	Frame de referencia:	frame programable $P_PFRAME
	Punto de referencia:	origen del sistema de coordinadas de pieza actual establecido con G54 ... G57, G505 ... G599
AROT:	Rotación aditiva	
	Frame de referencia:	frame programable $P_PFRAME
	Punto de referencia:	origen del sistema de coordinadas de pieza actual establecido con G54 ... G57, G505 ... G599
<n-te GeoAx>:	Identificador del enésimo eje geométrico alrededor del cual debe girarse el ángulo indicado. Para ejes geométricos no programados se establece implícitamente el valor 0° como ángulo de giro.	

RPL:	rotación alrededor del eje geométrico perpendicular al plano activo (G17, G18, G19) con el ángulo indicado	
	Frame de referencia:	frame programable $P_PFRAME
	Punto de referencia:	origen del sistema de coordinadas de pieza actual establecido con G54 ... G57, G505 ... G599
<Ángulo>	Especificación del ángulo en grados	
	Rango de valores:	-360° ≤ ángulo ≤ 360°

Ejemplos

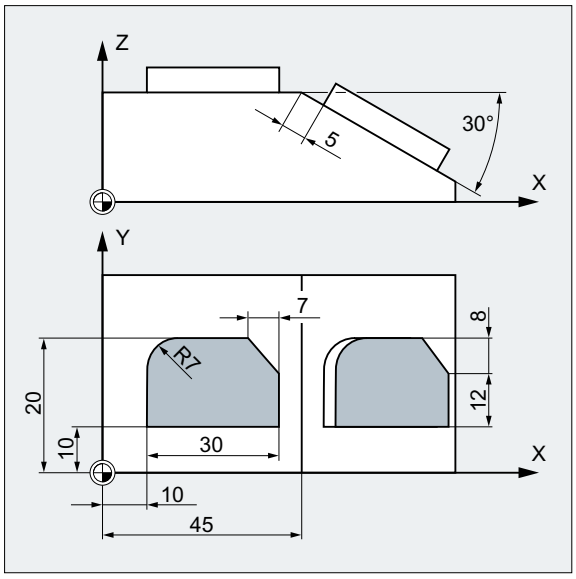
Ejemplo 1: Rotación en el plano G17



En esta pieza, los perfiles indicados aparecen varias veces en el mismo programa. Se deben realizar giros además de decalajes ya que las geometrías no se encuentran paralelas a los ejes.

Código de programa	**Comentarios**
N10 G17 G54	; Plano de trabajo X/Y, origen de la pieza
N20 TRANS X20 Y10	; Decalaje absoluto
N30 L10	; Llamada de subprograma
N40 TRANS X55 Y35	; Decalaje absoluto
N50 AROT RPL=45	; Rotación aditiva en torno al eje Z perpendicular
	; al plano G17, de 45°
N60 L10	; Llamada de subprograma
N70 TRANS X20 Y40	; Decalaje absoluto
	; (resetea todos los decalajes anteriores)
N80 AROT RPL=60	; Rotación aditiva en torno al eje Z perpendicular
	; al plano G17, de 60°
N90 L10	; Llamada de subprograma
N100 G0 X100 Y100	; Retirar
N110 M30	; Fin del programa

Ejemplo 2: Rotación en el espacio alrededor del eje Y

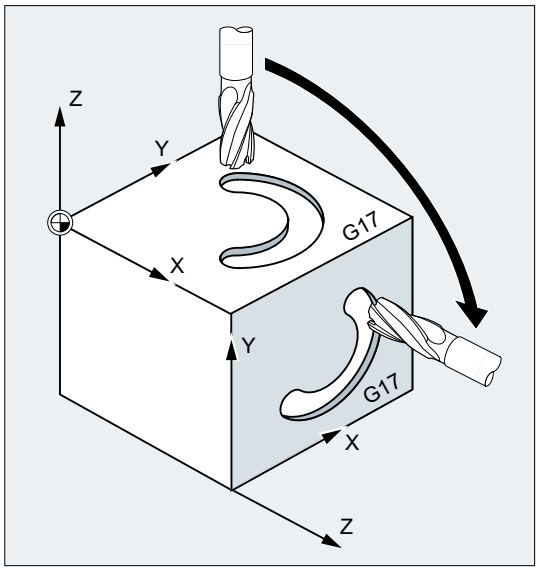


En este ejemplo se debe mecanizar con el mismo amarre de pieza la misma geometría en distintos planos.

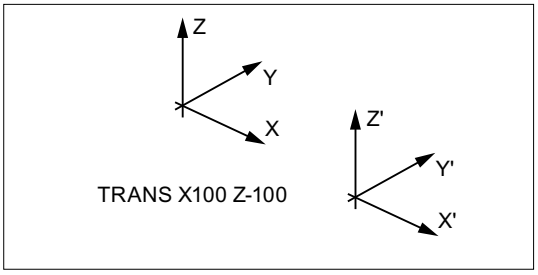
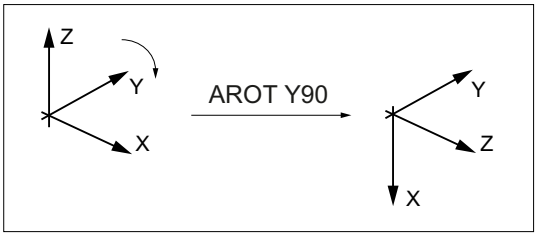
Requisitos:
la herramienta se debe posicionar perpendicular al plano inclinado en la dirección del nuevo eje Z.

Código de programa	**Comentarios**
N10 G17 G54	; Plano de trabajo X/Y, origen de la pieza
N20 TRANS X10 Y10	; Decalaje absoluto
N30 L10	; Llamada de subprograma
N40 ATRANS X35	; Decalaje aditivo
N50 AROT Y30	; Rotación aditiva alrededor del eje Y
N60 ATRANS X5	; Decalaje aditivo
N70 L10	; Llamada de subprograma
N80 G0 X300 Y100 M30	; Retirar, fin de programa

Ejemplo 3: Mecanizado de varios lados



En este ejemplo se mecaniza la misma geometría en dos planos de la pieza perpendiculares entre sí, haciendo uso de subprogramas. En el nuevo sistema de coordenadas de la superficie de trabajo derecha, la dirección de penetración, el plano de trabajo y el origen son idénticos al plano de trabajo superior. De este modo se siguen aplicando las condiciones necesarias para la ejecución del subprograma: Plano de trabajo `G17`, plano de coordenadas X/Y, dirección de penetración Z.

Código de programa	**Comentarios**
`N10 G17 G54`	`; Plano de trabajo X/Y, origen de la pieza`
`N20 L10`	`; Llamada de subprograma`
`N30 TRANS X100 Z-100`	`; Decalaje absoluto del WKS`
	
`N40 AROT Y90`	`; Rotación aditiva del WKS en torno a Y de 90°`
	

Código de programa	Comentarios
N50 AROT Z90	; Rotación aditiva del WKS en torno a Z de 90°
N60 L10	; Llamada de subprograma
N70 G0 X300 Y100 M30	; Retirar, fin de programa

Información adicional

Rotación en el plano activo

En la programación mediante RPL=..., el WKS gira en torno al eje perpendicular al plano activo.

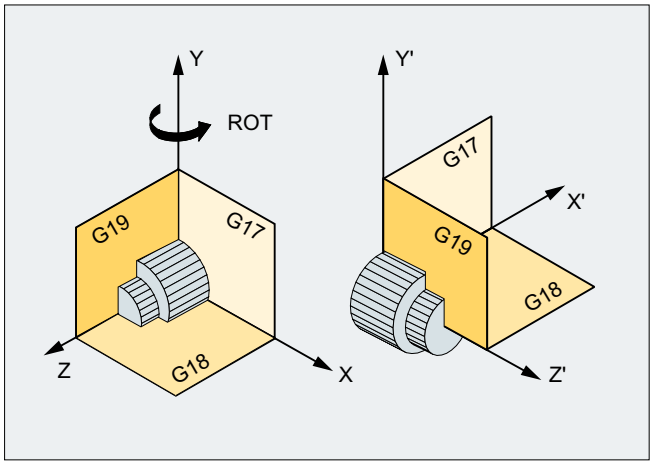


Figura 2-19 Rotación alrededor del eje Y o en el plano G18

⚠ ADVERTENCIA

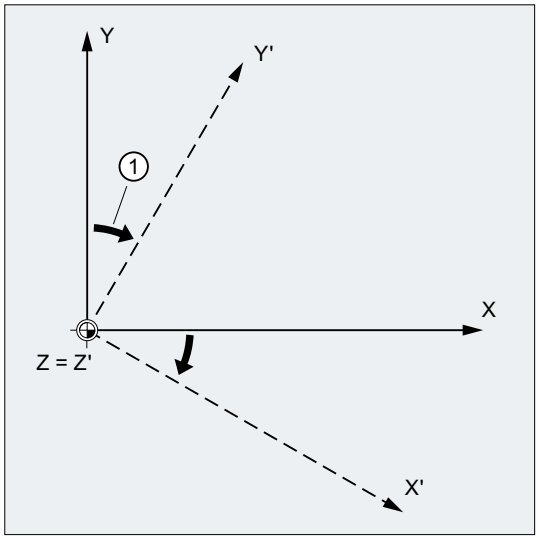
Cambio de plano

Si después de una rotación se programa un cambio de plano (G17, G18, G19), los ángulos de giro actuales de cada eje se mantienen y actúan también sobre el nuevo plano. Por eso, se recomienda encarecidamente resetear el ángulo de giro actual a 0 antes de un cambio de plano:

- N100 ROT X0 Y0 Z0 ; programación de ángulos explícita
- N100 ROT ; programación de ángulos implícita

Rotación absoluta con ROT X... Y... Z...

El WKS se gira sobre el ángulo programado alrededor de los ejes indicados.

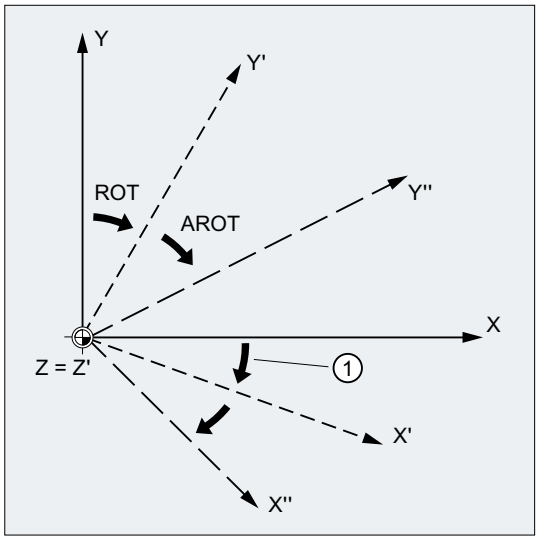


① Ángulo de giro

Figura 2-20 Rotación absoluta alrededor del eje Z

Rotación aditiva con AROT X... Y... Z...

El WKS se gira el ángulo programado alrededor de los ejes indicados.



① Ángulo de giro

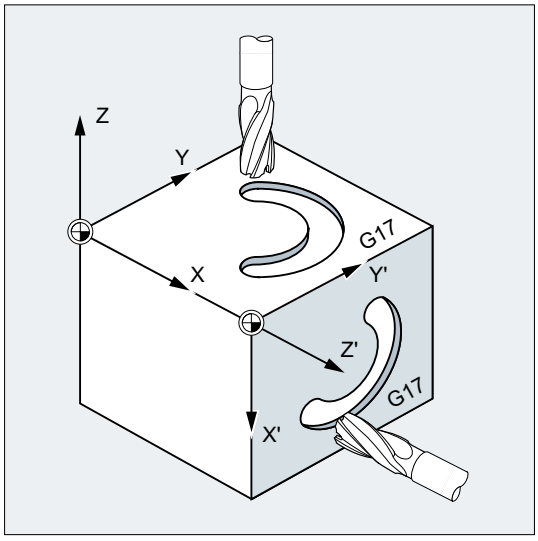
Figura 2-21 Rotación absoluta y aditiva alrededor del eje Z

Rotación del plano de trabajo

Con una rotación mediante `ROT`/`AROT`, el plano de trabajo gira también (`G17`, `G18`, `G19`).

Ejemplo: Plano de trabajo G17
El WKS está en la superficie superior de la pieza. El sistema de coordenadas se desplaza a una de las superficies laterales de la pieza mediante decalaje y rotación. El plano de trabajo G17 también gira. De esta forma pueden seguir programándose desplazamientos en el plano G17 a través de X e Y y las profundidades de penetración a través de Z.

Requisito:
La herramienta debe encontrarse perpendicularmente al plano de trabajo y el sentido positivo del eje de penetración aleja la herramienta de la pieza.
Mediante la introducción de `CUT2DF` se valida la corrección del radio de herramienta en el plano girado.



2.12.6 Rotaciones de frames progr. con áng. espaciales (ROTS, AROTS, CROTS)

Con los comandos `ROTS`, `AROTS` y `CROTS` pueden especificarse rotaciones del sistema de coordenadas de pieza en ángulos espaciales. Los ángulos espaciales son aquellos que forman las líneas de intersección del plano deseado girado en el espacio con los planos principales del sistema de coordenados aún no girado.

Nota

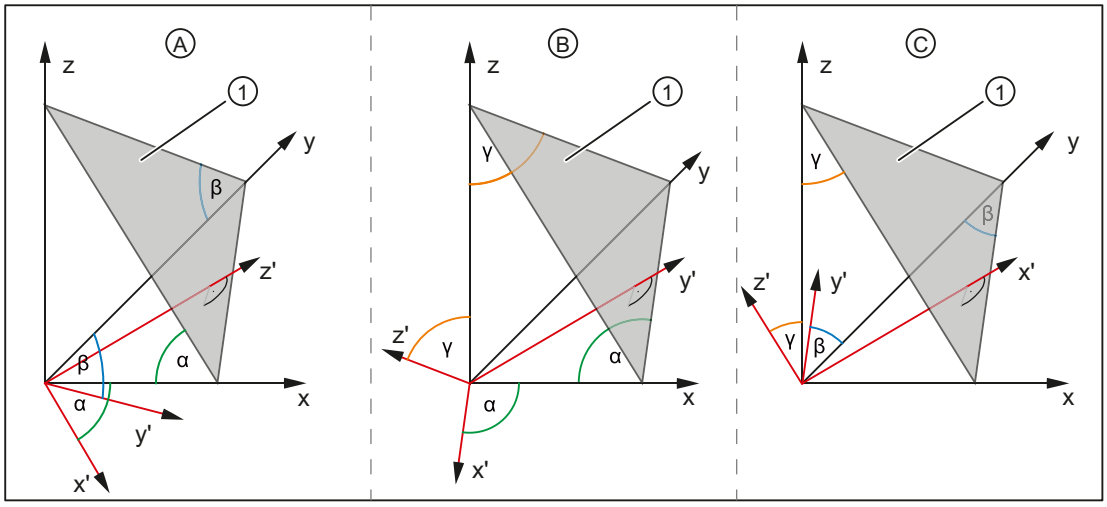
Identificador de eje geométrico

Por ejemplo, para la posterior descripción se aplica la definición siguiente:

- 1.er eje geométrico: X
- 2.º eje geométrico: Y
- 3.er eje geométrico: Z

La programación `ROTS` `X`α `Y`β ocasiona, por ejemplo, una alineación del plano G17 del WKS en paralelo al plano inclinado representado (como se muestra en la siguiente figura). La posición del origen del WKS permanece inalterada.

La orientación del WKS girado se establece de forma que el primer eje rotado se encuentra en el plano que forma éste con el 3.er eje del sistema de coordenadas original. En el ejemplo: X' se encuentra en el plano X/Z original.



① Plano inclinado

α, β, γ Ángulo espacial

A Alineación del plano G17 paralelo al plano inclinado:

- 1.ª rotación
  Rotación de x alrededor de y el ángulo α ⇒
  Eje x' paralelo al plano inclinado
- 2.ª rotación
  Rotación de y' alrededor de x' el ángulo β ⇒
  Eje y' paralelo al plano inclinado
  ⇒ eje z' perpendicular al plano inclinado
  ⇒ G17 paralelo al plano inclinado

B Alineación del plano G18 paralelo al plano inclinado:

- 1.ª rotación
  Rotación de z alrededor de x el ángulo γ ⇒
  eje z' paralelo al plano inclinado
- 2.ª rotación
  Rotación de x' alrededor de z' el ángulo α ⇒
  Eje x' paralelo al plano inclinado
  ⇒ eje y' perpendicular al plano inclinado
  ⇒ G18 paralelo al plano inclinado

C Alineación del plano G19 paralelo al plano inclinado:

- 1.ª rotación
  Rotación de y alrededor de z el ángulo β ⇒
  Eje y' paralelo al plano inclinado
- 2.ª rotación
  Rotación de z' alrededor de y' el ángulo γ ⇒
  eje z' paralelo al plano inclinado
  ⇒ eje x' perpendicular al plano inclinado
  ⇒ G19 paralelo al plano inclinado

Sintaxis

Definiciones

La posición de un plano en el espacio viene determinada de forma unívoca por dos ángulos espaciales. La indicación de un tercer ángulo especial haría que el plano estuviera sobredefinido. Por tanto, no está permitida.

Si se programa un único ángulo espacial, la rotación del WKS se produce de forma idéntica a `ROT`, `AROT` (ver capítulo "Rotación programable (ROT, AROT, RPL) (Página 325)").

A través de los dos ejes programados se define un plano según las posiciones de plano para `G17`, `G18` y `G19`. De esta forma se define el orden de los ejes de coordenadas (1.er eje/2.º eje del plano) o el orden de las rotaciones con el ángulo espacial:

Plano	1.er eje	2.º eje
G17	X	Y
G18	Z	X
G19	Y	Z

Alineación del plano G17 ⇒ ángulo espacial para X e Y

- 1.ª rotación: X alrededor de Y el ángulo α
- 2.ª rotación: Y alrededor de X' el ángulo β
- Orientación: X' se encuentra en el plano Z/X original.

```
ROTS  X<α> Y<β>
AROTS X<α> Y<β>
CROTS X<α> Y<β>
```

Alineación del plano G18 ⇒ ángulo espacial para Z y X

- 1.ª rotación: Z alrededor de X el ángulo γ
- 2.ª rotación: X alrededor de Z' el ángulo α
- Orientación: Z' se encuentra en el plano Y/Z original.

```
ROTS  Z<γ> X<α>
AROTS Z<γ> X<α>
CROTS Z<γ> X<α>
```

Alineación del plano G19 ⇒ ángulo espacial para Y y Z

- 1.ª rotación: Y alrededor de Z el ángulo β
- 2.ª rotación: Z alrededor de Y' el ángulo γ
- Orientación: Y' se encuentra en el plano X/Y original.

```
ROTS  Y<β> Z<γ>
AROTS Y<β> Z<γ>
CROTS Y<β> Z<γ>
```

Descripción

ROTS:	Rotaciones de frame con ángulos espaciales absolutos, frame de referencia: frame programable $P_PFRAME
AROTS:	Rotaciones de frame con ángulos espaciales aditivos, frame de referencia: frame programable $P_PFRAME
CROTS:	Rotaciones de frame con ángulos espaciales absolutos, frame de referencia: frame programado $P_...
X, Y, Z:	Identificador de eje geométrico (ver nota anterior: identificador de eje geométrico)
α, β, γ:	Ángulo espacial referido al eje geométrico correspondiente: • α → X • β → Y • γ → Z

2.12.7 Factor de escala programable (SCALE, ASCALE)

Con SCALE/ASCALE se pueden programar factores de escala de ampliación o reducción para todos los ejes de contorneado, ejes síncronos y ejes de posicionado en la dirección de los ejes respectivamente indicados. De este modo se pueden considerar en la programación formas geométricas similares o medidas de contracción distintas.

Sintaxis

```
SCALE X… Y… Z…
ASCALE X… Y… Z…
```

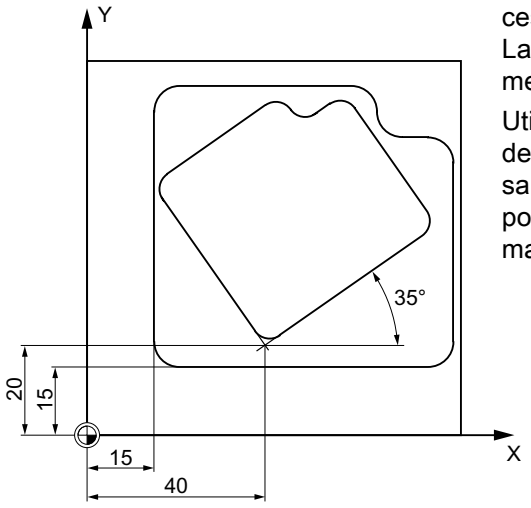
Nota

Las instrucciones frame se programan siempre en una secuencia CN propia.

Descripción

SCALE:	Ampliación/reducción absolutas, referidas al sistema de coordenadas actualmente válido ajustado con G54 ... G57, G505 ... G599
ASCALE:	Ampliación/reducción aditivas referidas al sistema de coordenadas actualmente ajustado o programado
X… Y… Z…:	Factores de escala en la dirección de los ejes geométricos indicados

Ejemplo



En esta pieza se repiten las dos cajas dos veces pero con diferente orientación y tamaño. La secuencia de mecanizado de dicha geometría se memoriza en un subprograma.

Utilizar el decalaje de origen y la rotación para definir los puntos de origen de la pieza necesarios, y la escala para reducir el contorno y posteriormente volver a llamar al subprograma.

Código de programa	Comentarios
N10 G17 G54	; Plano de trabajo X/Y, origen de la pieza
N20 TRANS X15 Y15	; Decalaje absoluto
N30 L10	; Mecanizar caja grande
N40 TRANS X40 Y20	; Decalaje absoluto
N50 AROT RPL=35	; Rotación de 35° en el plano
N60 ASCALE X0.7 Y0.7	; Factor de escala para la caja pequeña
N70 L10	; Mecanizar caja pequeña
N80G0 X300 Y100 M30	; Retirar, fin de programa

Información adicional

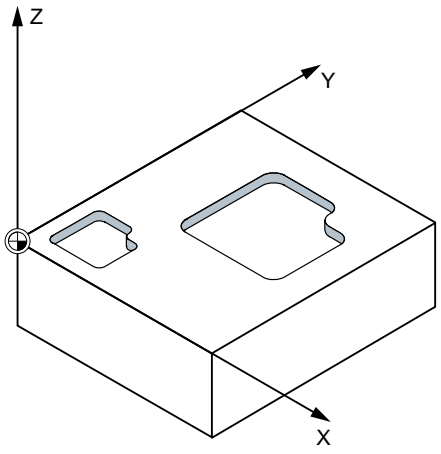
SCALE X... Y... Z...

Se puede especificar un factor de escala para la ampliación/reducción de cada uno de los ejes. La escala se refiere al sistema de coordenadas de pieza ajustado con G54 ... G57, G505 ... G599.

ATENCIÓN

Sin frame de origen

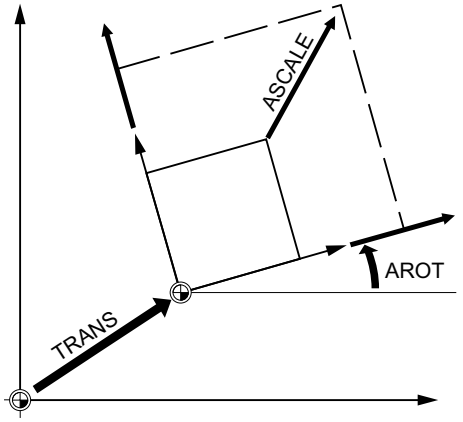
El comando `SCALE` resetea todos los componentes del frame anteriormente programado.



ASCALE X... Y... Z...

Se puede añadir un cambio de escala a los frames existentes con la programación de `ASCALE`. En este caso se multiplica el último factor de escala introducido por el nuevo valor.

Como referencia para el cambio de escala se utiliza el sistema de coordenadas actual o el último sistema de coordenadas programado.



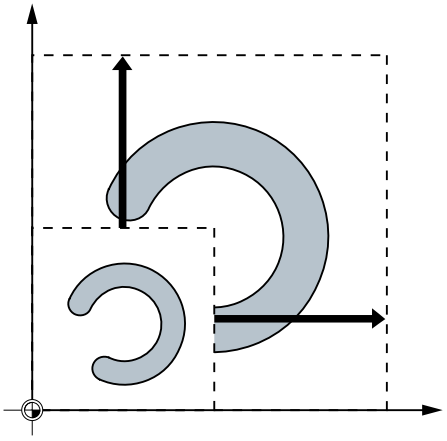
Escala y decalaje

Nota

Si se programa un decalaje con `ATRANS` después de programar `SCALE`, los valores de decalaje también se ven afectados por el factor de escala.

Factores de escala diferentes

ATENCIÓN
Peligro de colisión
¡Cuidado cuando haya factores de escala diferentes! Las interpolaciones circulares solamente se pueden ampliar o reducir con los mismos factores de escala.



Nota

Sin embargo, se pueden utilizar distintos factores de escala para programar elipses.

2.12.8 Simetría programable (MIRROR, AMIRROR)

Con `MIRROR`/`AMIRROR` se pueden mecanizar geometrías simétricas respecto a los ejes de coordenadas. Todos los desplazamientos programados posteriormente, p. ej., en el subprograma, se ejecutan en la imagen simétrica.

Sintaxis

```
MIRROR X... Y... Z...
AMIRROR X... Y... Z...
```

Nota

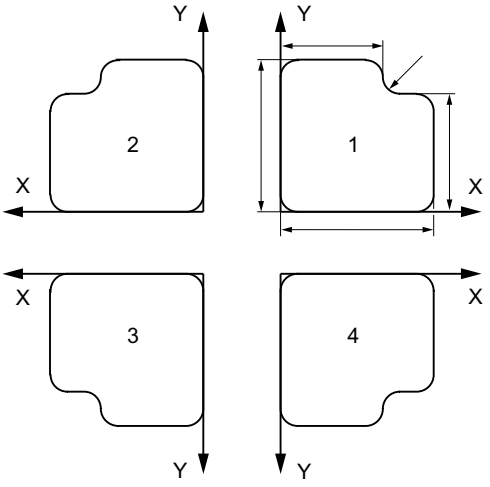
Las instrucciones frame se programan siempre en una secuencia CN propia.

Descripción

MIRROR:	Simetría absoluta, referida al sistema de coordenadas actualmente válido ajustado con G54 ... G57, G505 ... G599
AMIRROR:	Simetría aditiva referida al sistema de coordenadas ajustado actualmente o programado
X... Y... Z...:	Eje geométrico para el que se desea cambiar el sentido. El valor aquí indicado puede seleccionarse libremente, p. ej., X0 Y0 Z0.

Ejemplos

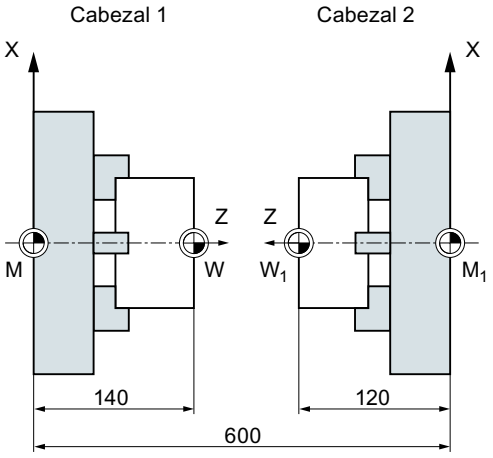
Ejemplo 1: fresado



El contorno aquí indicado se programa una sola vez como subprograma. Los otros tres contornos se crean por simetría. El origen de pieza se ha definido en el centro de los contornos.

Código de programa	**Comentarios**
N10 G17 G54	; Plano de trabajo X/Y, origen de la pieza
N20 L10	; Mecanizar el primer contorno superior derecho
N30 MIRROR X0	; Simetría del eje X (en X cambia el sentido)
N40 L10	; Mecanizar el segundo contorno superior izquierdo
N50 AMIRROR Y0	; Simetría del eje Y (en Y cambia el sentido)
N60 L10	; Mecanizar el tercer contorno inferior izquierdo
N70 MIRROR Y0	; MIRROR resetea los frames anteriores. Simetría del eje Y (en Y cambia el sentido)
N80 L10	; Mecanizar el cuarto contorno inferior derecho
N90 MIRROR	; Desactivar simetrías
N100 G0 X300 Y100 M30	; Retirar, fin de programa

Ejemplo 2: torneado



El mecanizado propiamente dicho se guarda como subprograma y la ejecución en el cabezal en cuestión se realiza mediante simetrías y decalajes.

Código de programa	**Comentarios**
`N10 TRANS X0 Z140`	`; Decalaje de origen a W`
`...`	`; Mecanizado de la 1.ª cara con el cabezal 1`
`N30 TRANS X0 Z600`	`; Decalaje de origen al cabezal 2`
`N40 AMIRROR Z0`	`; Simetría del eje Z`
`N50 ATRANS Z120`	`; Decalaje de origen a W1`
`...`	`; Mecanizado de la 2.ª cara con el cabezal 2`

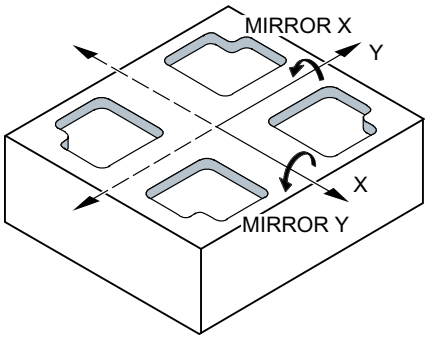
Información adicional

MIRROR X... Y... Z...

La simetría se programa mediante un cambio en el sentido del eje dentro del plano de trabajo seleccionado.

Ejemplo: Plano de trabajo G17 X/Y

La simetría del eje Y requiere un cambio en el sentido del eje X y, por consiguiente, se programa como `MIRROR X0`. El contorno se mecaniza en simetría especular en el lado opuesto del eje de simetría Y.

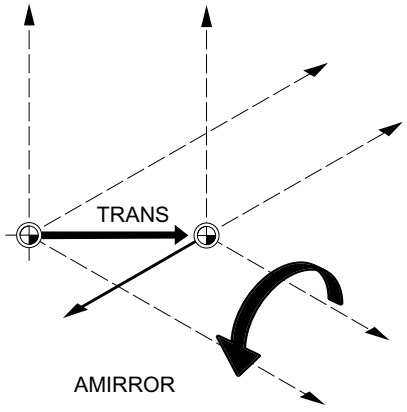


La simetría se refiere al sistema de coordenadas actualmente válido ajustado con G54 ... G57, G505 ... G599.

ATENCIÓN
Sin frame de origen
El comando `MIRROR` resetea todos los componentes del frame anteriormente programado.

AMIRROR X... Y... Z...

Se puede añadir una simetría a una transformación ya existente con la programación de `AMIRROR`. Se toma como referencia el sistema de coordenadas actualmente ajustado o el último programado.



Desactivar simetría

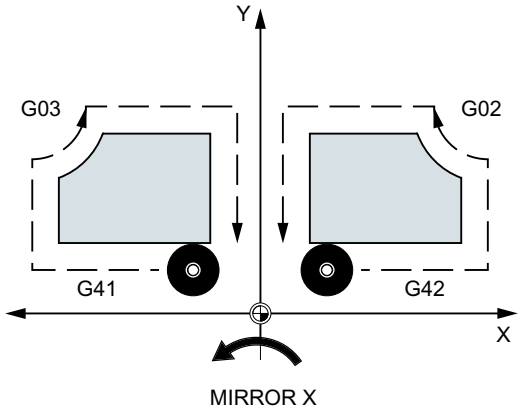
Para todos los ejes: `MIRROR` (sin indicación de eje)

Se resetean todos los componentes del frame anteriormente programado.

Corrección de radio de herramienta

Nota

El comando de simetría hace que el control conmute automáticamente los comandos (`G41`/`G42` o `G42`/`G41`) para compensar el radio de la herramienta de acuerdo con el nuevo sentido de mecanizado.



El mismo criterio se aplica para cambiar el sentido de mecanizado de los arcos de circunferencia (G2/G3 ó G3/G2).

Nota

Si se programa una rotación aditiva con el comando `AROT` después de `MIRROR`, se deberá invertir el sentido de rotación (positivo/negativo o negativo/positivo) según sea necesario. Las simetrías en los ejes geométricos son convertidos automáticamente por el control numérico en rotaciones y, en su caso, en simetrías en el eje de simetría especificado en datos de máquina. Esto también se aplica para decalajes de origen ajustables.

Eje de simetría

Mediante un dato de máquina se puede definir respecto de qué eje se desea realizar la simetría:

DM10610 $MN_MIRROR_REF_AX = <Valor>

Valor	Descripción
0	La simetría se realiza con respecto al eje programado (negación de los valores).
1	El eje X es el eje de referencia.
2	El eje Y es el eje de referencia.
3	El eje Z es el eje de referencia.

Interpretación de los valores programados

Mediante un dato de máquina se puede definir cómo deben interpretarse los valores programados:

DM10612 $MN_MIRROR_TOGGLE = <Valor>

Valor	Descripción
0	Los valores de eje programados no se evalúan.
1	Los valores de eje programados se evalúan: • Con valores de eje programados ≠ 0 se ejecuta la simetría del eje si ésta no se ha realizado todavía. • Con un valor de eje programado = 0 se desactiva la simetría.

2.12.9 Creación de frame por orientación de herramienta (TOFRAME, TOROT, PAROT)

TOFRAME genera un sistema de coordenadas rectangular cuyo eje Z coincide con la orientación de la herramienta actual. De este modo, el usuario tiene la posibilidad de retirar la herramienta en la dirección Z sin colisiones (p. ej. cuando se produce una rotura de la herramienta en un programa de 5 ejes).

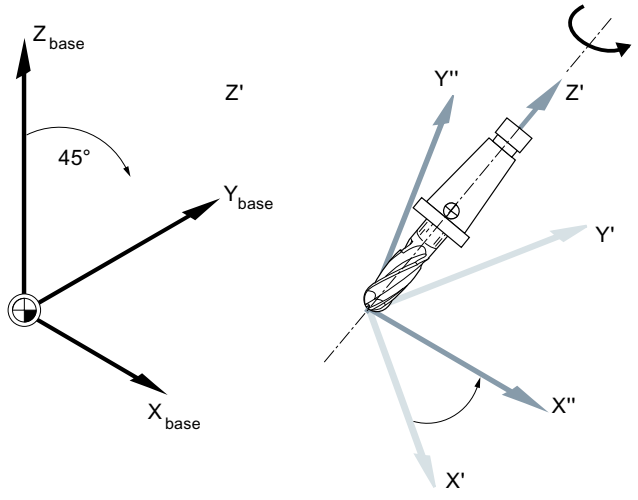
La posición de los ejes X e Y depende del ajuste en el dato de máquina DM21110 $MC_X_AXES_IN_OLD_X_Z_PLANE (sistema de coordenadas en la definición automática del frame). El nuevo sistema de coordenadas se deja, bien como se obtiene de la cinemática de la máquina, o bien se gira alrededor del nuevo eje Z hasta que el nuevo eje X se sitúe en el antiguo plano Z-X (ver las indicaciones del fabricante de la máquina).

El frame resultante, que describe la orientación, se encuentra en las variables del sistema para frames programables ($P_PFRAME).

Con TOROT sólo se sobrescribe la parte de rotación en el frame programado. Todos los demás componentes permanecen invariables.

TOFRAME y TOROT están adaptados a fresados en los que G17 suele estar activo (plano de trabajo X/Y). En cambio, en torneados o siempre que G18 o G19 estén activos, se necesitan frames en los que el eje X o Y coincida con la orientación de la herramienta. Estos frames se programan con los comandos TOFRAMEX/TOROTX o TOFRAMEY/TOROTY.

Con PAROT, el sistema de coordenadas de pieza (WKS) se orienta en la pieza.



Sintaxis

```
TOFRAME/TOFRAMEZ/TOFRAMEY/TOFRAMEX
...
TOROTOF

TOROT/TOROTZ/TOROTY/TOROTX
...
TOROTOF

PAROT
...
PAROTOF
```

Descripción

TOFRAME:	Orientar el eje Z del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame
TOFRAMEZ:	Como TOFRAME
TOFRAMEY:	Orientar el eje Y del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame
TOFRAMEX:	Orientar el eje X del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame
TOROT:	Orientar el eje Z del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame La rotación definida por TOROT es la misma que con TOFRAME.
TOROTZ:	Como TOROT
TOROTY:	Orientar el eje Y del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame
TOROTX:	Orientar el eje X del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame
TOROTOF:	Desactivar la orientación paralela a la orientación de la herramienta
PAROT:	Orientar el sistema de coordenadas de pieza a la pieza mediante rotación del frame Se conservan las traslaciones, las escalas y las simetrías en el frame activo.
PAROTOF:	La rotación del frame relativa a la pieza activada con PAROT se desactiva con PAROTOF.

Nota

Con el comando TOROT se consigue una programación consistente con portaherramientas orientables de forma activa para cada tipo de cinemática.

Por analogía a la situación con un portaherramientas orientable, se puede activar con PAROT un giro de la mesa de herramienta. De este modo, se define un frame que modifica la posición del sistema de coordenadas de pieza de tal manera que no se produce ningún movimiento compensatorio de la máquina. La instrucción de programación PAROT no se rechaza si no está activo ningún portaherramientas orientable.

Ejemplo

Código de programa	Comentarios
N100 G0 G53 X100 Z100 D0	
N120 TOFRAME	
N140 G91 Z20	; TOFRAME se incluye en el cálculo, y todos los desplazamientos programados para ejes geométricos toman como referencia el nuevo sistema de coordenadas.
N160 X50	
...	

Información adicional

Asignación dirección de eje

Si en lugar de TOFRAME/TOFRAMEZ o TOROT/TOROTZ se programa uno de los comandos TOFRAMEX, TOFRAMEY, TOROTX o TOROTY, se aplican las asignaciones de las direcciones de eje de acuerdo con esta tabla:

Comando	Dirección de la herramienta (aplicada)	Eje secundario (abscisas)	Eje secundario (ordenadas)
TOFRAME/TOFRAMEZ/ TOROT/TOROTZ	Z	X	Y
TOFRAMEY / TOROTY	Y	Z	X
TOFRAMEX / TOROTX	X	Y	Z

Frame de sistema propio para TOFRAME o TOROT

Los frames producidos por TOFRAME o TOROT se pueden escribir en un frame de sistema propio $P_TOOLFRAME. Para ello debe ajustarse el bit 3 en el dato de máquina MD28082 $MC_MM_SYSTEM_FRAME_MASK. En este proceso, el frame programable no varía. Las diferencias se producen si el frame programable se sigue editando.

2.12.10 Cancelar frame (G53, G153, SUPA, G500)

Al ejecutar determinados procesos, p. ej., desplazamiento al punto de cambio de herramienta, se tienen que definir distintos componentes de frame y suprimirlos de forma definida en el tiempo.

Los frames ajustables se pueden desactivar de forma modal o suprimir por secuencias.

Los frames programables pueden suprimirse por secuencias o borrarse.

Sintaxis

```
G53
G153
SUPA
G500
TRANS
ROT
SCALE
MIRROR
```

Descripción

`G53`:	Supresión activa por secuencias de todos los frames programables y ajustables
`G153`:	`G153` actúa como `G53` y suprime además el frame básico total ($P_ACTBFRAME)
`SUPA`:	`SUPA` actúa como `G153` y suprime además: • Decalajes con volante (DRF) • Movimientos superpuestos • Decalaje de origen externo • Decalaje de PRESET
`G500`:	Desactivación modalmente activa de todos los frames ajustables (G54 ... G57, G505 ... G599) cuando no hay ningún valor en G500.
`TRANS ROT SCALE MIRROR`:	sin indicación de eje, provoca la eliminación de los frames programables.

2.12.11 Programación: Cancelar superposiciones específicas de eje (CORROF)

Con el procedimiento CORROF se borran las siguientes superposiciones específicas de eje:

- Decalajes de origen aditivos ajustados mediante desplazamiento con volante (decalajes DRF)
- Offsets de posición programados con la variable de sistema $AA_OFF

Al borrar de un valor de superposición, se dispara una parada de decodificación previa y se aplica la componente de posición del movimiento superpuesto cancelado en la posición en el sistema de coordenadas básico. No se desplaza ningún eje.

El valor de posición en el sistema de coordenadas de **máquina** legible con la variable de sistema $AA_IM (consigna actual del **MKS** del eje) **no se modifica**.

El valor de posición en el sistema de coordenadas de **pieza** legible con la variable de sistema $AA_IW (consigna actual del **MKS** del eje) **se modifica**, ya que ahora contiene la componente cancelada del movimiento superpuesto.

Nota

Es posible programar `CORROF` en un **programa CN**.

No es posible programar `CORROF` en una **acción síncrona**.

Sintaxis

```
CORROF(<Axis>,"<String>"[,<Axis>,"<String>"])
```

Descripción

`CORROF`:	Procedimiento para cancelar los siguientes decalajes o superposiciones de un eje: • Decalaje DRF • Offset de posición ($AA_OFF)	
	Efecto:	modal
`<Axis>`:	Identificador de eje (identificador de eje de canal, geométrico o de máquina)	
	Tipo de dato:	AXIS
`<Cadena_de_caracteres>`:	Cadena de caracteres para definir el tipo de superposición	
	Tipo de dato:	BOOL
	Valor	**Descripción**
	`DRF`	Decalaje DRF
	`AA_OFF`	Offset de posición ($AA_OFF)

Ejemplos

Ejemplo 1: Cancelación específica de eje de un decalaje DRF (1)

A través del desplazamiento con volante DRF se produce un decalaje DRF en el eje X. Para el resto de ejes del canal no están activos decalajes DRF.

Código de programa	**Comentarios**
`N10 CORROF(X,"DRF")`	`; CORROF actúa aquí como DRFOF.`
`...`	

Ejemplo 2: Cancelación específica de eje de un decalaje DRF (2)

A través del desplazamiento con volante DRF se produce un decalaje DRF en el eje X y en el eje Y. Para el resto de ejes del canal no están activos decalajes DRF.

```
Código de programa        Comentarios
; Solo se cancela el decalaje DRF del eje X; el decalaje DRF del eje Y se conserva.
; Con DRFOF se habrían cancelado ambos decalajes.
N10 CORROF(X,"DRF")
...
```

Ejemplo 3: Cancelación específica de eje de un offset de posición $AA_OFF

```
Código de programa                                    Comentarios
; Para el eje X se efectúa una interpolación de un offset de posición == 10.
N10 WHEN TRUE DO $AA_OFF[X]=10 G4 F5


...
; El offset de posición del eje X se cancela: $AA_OFF[X]=0
; El eje X no se desplaza.
; El offset de posición se suma a la posición actual del eje X.
N80 CORROF(X,"AA_OFF")
...
```

Ejemplo 4: Cancelación específica de eje de un decalaje DRF y de un offset de posición $AA_OFF (1)

A través del desplazamiento con volante DRF se produce un decalaje DRF en el eje X. Para el resto de ejes del canal no están activos decalajes DRF.

```
Código de programa                                    Comentarios
; Para el eje X se efectúa una interpolación de un offset de posición 10.
N10 WHEN TRUE DO $AA_OFF[X]=10 G4 F5
...
; Solo se cancelan el decalaje DRF y el offset de posición del eje X.
; El decalaje DRF del eje Y se conserva.
N70 CORROF(X,"DRF",X,"AA_OFF")
...
```

Ejemplo 5: Cancelación específica de eje de un decalaje DRF y de un offset de posición $AA_OFF (2)

A través del desplazamiento con volante DRF se produce un decalaje DRF en el eje X y en el eje Y. Para el resto de ejes del canal no están activos decalajes DRF.

```
Código de programa                                    Comentarios
; Para el eje X se efectúa una interpolación de un offset de posición == 10.
N10 WHEN TRUE DO $AA_OFF[X]=10 G4 F5
...
```

Código de programa	Comentarios
; El decalaje DRF del eje Y y el offset de posición del eje X se cancelan.	
; El decalaje DRF del eje X se conserva.	
N70 CORROF(Y,"DRF",X,"AA_OFF")	
...	

Información adicional

$AA_OFF_VAL

Después de cancelar el offset de posición debido a $AA_OFF, la variable de sistema $AA_OFF_VAL (recorrido integrado de la superposición de ejes) del eje correspondiente es igual a cero.

$AA_OFF en el modo de operación JOG

También en el modo de operación JOG, con una variación de $AA_OFF se produce una interpolación del offset de posición como movimiento superpuesto si esta función se ha habilitado mediante el dato de máquina DM36750 $MA_AA_OFF_MODE.

$AA_OFF en una acción síncrona

Si al cancelar el offset de posición a través de `CORROF(<Eje>,"AA_OFF")` está activa una acción síncrona que inmediatamente vuelve a activar $AA_OFF (`DO $AA_OFF[<Eje>]=<Valor>`), entonces $AA_OFF se cancela y no se vuelve a activar, y se muestra la alarma 21660. Sin embargo, si la acción síncrona pasa a estar activa más tarde, p. ej., en la secuencia después de `CORROF`, se ajusta $AA_OFF y se produce la interpolación de un offset de posición.

Cambio de eje de canal automático

En caso de que un eje para el que se ha programado un `CORROF` esté activo en otro canal, éste se tomará entonces en el canal con el cambio de eje (requisito: DM30552 $MA_AUTO_GET_TYPE > 0) y se cancelará el offset de posición y/o el decalaje DRF.

2.12.12 Cancelación de decalajes de origen aditivos (DRFOF)

Los decalajes de origen aditivos ajustados mediante desplazamiento con volante (decalajes DRF) se cancelan a través del procedimiento `DRFOF`.

La cancelación dispara una parada de decodificación previa y hace que se aplique la componente de posición del decalaje DRF cancelado en la posición en el sistema de coordenadas básico, sin que se desplace ningún eje. El valor de las variables del sistema $AA_IM (consigna actual del MKS de un eje) no se modifica; el valor de las variables del sistema $AA_IW (consigna actual del WKS de un eje) se modifica, ya que ahora contiene la componente cancelada del movimiento superpuesto.

Sintaxis

```
DRFOF
```

Descripción

DRFOF:	Procedimiento para cancelar los decalajes DRF para todos los ejes activos del canal	
	Eficacia:	modal

Ejemplos

Ejemplo 1: Cancelación específica de eje de un decalaje DRF (1)

A través del desplazamiento con volante DRF se produce un decalaje DRF en el eje X. Para el resto de ejes del canal no están activos decalajes DRF.

Código de programa	Comentarios
N10 CORROF(X,"DRF")	; CORROF actúa aquí como DRFOF.
...	

Ejemplo 2: Cancelación específica de eje de un decalaje DRF (2)

A través del desplazamiento con volante DRF se produce un decalaje DRF en el eje X y en el eje Y. Para el resto de ejes del canal no están activos decalajes DRF.

Código de programa	Comentarios
N10 CORROF(X,"DRF")	; Solo se cancela el decalaje DRF del eje X; el decalaje DRF del eje Y se conserva (con DRFOF se habrían cancelado los dos decalajes).
...	

Ejemplo 3: Cancelación específica de eje de un decalaje DRF y de un offset de posición $AA_OFF (1)

A través del desplazamiento con volante DRF se produce un decalaje DRF en el eje X. Para el resto de ejes del canal no están activos decalajes DRF.

Código de programa	Comentarios
N10 WHEN TRUE DO $AA_OFF[X]=10 G4 F5	; Para el eje X se efectúa una interpolación de un offset de posición == 10.
...	
N70 CORROF(X,"DRF",X,"AA_OFF")	; Solo se cancelan el decalaje DRF y el offset de posición del eje X; el decalaje DRF del eje Y se conserva.
...	

Ejemplo 4: Cancelación específica de eje de un decalaje DRF y de un offset de posición $AA_OFF (2)

A través del desplazamiento con volante DRF se produce un decalaje DRF en el eje X y en el eje Y. Para el resto de ejes del canal no están activos decalajes DRF.

Código de programa	Comentarios
N10 WHEN TRUE DO $AA_OFF[X]=10 G4 F5	; Para el eje X se efectúa una interpolación de un offset de posición == 10.

Código de programa	Comentarios
...	
N70 CORROF(Y,"DRF",X,"AA_OFF")	; El decalaje DRF del eje Y y el offset de posición del eje X se cancelan; el decalaje DRF del eje X se conserva.
...	

2.12.13 Decalajes de origen específicos de rectificado (GFRAME0, GFRAME1 ... GFRAME100)

Comando para activar un frame de rectificado en el canal

Al programar el comando GFRAME<n> se activa el frame de rectificado correspondiente de la gestión de datos $P_GFR[<n>] en el canal. Para ello, el frame de rectificado activo $P_GFRAME se iguala al frame de rectificado de la gestión de datos $P_GFR[<n>]:

GFRAME<n> ⇒ $P_GFRAME = $P_GFR[<n>]

Comando	Frame de rectificado activado en el canal
GFRAME0	$P_GFR[0] (**frame cero**)
GFRAME1	$P_GFR[1]
...	...
GFRAME100	$P_GFR[100]

Sintaxis

GFRAME<n>

Descripción

GFRAME<n>:	Activación del frame de rectificado <n> de la gestión de datos	
	Grupo G:	64
	Posición básica:	MD20150 $MC_GCODE_RESET_VALUES[**63**]
	Efecto:	Modal
<n>:	Número del frame de rectificado	
	Rango de valores:	0, 1, 2 ... 100

2.13 Emisión de funciones auxiliares

Función

Las funciones auxiliares se emiten para informar oportunamente al PLC acerca del instante en el que el programa de pieza desea que él realice determinadas maniobras en la máquina herramienta. Esto ocurre transfiriendo las correspondientes funciones auxiliares con sus parámetros a la interfaz del PLC. El procesamiento de los valores y señales transmitidos se realiza en el programa de usuario del PLC.

funciones auxiliares

Las siguientes funciones auxiliares pueden transmitirse al PLC:

Función auxiliar	Dirección
Selección de herramienta	T
Corrección de herramienta	D, DL
Avance	F/FA
velocidad de giro del cabezal	S
Funciones M	M
Funciones H	H

Para cada grupo de funciones o función individual se define con datos de máquina si la emisión debe activarse **antes**, **durante** o **después** del movimiento desplazamiento.

Se puede lograr que el PLC se comporte con diferentes modos de acuse para la emisión de funciones auxiliares.

Propiedades

La siguiente tabla general resume propiedades importantes de las funciones auxiliares:

Función	Direcciones extendidas		Valor			Explicación	Cantidad máxima por secuencia
	Descripción	Rango	Rango	Tipo	Descripción		
M	-	0 (implícito)	0 ... 99	INT	Función	Para el rango de valores de 0 a 99, la dirección extendida es 0. Obligatoriamente sin dirección extendida: M0, M1, M2, M17, M30	5
	N.º de cabezal	1 - 12	1 ... 99	INT	Función	M3, M4, M5, M19, M70 con dirección extendida. N.º de cabezal (p. ej. `M2=5` ; parada del cabezal para cabezal 2). Si no hay n.º de cabezal, la función es válida para el cabezal maestro.	
	Cualquiera	0 - 99	100 ... 2147483647	INT	Función	Función M de usuario*	
S	N.º de cabezal	1 - 12	0 ... ± 1,8*10^{308}	REAL	Velocidad de giro	Si no hay n.º de cabezal, la función es válida para el cabezal maestro.	3
H	Cualquiera	0 - 99	0 ... ± 2147483647 ± 1,8*10^{308}	INT REAL	Cualquiera	Funciones que carecen de efecto en el CN, y deben ser implementadas exclusivamente mediante el PLC.*	3
T	Nº de cabezal (con gestión de htas. activa)	1 - 12	0-32000 (también nombres de htas. con gestión de htas. activa)	INT	Selección de hta.	Los nombres de hta. no se transfieren a la interfaz del PLC.	1
D	-	-	0 - 12	INT	Selección de la corrección de hta.	D0: Cancelación Preajuste: D1	1
DL	Corrección dependiente de la posición	1 - 6	0 ... ± 1,8*10^{308}	REAL	Selección de la corrección fina de herramienta	Se refiere al número D seleccionado anteriormente.	1
F	-	-	0.001 - 999 999,999	REAL	Avance de contorneado		6
FA	Nº de eje	1 - 31	0.001 - 999 999,999	REAL	Avance del eje		
* El significado de las funciones es determinado por el fabricante de la máquina (**ver indicaciones del fabricante de la máquina**).							

Información adicional

Cantidad de funciones que se pueden emitir en una secuencia CN

En una secuencia CN se puede programar un máximo de 10 emisiones de funciones. También se pueden emitir funciones auxiliares desde la sección de **acciones síncronas**.

Agrupación

Las funciones auxiliares mencionadas se pueden agrupar. Para algunas funciones M existe ya una división predefinida de los grupos. Con la agrupación se puede definir el comportamiento para el acuse.

Salida de funciones rápida (QU)

Las funciones que no hayan sido definidas como salidas rápidas se pueden utilizar como tales mediante la palabra reservada `QU`. La ejecución del programa continúa sin esperar al acuse de la ejecución de dicha función. De este modo, se pueden evitar paradas innecesarias así como la interrupción de secuencias de desplazamiento.

Nota

Para la función "Salida de funciones rápida" deben estar activados los correspondientes datos de máquina (→ **fabricante de la máquina**).

Emisión de funciones en secuencias con desplazamiento

La transferencia de información, así como la espera de las correspondientes reacciones, requieren tiempo y, por ello, pueden afectar también a los desplazamientos.

Acuse rápido sin retardo de cambio de secuencia

Se puede influenciar mediante dato de máquina el comportamiento para el cambio de secuencia. Con el ajuste "sin retardo en el cambio de secuencia", se obtiene el siguiente comportamiento para funciones auxiliares rápidas:

Emisión de funciones auxiliares	Comportamiento
antes del desplazamiento	La transición de secuencias con emisión de funciones auxiliares rápidas se realiza **sin** interrupción y **sin** reducción de la velocidad. Las funciones auxiliares se emiten en el primer ciclo de interpolador de la secuencia. La siguiente secuencia se ejecuta sin retardo en el acuse.
Durante el desplazamiento	La transición de secuencias con emisión de funciones auxiliares rápidas se realiza **sin** interrupción y **sin** reducción de la velocidad. Las funciones auxiliares se emiten durante la ejecución de la secuencia. La siguiente secuencia se ejecuta sin retardo en el acuse.
Después del desplazamiento.	El movimiento se detiene al final de la secuencia. La emisión de las funciones auxiliares se realiza al final de la secuencia. La siguiente secuencia se ejecuta sin retardo en el acuse.

⚠ PRECAUCIÓN

Emisión de funciones en modo de contorneado

La emisión de funciones **antes** de los desplazamientos interrumpe el modo de contorneado (G64/G641) y provoca una parada precisa para la secuencia precedente.

La emisión de funciones **después** de los desplazamientos interrumpe el modo de contorneado (G64/G641) y provoca una parada precisa para la secuencia actual.

Importante: La espera de una señal de acuse del PLC también puede causar la interrupción del modo de contorneado, p. ej., en sucesiones de comandos M en secuencias con desplazamientos muy cortos.

2.13.1 Funciones M

Con las funciones M pueden activarse en la máquina operaciones de maniobra tales como "CON/DES refrigerante", así como otras funcionalidades.

Sintaxis

```
M<Valor>
M[<Dirección extendida>]=<Valor>
```

Descripción

`M`:	Dirección para la programación de las funciones M	
`<Dirección extendida>`:	En algunas funciones M pueden utilizarse direcciones extendidas (p. ej., indicación del número de cabezal en funciones de cabezal).	
`<Valor>`:	La adjudicación de valores (número de función M) permite la asignación a una determinada función de la máquina.	
	Tipo:	INT
	Rango de valores:	0 ... 2147483647 (máx. valor INT)

Funciones M predefinidas

Algunas funciones M que son importantes para el desarrollo de los programas están definidas de forma por defecto:

Función M	Descripción
M0*	Parada programada
M1*	Parada opcional
M2*	Fin del programa principal (como M30)
M3	Sentido de giro del cabezal a derechas
M4	Sentido de giro del cabezal a izquierdas
M5	Parada del cabezal

Función M	Descripción
M6	Cambio de herramienta (ajuste por defecto)
M17*	Fin del subprograma
M19	Posicionar cabezal
M30*	Fin del programa principal (como M2)
M40	Cambio automático de gama o escalón de reducción
M41	Escalón de reducción 1
M42	Escalón de reducción 2
M43	Escalón de reducción 3
M44	Escalón de reducción 4
M45	Escalón de reducción 5
M70	El cabezal se conmuta al modo Eje

Nota

Para las funciones M marcadas con * no se pueden utilizar direcciones extendidas.

Las funciones M0, M1, M2, M17 y M30 siempre se activan **tras finalizar** el desplazamiento de los ejes.

Funciones M definidas por el fabricante de la máquina

El fabricante de la máquina puede asignar todos los números de función M libres, p. ej. con funciones de conmutación para controlar dispositivos tensores o activar/desactivar otras funciones de la máquina.

Nota

Las funcionalidades asignadas a los números de función M libres son específicas de la máquina. Por ello, una determinada función M puede poseer funcionalidades diferentes en distintas máquinas.

Las funciones M disponibles en una máquina y sus funcionalidades figuran en las indicaciones del fabricante de la máquina.

Ejemplos

Ejemplo 1: número máximo de funciones M en la secuencia

```
Código de programa          Comentarios
N10 S...
N20 X... M3                 ; Función M en la secuencia con desplazamiento de eje,
                            ; El cabezal arranca antes del desplazamiento del eje X.
N180 M789 M1767 M100 M102   ; Máximo 5 funciones M en la secuencia.
M376
```

Ejemplo 2: función M como salida rápida

Código de programa	Comentarios
`N10 H=QU(735)`	`; Salida rápida para H735.`
`N10 G1 F300 X10 Y20 G64`	
`N20 X8 Y90 M=QU(7)`	`; Salida rápida para M7.`

Se ha programado la función M7 como salida rápida, de manera que no se interrumpa el modo de contorneado (G64).

Nota

Sólo se debe de emplear esta función en casos aislados, ya que puede alterar los tiempos para la emisión de otras funciones.

Más información sobre los comandos M predefinidos

Parada programada: M0

Se detiene el mecanizado en una secuencia CN con `M0`. A continuación se pueden realizar operaciones tales como, p. ej., evacuar viruta, medir, etc.

Parada programada 1, parada opcional: M1

M1 se puede ajustar con:

- Diálogo HMI "Influenciación del programa"
  o bien
- Interfaz CN/PLC

La ejecución de programas del CN se detiene en las secuencias programadas.

Parada programada 2 - una función auxiliar asociada con M1 con parada en la ejecución del programa

La parada programada Parada 2 se puede ajustar a través del diálogo HMI "Influenciación del programa" y permite en todo momento una interrupción de procesos tecnológicos al final de la pieza a mecanizar. De este modo, el operador puede intervenir en la producción, p. ej., para eliminar virutas.

Fin del programa: M2, M17, M30

Los programas se finalizan con `M2`, `M17` o `M30`. Si se llama al programa principal desde otro programa (o subprograma), `M2`/`M30` actúan como `M17` y viceversa, es decir, `M17` actúa en el programa principal como `M2`/`M30`.

Funciones de cabezal: M3, M4, M5, M19, M70

En todas las funciones de cabezal se pueden utilizar direcciones ampliadas con indicación del número del cabezal.

Ejemplo:

Código de programa	**Comentarios**
M2=3	; Giro a derechas del segundo cabezal

Si se omite la dirección ampliada, la función es válida para el cabezal maestro.

2.14 Comandos complementarios

2.14.1 Salida de avisos (MSG)

La instrucción `MSG()` permite emitir una cadena de caracteres cualquiera como aviso para el operador desde el programa de pieza.

Sintaxis

```
MSG("<Texto de aviso>"[,<Ejecución>])
...
MSG ()
```

Descripción

`MSG`:	Llamada predefinida a un subprograma para la emisión de un aviso		
`<Texto de aviso>`:	Cualquier cadena de caracteres para mostrar como aviso		
	Tipo:	STRING	
	Longitud máxima:	124 caracteres; se visualizan en dos líneas (2*62 caracteres)	
	En el texto del aviso también pueden mostrarse variables utilizando el operador de concatenación "<<".		
`<Ejecución>`:	Parámetro para definir el instante en que se ejecutará la escritura del aviso (opcional)		
	Tipo:	INT	
	Valor:	0 (ajuste básico)	Para la escritura del aviso no se genera una secuencia propia del proceso principal. Se realiza en la siguiente secuencia CN ejecutable. No se interrumpe un modo de contorneado activo.
		1	Para la escritura del aviso se genera una secuencia propia del proceso principal. Se interrumpe un modo de contorneado activo.
`MSG()`:	La programación de `MSG()` sin texto de aviso borra el aviso actual. Sin borrado, la indicación se mantiene hasta que aparezca el siguiente aviso.		

Nota

Si un aviso debe emitirse en el idioma activo en la interfaz de usuario, el usuario necesitará información sobre el idioma ajustado actualmente en el HMI. Esta información se puede consultar en el programa de pieza y en acciones síncronas, a través de la variable de sistema $AN_LANGUAGE_ON_HMI (Página 1355).

Ejemplos

Ejemplo 1: emitir/borrar aviso

Código de programa	Comentarios
N10 G91 G64 F100	; Modo de contorneado
N20 X1 Y1	
N... X... Y...	
N20 MSG ("Mecanizado 1.ª parte")	; El aviso se emite solo a partir de N30.
	; Se mantiene el modo de contorneado.
N30 X... Y...	
N... X... Y...	
N400 X1 Y1	
N410 MSG ("Mecanizado 2.ª parte",1)	; El aviso se emite con N410.
	; El modo de contorneado se interrumpe.
N420 X1 Y1	
N... X... Y...	
N900 MSG ()	; Borrar aviso.

Ejemplo 2: texto de aviso con variable

Código de programa	Comentarios
N10 R12=$AA_IW [X]	; Posición actual del eje X en R12.
N20 MSG("Posición del eje X"<<R12<<"comprobar")	; Emitir aviso con variable R12.
...	
N90 MSG ()	; Borrar el aviso de N20.

2.14.2 Escribir cadena de caracteres en variable BTSS (WRTPR)

La función `WRTPR()` permite escribir cualquier cadena de caracteres desde el programa de pieza en la variable progProtText de BTSS.

Sintaxis

```
WRTPR(<String>[,<ExecTime>])
```

Significado

WRTPR:	Llamada de función para emitir una cadena de caracteres.		
<Cadena_de_caracteres>:	Cadena de caracteres cualquiera que se escribe en la variable progProtText de BTSSC.		
	Tipo:	STRING	
	Longitud máxima:	128 caracteres	
<ExecTime>:	Parámetro opcional para definir el instante en que se ejecutará la escritura de la cadena de caracteres.		
	Tipo:	INT	
	Rango de valores:	0, 1	
		0 (por defecto)	Para la escritura de la cadena de caracteres no se genera una secuencia propia del proceso principal. Se realiza en la siguiente secuencia CN ejecutable. No se interrumpe un modo de contorneado activo.
		1	Para la escritura de la cadena de caracteres se genera una secuencia propia del proceso principal. Se interrumpe un modo de contorneado activo.

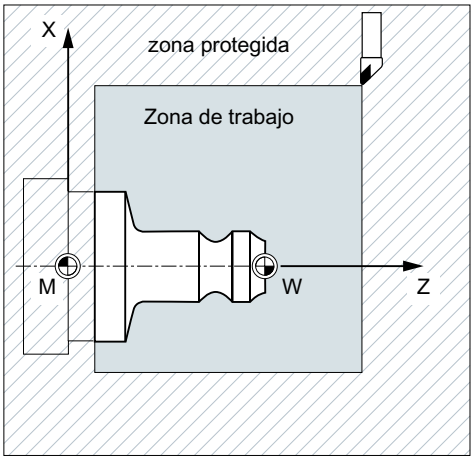
Ejemplos

```
Código de programa          Comentarios
N10 G91 G64 F100            ; Modo de contorneado
N20 X1 Y1
N30 WRTPR("N30")            ; La cadena de caracteres "N30" se escribe primero en N40.
                            ; Se mantiene el modo de contorneado.
N40 X1 Y1
N50 WRTPR("N50",1)          ; La cadena de caracteres "N50" se escribe primero en N50.
                            ; El modo de contorneado se interrumpe.
N60 X1 Y1
```

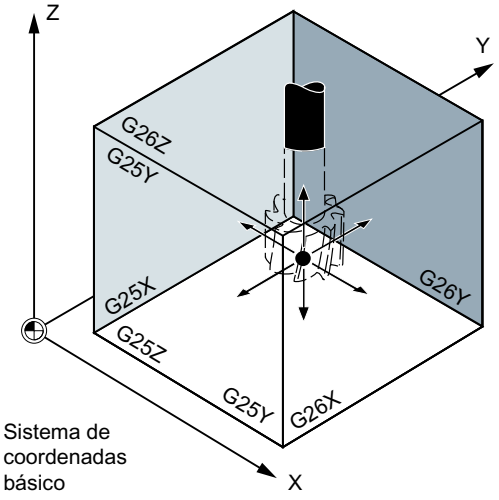
2.14.3 Limitación del campo de trabajo

2.14.3.1 Limitación del campo de trabajo en BKS (G25/G26, WALIMON, WALIMOF)

Con G25/G26 se limita la zona por donde se puede desplazar la herramienta (campo de trabajo, zona de trabajo). Esta limitación es válida para todos los ejes de canal. Las áreas fuera de las limitaciones de la zona de trabajo definidas con G25/G26 están bloqueadas para los movimientos de herramientas.



Las coordenadas introducidas para cada uno de los ejes se refieren al sistema de coordenadas básico:



La limitación del campo de trabajo para todos los ejes definidos como válidos tiene que estar programada con el comando WALIMON. WALIMOF desactiva el campo de trabajo. WALIMON es el ajuste estándar y solamente se debe de volver a programar en el caso de que se haya desactivado previamente la limitación del campo de trabajo.

Sintaxis

```
G25 X…Y…Z…
G26 X…Y…Z…
WALIMON
...
WALIMOF
```

Descripción

G25:	Limitación **inferior** del campo de trabajo Asignación de valores en ejes de canal en el sistema de coordenadas básico
G26:	Limitación **superior** del campo de trabajo Asignación de valores en ejes de canal en el sistema de coordenadas básico
X… Y… Z…:	Límites inferior o superior del campo de trabajo para cada uno de los ejes de canal Las indicaciones se refieren al sistema de coordenadas básico.
WALIMON:	**Act**ivar la limitación del campo de trabajo para todos los ejes
WALIMOF:	**Des**activar la limitación del campo de trabajo para todos los ejes

Además de la entrada programable de los valores a través de G25/G26 también es posible la entrada a través de datos de operador específicos del eje.

SD43420 $SA_WORKAREA_LIMIT_PLUS (limitación del campo de trabajo positiva)

SD43430 $SA_WORKAREA_LIMIT_MINUS (limitación del campo de trabajo negativa)

La activación y la desactivación de la limitación del campo de trabajo parametrizada con SD43420 y SD43430 se realizan específicamente para la dirección a través de los datos de operador específicos del eje y con efecto inmediato:

SD43400 $SA_WORKAREA_PLUS_ENABLE (limitación del campo de trabajo en dirección positiva activa)

SD43410 $SA_WORKAREA_MINUS_ENABLE (limitación del campo de trabajo en dirección negativa activa)

Con la activación/desactivación específica de la dirección es posible limitar la zona de trabajo para un eje sólo en una dirección.

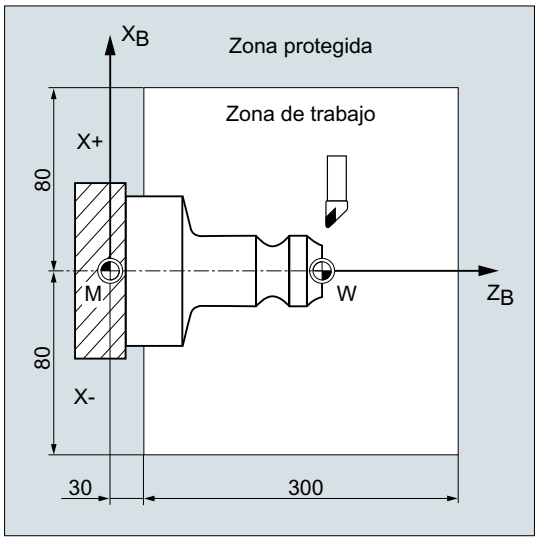
Nota

La limitación del campo de trabajo programada con G25/G26 tiene prioridad y sobrescribe los valores introducidos en SD43420 y SD43430.

Nota

Con G25/G26 también se puede limitar la velocidad del cabezal mediante la dirección S. Para más información al respecto, ver "Límite programable de la velocidad del cabezal (G25, G26) (Página 107)".

Ejemplo



La limitación del campo de trabajo con G25/26 permite delimitar la zona de trabajo de un torno de modo que los equipos periféricos, tales como torretas, palpadores de medida, etc., estén protegidos contra daños.

Ajuste por defecto: WALIMON

Código de programa	Comentarios
N10 G0 G90 F0.5 T1	
N20 G25 X-80 Z30	; Definición de la limitación inferior para los diferentes ejes de coordenadas
N30 G26 X80 Z330	; Definición de la limitación superior
N40 L22	; Programa de desbaste
N50 G0 G90 Z102 T2	; Al punto de cambio de herramienta
N60 X0	
N70 WALIMOF	; Desactivar limitación del campo de trabajo
N80 G1 Z-2 F0.5	; Taladrado
N90 G0 Z200	; Retorno
N100 WALIMON	; Activar limitación del campo de trabajo
N110 X70 M30	; Fin del programa

Información adicional

Punto de referencia en la herramienta

Cuando está activa la corrección longitudinal de herramienta, se vigila la punta de la herramienta como punto de referencia; en caso contrario, el punto de referencia se encuentra situado en el portaherramientas.

La consideración del radio de herramienta se tiene que activar por separado. Esto se realiza a través del dato de máquina específico del canal:

MD21020 $MC_WORKAREA_WITH_TOOL_RADIUS

Si el punto de referencia de la herramienta se encuentra fuera del área definida por la limitación del campo de trabajo o abandona dicha zona, la ejecución del programa se detiene.

Nota

Si hay transformadas activas, la consideración de los datos de herramienta (longitud de herramienta y radio de herramienta) puede diferir del comportamiento descrito.

Limitación programable del campo de trabajo, G25/G26

Para cada eje se pueden definir una limitación superior (G26) y otra inferior (G25) para el campo de trabajo. Estos valores se aplican con efecto inmediato y se conservan con el ajuste de DM correspondiente (→ MD10710 $MN_PROG_SD_RESET_SAVE_TAB) después del RESET y de la reconexión.

Nota

La función predefinida CALCPOSI (Página 599) permite comprobar, antes de efectuar desplazamientos, si se ejecuta el recorrido previsto, teniendo en cuenta las limitaciones del campo de trabajo y/o las zonas protegidas.

2.14.3.2 Limitación del campo de trabajo en WKS/ENS (WALCS0 ... WALCS10)

La "limitación del campo de trabajo en WKS/ENS" permite limitar de manera flexible para cada pieza la zona de desplazamiento de los ejes de canal en el sistema de coordenadas de pieza (WKS) o en el sistema de origen ajustable (ENS). Está pensada fundamentalmente para su uso en el ámbito de los tornos convencionales.

Requisitos

Los ejes de canal deben estar referenciados.

Grupo de limitación del campo de trabajo

Pueden utilizarse grupos de limitación del campo de trabajo para que, al cambiar asignaciones de ejes, p. ej., al activar/desactivar transformadas o el frame activo, no sea necesario reescribir las limitaciones del campo de trabajo específicas de eje para todos los ejes de canal.

Cada grupo de limitación del campo de trabajo incluye los siguientes datos:

- Limitaciones del campo de trabajo para todos los ejes de canal
- Sistema de referencia de la limitación del campo de trabajo

Sintaxis

```
...
$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
```

Descripción

$P_WORKAREA_CS_COORD_SYSTEM[<WALimNo>]=<Value>		
Sistema de coordenadas al cual se refiere la limitación del campo de trabajo de un grupo:		
<WALimNo>:	Grupo de limitación del campo de trabajo	
	Tipo:	INT
	Rango de valores:	0 (grupo 1) ... 9 (grupo 10)
<Value>:	Valor del tipo INT	
	1	Sistema de coordenadas de pieza (WKS)
	3	Sistema de origen ajustable (ENS)

$$P_WORKAREA_CS_PLUS_ENABLE[<WALimNo>,<Ax>]=<Value>		
Habilitar la limitación del campo de trabajo en dirección de eje **positiva** para el eje de canal especificado		
<WALimNo>:	Grupo de limitación del campo de trabajo	
	Tipo:	INT
	Rango de valores:	0 (grupo 1) ... 9 (grupo 10)
<Ax>:	Nombre de eje de canal	
<Value>:	Valor del tipo BOOL	
	0 (FALSE)	Sin habilitación
	1 (TRUE)	Habilitación

$P_WORKAREA_CS_MINUS_ENABLE[<WALimNo>,<Ax>]=<Value>		
Habilitar la limitación del campo de trabajo en dirección de eje **negativa** para el eje de canal especificado		
<WALimNo>:	Grupo de limitación del campo de trabajo	
	Tipo:	INT
	Rango de valores:	0 (grupo 1) ... 9 (grupo 10)
<Ax>:	Nombre de eje de canal	
<Value>:	Valor del tipo BOOL	
	0 (FALSE)	Sin habilitación
	1 (TRUE)	Habilitación

$P_WORKAREA_CS_LIMIT_PLUS[<WALimNo>,<Ax>]=<Value>		
Limitación del campo de trabajo en dirección **positiva** del eje de canal especificado		
<WALimNo>:	Grupo de limitación del campo de trabajo	
	Tipo:	INT
	Rango de valores:	0 (grupo 1) ... 9 (grupo 10)
<Ax>:	Nombre de eje de canal	
<Value>:	Valor del tipo REAL	

$P_WORKAREA_CS_LIMIT_MINUS[<WALimNo>,<Ax>]=<Value>		
Limitación del campo de trabajo en dirección **negativa** del eje de canal especificado		
<WALimNo>:	Grupo de limitación del campo de trabajo	
	Tipo:	INT
	Rango de valores:	0 (grupo 1) ... 9 (grupo 10)
<Ax>:	Nombre de eje de canal	
<Value>:	Valor del tipo REAL	

WALCS<n>:	Activación de las limitaciones del campo de trabajo de un grupo de limitaciones de campo de trabajo	
<n>:	Número del grupo de limitación del campo de trabajo	
	Rango de valores:	1 ... 10

WALCS0:	Desactivación de las limitaciones del campo de trabajo activas en el canal

Nota

El número realmente disponible de grupos de limitación del campo de trabajo varía en función de la configuración (→ ver las indicaciones del fabricante de la máquina).

Ejemplo

En el canal se encuentran definidos 3 ejes: X, Y y Z

Se quiere definir y, a continuación, activar un grupo de limitación del campo de trabajo nº 2 en el cual los ejes en el WKS son limitados según las siguientes especificaciones:

- Eje X en dirección positiva: 10 mm
- Eje X en dirección negativa: Sin limitaciones
- Eje Y en dirección positiva: 34 mm
- Eje Y en dirección negativa: -25 mm

- Eje Z en dirección positiva: Sin limitaciones
- Eje Z en dirección negativa: -600 mm

Código de programa	Comentarios
...	
N51 $P_WORKAREA_CS_COORD_SYSTEM[1]=1	; La limitación del campo de trabajo del grupo de limitación del campo de trabajo 2 es válida en el WKS.
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	; Activar el grupo de limitación del campo de trabajo 2.
...	

Información adicional

Efecto

La limitación del campo de trabajo con WALCS1 - WALCS10 actúa independientemente de la limitación del campo de trabajo con WALIMON. Si ambas funciones están activas actúa la primera limitación con la cual se encuentra el desplazamiento del eje.

Punto de referencia en la herramienta

La consideración de los datos de herramienta (longitud y radio de la herramienta) y, en consecuencia, el punto de referencia en la herramienta en la vigilancia de la limitación del campo de trabajo corresponde al comportamiento en la limitación del campo de trabajo con WALIMON.

2.14.4 Búsqueda del punto de referencia (G74)

Tras la conexión de la máquina, todos los ejes equipados con sistemas de medida del tipo incremental deben realizar un desplazamiento para buscar el punto de referencia. Una vez realizada la operación de búsqueda de referencia se pueden desplazar los ejes mediante secuencias de programa de pieza.

La función G74 permite buscar el punto de referencia dentro del programa de CN.

Sintaxis

`G74 X1=0 Y1=0 Z1=0 A1=0 …` ; Programación en una secuencia CN propia

Descripción

`G74`:	Llamada a comando G aproximación al punto de referencia
`X1=0 Y1=0 Z1=0 …`:	La dirección de ejes de máquina indicada `X1`, `Y1`, `Z1`… para **ejes lineales** lleva al punto de referencia
`A1=0 B1=0 C1=0 …`:	La dirección de ejes de máquina indicada `A1`, `B1`, `C1`… para **ejes giratorios** lleva al punto de referencia

Nota

Antes de realizar el desplazamiento para buscar el punto de referencia no se puede activar ninguna transformada en la que se haya involucrado alguno de los ejes que debe realizar dicho desplazamiento mediante la instrucción `G74`.

La transformada se desactiva con el comando `TRAFOOF`.

Ejemplo

Cuando se cambia el sistema de medida se realiza un desplazamiento al punto de referencia y se define el origen de la pieza.

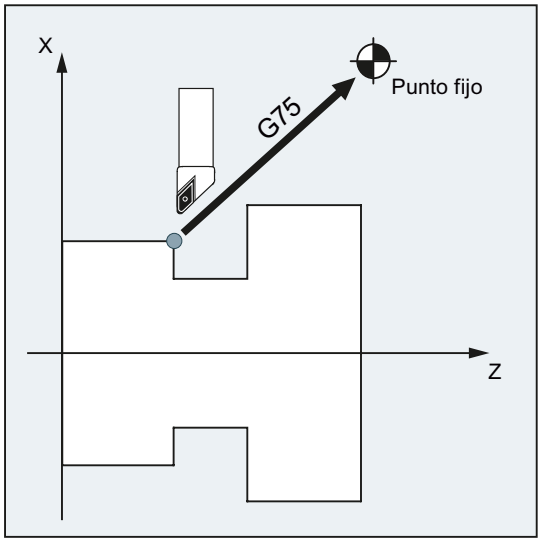
Código de programa	**Comentarios**
`N10 SPOS=0`	`; Cabezal en regulación de posición`
`N20 G74 X1=0 Y1=0 Z1=0 C1=0`	`; Búsqueda del punto de referencia para ejes lineales y giratorios`
`N30 G54`	`; Decalaje de origen`
`N40 L47`	`; Programa de desbaste`
`N50 M30`	`; Fin del programa`

2.14.5 Desplazamiento a punto fijo (G75).

Con el comando `G75` activo por secuencias, pueden desplazarse ejes individual e independientemente entre sí a puntos fijos del espacio de la máquina, p. ej., a puntos de cambio de herramienta, puntos de carga, puntos de cambio de paleta, etc.

Los puntos fijos son posiciones en el sistema de coordenadas de máquina guardadas en datos de máquina (DM30600 $MA_FIX_POINT_POS[n]). Por cada eje pueden definirse 4 puntos fijos como máximo.

Es posible realizar aproximaciones a los puntos fijos desde cualquier programa CN con independencia de las posiciones de herramienta o de pieza actuales. Antes del movimiento de los ejes, se realiza una parada de decodificación previa interna.



Requisitos

Para el desplazamiento a puntos fijos con `G75` deben cumplirse los siguientes requisitos:

- Las coordenadas de punto fijo deben determinarse exactamente y estar guardadas en datos de máquina.
- Los puntos fijos deben estar situados dentro de la zona de desplazamiento válida (→ prestar atención a los límites de fin de carrera de software)
- Los ejes que deben desplazarse deben estar referenciados.
- No debe estar activa ninguna corrección de radio de herramienta.
- No puede estar activa ninguna transformación cinemática.
- Los ejes que deben desplazarse no deben participar en ninguna transformada activa.
- Ninguno de los ejes que deben desplazarse puede ser el eje esclavo de un acoplamiento activo.
- Ninguno de los ejes que deben desplazarse puede ser eje de un conjunto Gantry.
- Los ciclos de compilación no deben aplicar aditivamente partes de movimiento.

Sintaxis

```
G75 <Nombre eje><Posición eje> ... FP=<n>
```

Descripción

`G75`:	Desplazamiento a punto fijo
`<Nombre eje>`:	Nombre del eje de máquina que debe desplazarse al punto fijo. Están permitidos todos los identificadores de eje.

<Posición eje>:	El valor de posición carece de significado. Por eso, en general se indica el valor "0".		
FP=:	Punto fijo al que debe realizarse el desplazamiento.		
	<n>:	Número de punto fijo	
		Rango de valores:	1, 2, 3, 4
	Nota: si no está programado ningún FP=<n> o ningún número de punto fijo o si está programado FP=0, esto se interpreta como FP=1 y se realiza el desplazamiento al punto fijo 1.		

Nota

En una secuencia G75 pueden programarse también varios ejes. Entonces, los ejes se desplazan simultáneamente al punto fijo indicado.

Nota

El valor de la dirección FP no debe ser mayor que la cantidad de puntos fijos definidos para cada eje programado (DM30610 $MA_NUM_FIX_POINT_POS).

Ejemplo

Para un cambio de herramienta, los ejes X (= AX1) y Z (= AX3) deben desplazarse a la posición fija del eje de máquina 1 con X = 151,6 y Z = -17,3.

Datos de máquina:

- DM30600 $MA_FIX_POINT_POS[AX1,0] = 151.6
- DM30600 $MA_FIX_POINT[AX3,0] = 17.3

Programa CN:

```
Código de programa          Comentarios
…
N100 G55                    ; Activar decalaje de origen ajustable.
N110 X10 Y30 Z40            ; Desplazarse a posiciones en el WKS.
N120 G75 X0 Z0 FP=1 M0      ; El eje X se desplaza a 151,6
                            ; y el eje Z a 17,3 (en el MKS).
                            ; Cada eje se desplaza por sí mismo con su máxima velo-
                            cidad.
                            ; En esta secuencia no deben estar activos movimientos
                            adicionales.
                            ; Para que después de alcanzar las posiciones finales
                            ; no se sigan realizando movimientos adicionales,
                            ; aquí se ha insertado una parada.
N130 X10 Y30 Z40            ; Se vuelve a alcanzar la posición de N110.
                            ; El decalaje de origen está activo otra vez.
…
```

Nota

Si la función "Gestión de herramientas con almacenes" está activa, la función auxiliar `T...` o `M...` (típicamente `M6`) no basta para provocar el bloqueo del cambio de secuencia al final del movimiento `G75`.

Motivo: con el ajuste "Gestión de herramientas con almacenes" activo, las funciones auxiliares para el cambio de herramienta no se envían al PLC.

Información adicional

G75

Los ejes se desplazan en rápido como ejes de máquina. El movimiento se refleja internamente mediante las funciones "SUPA" (supresión de todos los frames) y "G0 RTLIOF" (desplazamiento en rápido con interpolación de ejes individuales).

Si no se cumplen las condiciones para "RTLIOF" (interpolación de ejes individuales), en la trayectoria se realiza el desplazamiento al punto fijo.

Al alcanzar el punto fijo, se detienen los ejes dentro de la ventana de tolerancia "Parada precisa fina".

Dinámica parametrizable para G75

Para desplazamientos de posicionado a punto fijo (G75) puede ajustarse el modo dinámico deseado mediante el siguiente dato de máquina:

MD18960 $MN_POS_DYN_MODE (tipo de dinámica del eje de posicionamiento)

Movimientos adicionales por eje

Los siguientes movimientos adicionales por eje se tienen en cuenta en el momento de interpretar la secuencia G75:

- Decalaje de origen externo
- DRF
- Offset de sincronización ($AA_OFF)

Después, los movimientos adicionales de los ejes no deben modificarse hasta que se haya alcanzado el final del desplazamiento mediante la secuencia G75.

Los movimientos adicionales después de la interpretación de la secuencia G75 provocan el correspondiente decalaje del punto fijo al que se ha realizado el desplazamiento.

Los siguientes movimientos adicionales no se tienen en cuenta independientemente del momento de la interpretación y provocan el correspondiente decalaje de la posición de destino:

- Corrección de herramienta online
- Movimientos adicionales de ciclos de compilación en el BKS igual que el MKS

Frames activos

Se ignoran todos los frames activos. El desplazamiento se realiza en el sistema de coordenadas de máquina.

Limitación del campo de trabajo en el WKS/ENS

La limitación del campo de trabajo específica del sistema de coordenadas (WALCS0 ... WALCS10) no actúa en la secuencia con G75. El punto final se vigila como punto inicial de la secuencia siguiente.

Movimiento del eje/cabezal con POSA/SPOSA

Si antes se han desplazado ejes/cabezales programados con POSA o SPOSA, estos movimientos llegan primero hasta el final antes de alcanzar el punto fijo.

Funciones de cabezal en la secuencia G75

Si el cabezal está excluido del "desplazamiento a punto fijo", pueden programarse también funciones de cabezal en la secuencia G75 (p. ej., posicionado con SPOS/SPOSA).

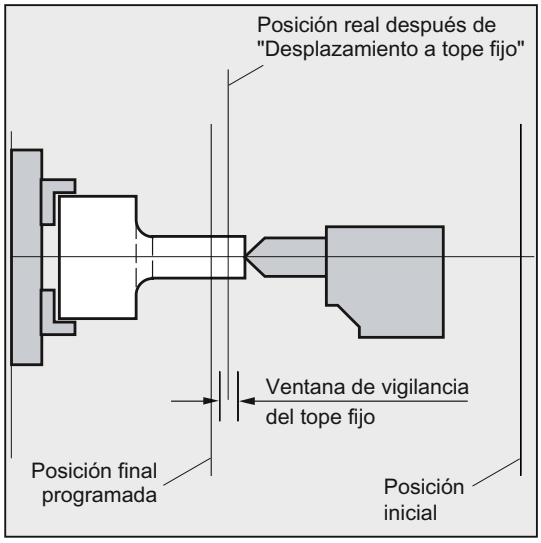
Ejes de módulo

En ejes de módulo, el desplazamiento al punto fijo se realiza por el recorrido más corto.

2.14.6 Desplazamiento a tope fijo (FXS, FXST, FXSW)

Función

Mediante la función "Desplazamiento a tope fijo" es posible generar la fuerza necesaria para el amarre de piezas, p. ej. para las garras, la pínula o las pinzas. Además, con esta función se puede realizar el desplazamiento a puntos de referencia mecánicos.



Con pares muy reducidos también se pueden realizar procesos de medida, evitando la necesidad de conectar un palpador. La función "Desplazamiento a tope fijo" se puede utilizar para ejes y también para cabezales utilizados como ejes.

Sintaxis

```
FXS[<Eje>]=…
```

```
FXST[<Eje>]=…
FXSW[<Eje>]=…
FXS[<Eje>]=… FXST[<Eje>]=…
FXS[<Eje>]=… FXST[<Eje>]=… FXSW[<Eje>]=…
```

Descripción

`FXS`:	Comando de activación y desactivación de la función "Desplazamiento a tope fijo"
	`FXS[<Eje>]=1`: Activar función
	`FXS=[<Eje>]=0`: Desactivar función
`FXST`:	Comando opcional para ajustar el par de apriete
	Indicación en % del par máximo del accionamiento.
`FXSW`:	Comando opcional para ajustar el ancho de la ventana para la vigilancia de tope fijo
	Dato en mm, pulgadas o grados.
`<Eje>`:	Nombre del eje de máquina
	Se programan los ejes de máquina (X1, Y1, Z1, etc.).

Nota

Los comandos `FXS`, `FXST` y `FXSW` son modalmente activos.

La programación de `FXST` y `FXSW` es opcional: si no existe ninguna indicación, se aplica el último valor programado o el valor ajustado en el correspondiente dato de máquina.

Activar desplazamiento a tope fijo: FXS[<Eje>] = 1

El desplazamiento al punto indicado se puede definir como un movimiento de un eje de contorneado o bien de un eje de posicionado. Los ejes de posicionado pueden realizar su desplazamiento incluso tras el final de secuencia.

Se puede programar el desplazamiento a tope fijo de varios ejes de forma simultánea al desplazamiento de otros ejes. El tope fijo debe encontrarse entre las posiciones inicial y final.

ATENCIÓN

Peligro de colisión

Durante el desplazamiento de un eje/cabezal a tope fijo no se puede programar para ese mismo eje/cabezal ninguna posición nueva.

Antes de seleccionar la función, los cabezales deben de conmutarse al modo regulado en posición.

Ejemplo:

Código de programa	**Comentarios**
`X250 Y100 F100 FXS[X1]=1 FXST[X1]=12.3 FXSW[X1]=2`	`; El eje X1 se desplaza con el avance F100 (indicación opcional) a la posición de destino X=250 mm.` `El par máximo admisible será del 12.3% del máximo par admisible para el accionamiento. La vigilancia se realiza dentro de una ventana de 2 mm de ancho.`
`...`	

Desactivar desplazamiento a tope fijo: FXS[<Eje>] = 0

La desactivación de la función provoca una parada de pretratamiento.

En la secuencia con `FXS[<Eje>]=0` puede y debe haber movimientos de desplazamiento.

ATENCIÓN

Peligro de colisión

El desplazamiento de retirada se debe realizar alejándose del tope fijo; en caso contrario, se puede provocar un daño en la máquina o en el tope fijo.

El cambio de secuencia se realiza cuando se ha alcanzado la posición de retirada. Si no se indica una posición de retirada, se produce inmediatamente y la limitación de par también se desactiva de forma inmediata.

Ejemplo:

Código de programa	**Comentarios**
`X200 Y400 G01 G94 F2000 FXS[X1] = 0`	`; El eje X1 se retira del tope fijo hasta la posición X=200 mm. El resto de parámetros son opcionales.`
`...`	

Par de apriete (FXST) y ventana de vigilancia (FXSW)

Una limitación de par programada `FXST` actúa desde el comienzo de la secuencia, por lo que también el desplazamiento al tope se realiza con un par reducido. `FXST` y `FXSW` pueden programarse o modificarse en cualquier momento en el programa de pieza. Los cambios se hacen efectivos antes de realizar los desplazamientos en la misma secuencia.

ATENCIÓN
Peligro de colisión La programación de una nueva ventana de vigilancia para el tope fijo provoca el cambio no solo del ancho de la ventana, sino también del punto de referencia para el centro de la ventana si el eje se ha movido antes. El centro de la nueva ventana de vigilancia será la posición actual del eje de máquina en el momento del cambio de ventana. La ventana de vigilancia se debe definir de tal manera que la vigilancia solamente se active en el caso de que se corra el tope fijo.

Información adicional

Rampa ascendente

A través de un dato de máquina se puede definir una rampa ascendente para el nuevo límite de par con el fin de evitar el ajuste brusco del límite de par (p. ej.: al introducir una caña del contrapunto).

Supresión de alarmas

En aplicaciones, la alarma de tope puede suprimirse desde el programa de pieza enmascarando la alarma en un dato de máquina y activando el nuevo ajuste del DM con NEW_CONF.

Activación

Los comandos para el desplazamiento al tope fijo se pueden llamar desde acciones síncronas/ciclos tecnológicos. La activación puede tener lugar también sin movimiento; el par se limita inmediatamente. Tan pronto como el eje se mueve hacia el lado del valor nominal, se controla en cuanto al tope.

Activar desde acciones síncronas

Ejemplo:

Cuando se produce el suceso esperado ($R1) y el desplazamiento al tope fijo no está ya en marcha, debe activarse `FXS` para el eje Y. El par debería ser del 10% del par nominal. Para el ancho de la ventana de vigilancia se aplica el valor predefinido.

Código de programa

```
N10 IDS=1 WHENEVER (($R1=1) AND ($AA_FXS[Y]==0)) DO $R1=0 FXS[Y]=1
FXST[Y]=10
```

El programa de pieza normal tiene que asegurar que $R1 se establezca en el momento deseado.

Desactivar desde acciones síncronas

Ejemplo:

Si se produce un suceso esperado ($R3) y existe el estado "Tope alcanzado" (variable de sistema $AA_FXS), se debería cancelar FXS.

Código de programa

```
IDS=4 WHENEVER (($R3==1) AND ($AA_FXS[Y]==1)) DO FXS[Y]=0
FA[Y]=1000 POS[Y]=0
```

Tope fijo alcanzado

Cuando se ha alcanzado el tope fijo:

- Se borra el trayecto residual y se modifica el valor de consigna para la posición.
- El par de accionamiento aumenta hasta el valor límite programado `FXSW` y después permanece constante.
- La vigilancia del tope fijo se activa dentro de la amplitud de ventana existente.

Condiciones

- Medición con borrado de trayecto residual
  Las funciones "Medición con borrado del trayecto residual" (comando `MEAS`) y "Desplazamiento a tope fijo" no pueden programarse a la vez en una secuencia.
  Excepción: Una de las funciones se refiere a un eje de contorneado y la otra a un eje de posicionado o bien las dos funciones se refieren a ejes de posicionado.
- Vigilancia del contorno
  No se realiza vigilancia del contorno mientras la función "Desplazamiento a tope fijo" esté activada.
- Ejes de posicionado
  En "Desplazamiento a tope fijo" con ejes de posicionado, el cambio de secuencia se realiza con independencia del movimiento al tope fijo.
- Ejes de acoplamiento y de contenedor
  El desplazamiento a tope fijo también es admisible para ejes de acoplamiento y de contenedor.
  El estado del eje de máquina asignado se conserva más allá de varios giros del contenedor. Esto también es válido para la limitación modal de par con FOCON.
- **No** se puede realizar el desplazamiento a tope fijo:
  - Para ejes Gantry
  - Para ejes de posicionado concurrentes, controlados exclusivamente desde el PLC (la activación de `FXS` debe realizarse desde el programa CN).
- Si el límite del par se reduce demasiado, el eje ya no puede seguir la variación de consigna; el regulador entra en limitación y la desviación del contorno aumenta. En este estado operativo se pueden producir movimientos bruscos si se aumenta el límite del par. Para asegurar que el eje pueda seguir adelante debe comprobarse que la desviación del contorno no sea mayor que con el par sin limitación.

2.14.7 Tiempo de espera (G4)

Con el comando `G4` se programa un tiempo (tiempo de espera) en una secuencia que transcurre cuando se mecaniza la secuencia en el proceso principal. El cambio a la siguiente secuencia se produce una vez que ha transcurrido todo el tiempo.

Nota

`G4` interrumpe el modo de contorneado.

Sintaxis

```
G4 F<Time>
G4 S<NumSpi>
G4 S<n> = <NumSpi>
```

Descripción

`G4`:	Activar tiempo de espera	
	Único elemento de la secuencia:	sí
`F<Time>`:	El tiempo de espera `<Time>` se programa en segundos bajo la dirección `F`.	
`S<NumSpi>`:	El tiempo de espera se indica en vueltas de cabezal `<NumSpi>` bajo la dirección S en relación con el cabezal principal actual.	
`S<n>=<NumSpi>`:	El tiempo de espera se indica en vueltas de cabezal `<NumSpi>` bajo la dirección S en relación con el cabezal direccionado con la dirección extendida `<n>`.	

Nota

Las direcciones `F` y `S` utilizadas en la secuencia de tiempo de espera `G4` para la indicación del tiempo no influyen en los avances `F...` ni en las velocidades de giro del cabezal `S...` del programa.

Condiciones

Acciones síncronas

En un programa existen dos acciones síncronas programadas de manera que la siguiente secuencia con tiempo de espera se convierte en la secuencia de acciones en la que se ejecutan las acciones síncronas. Una es una acción síncrona modal; la otra, una acción síncrona secuencial. Si la acción síncrona secuencial debe influir en la acción síncrona modal, por ejemplo, para habilitar el mecanizado mediante UNLOCK, debe haber disponibles **por lo menos dos ciclos de interpolador**, por ejemplo, `G4 F<Ciclo de interpolador * 2>`, como tiempo de espera activo.

El tiempo de espera activo depende del ajuste en el dato de máquina MD10280 $MN_PROG_FUNCTION_MASK, bit 4 = <Valor>

Valor	Descripción
0	El tiempo de espera activo es igual al tiempo de espera programado
1	El tiempo de espera activo es igual al tiempo de espera programado redondeado al múltiplo inmediatamente superior del ciclo de interpolador (MD10071 $MN_IPO_CYCLE_TIME)

Ejemplo de programación:

- MD10071 $MN_IPO_CYCLE_TIME == 8 ms
- MD10280 $MN_PROG_FUNCTION_MASK, bit 4 = 1

```
Código de programa                          Comentarios
N10 WHEN TRUE DO LOCK(1)                    ; SynAct secuencial: LOCK de
                                            ; SynAct modal ID=1
N20 G4 F2                                   ; Secuencia de acciones para SynAct de N10
N30 WHEN TRUE DO UNLOCK(1)                  ; SynAct secuencial: UNLOCK
                                            ; de SynAct modal ID=1
N40 ID=1 WHENEVER TRUE DO $R0=1 RDISABLE    ; SynAct modal ID=1
                                            ; Parámetro R R0=1
                                            ; Activar el bloqueo de lectura
N50 G4 F0.012                               ; Secuencia de acciones para SynAct de
                                            N40 y N50
                                            ; Ver más adelante el apartado "Descrip-
                                            ción"
N60 G4 F10
```

Descripción

El comportamiento deseado es que la acción síncrona secuencial de N30 retire el bloqueo activo (LOCK) de la acción síncrona modal con ID=1 de N40 y, de esa manera, escriba el parámetro R en N50 y se active el bloqueo de lectura. No obstante, este comportamiento solo se alcanza si el tiempo de espera activo posee una duración mínima de dos ciclos de interpolador.

El tiempo de espera activo se obtiene a partir del tiempo de espera programado, el ciclo de interpolador y el ajuste en MD10280 $MN_PROG_FUNCTION_MASK, bit 4. Para que el tiempo de espera activo posea una duración mínima de dos ciclos de interpolador, debe programarse el siguiente tiempo de espera:

- Bit 4 == 0: tiempo de espera programado ≥ 2 * ciclo de interpolador
- Bit 4 == 1: tiempo de espera programado ≥ 1,5 * ciclo de interpolador

Si el tiempo de espera activo es inferior a dos ciclos de interpolador, la escritura del parámetro R y el bloqueo de lectura no se ejecutarán hasta la secuencia N60.

Ejemplo

```
Código de programa          Comentarios
N10 G1 F200 Z-5 S300 M3     ; Avance F, velocidad de giro del cabezal S
```

Código de programa	Comentarios
N20 G4 F3	; Tiempo de espera: 3 s
N30 X40 Y10	
N40 G4 S30	; Esperar 30 vueltas del cabezal (con S = 300 r/min y 100% de corrección de velocidad de giro corresponde a: t = 0,1 min).
N50 X...	; El avance programado en N10 y la velocidad de giro de cabezal siguen activos.

2.14.8 Parada de decodificación interna

Función

Cuando se accede a los datos de estado de la máquina ($A...), el control numérico genera una parada interna de decodificación previa. La secuencia subsiguiente sólo se ejecuta cuando se hayan ejecutado completamente todas las secuencias previamente preparadas y memorizadas. La secuencia anterior se detiene en la parada precisa (como G9).

Ejemplo:

Código del programa	Comentarios
...	
N40 POSA[X]=100	
N50 IF $AA_IM[X]==R100 GOTOF LABEL1	; Acceso a los datos de estado de la máquina ($A...); el control genera una parada de decodificación previa interna.
N60 G0 Y100	
N70 WAITP(X)	
N80 MARCA1:	
...	

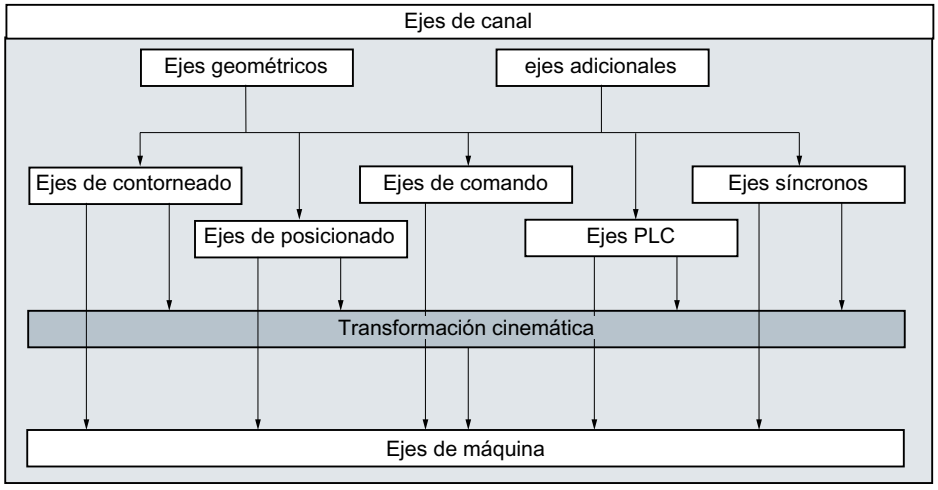
2.15 Otra información

2.15.1 Ejes

Tipos de eje

En el contexto de programación se distinguen los siguientes tipos de eje:

- Ejes principales/ejes geométricos
- Ejes adicionales
- Cabezal principal, cabezal maestro
- Ejes de máquina
- Ejes de canal
- Ejes de contorneado
- Ejes de posicionado
- Ejes síncronos
- Ejes de comando
- Ejes PLC/Ejes de posicionado concurrentes
- Ejes lincados o Link
- Ejes lincados guía



2.15.1.1 Ejes principales/Ejes geométricos

Los ejes principales forman un sistema de coordenadas cartesiano dextrógiro. Los desplazamientos de la herramienta se programan en este sistema de coordenadas.

En control numérico, los ejes principales se definen como ejes geométricos. Este concepto será utilizado en el presente manual de programación.

Ejes geométricos conmutables

Con la función "Ejes geométricos conmutables" (ver Manual de funciones Preparación del trabajo) se pueden modificar conjuntos de ejes geométricos configurados mediante un dato de máquina desde el programa de pieza. Así, un eje de canal definido como eje adicional síncrono puede sustituir a cualquier eje geométrico.

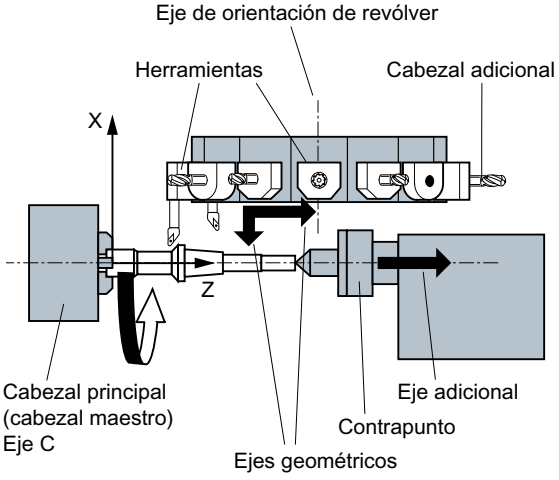
Identificador de eje

El identificador/nombre de un eje geométrico se puede definir mediante el dato de máquina siguiente:

MD20060 $MC_AXCONF_GEOAX_NAME_TAB (nombre de eje geométrico en el canal)

Denominación estándar en tornos:

1. Eje geométrico: X
2. Eje geométrico: Z



Denominación estándar en fresadoras:

1. Eje geométrico: X
2. Eje geométrico: Y
3. Eje geométrico: Z

Información adicional

Como máximo se pueden utilizar tres ejes geométricos para la definición y programación de frames y de geometrías de pieza (contornos).

Los identificadores para ejes geométricos y de canal pueden ser iguales si es posible hacerlos corresponder.

Los nombres de ejes geométricos y de canal pueden ser iguales en todos los canales. De este modo, un programa se puede ejecutar en cualquier canal.

2.15.1.2 Ejes adicionales

Al contrario que los ejes geométricos, los ejes adicionales no tienen ninguna relación con otros ejes.

Los ejes adicionales típicos son:

- Ejes de revólver de herramientas
- Ejes de mesa basculante
- Ejes de cabezal basculante
- Ejes de cargador

Identificador de eje

En un torno con almacén de torreta p. ej.:

- Posición de la torreta U
- Contrapunto V

Ejemplo de programación

Código de programa	Comentarios
`N10 G1 X100 Y20 Z30 A40 F300`	`; Movimientos del eje de contorneado.`
`N20 POS[U]=10POS[X]=20 FA[U]=200 FA[X]=350`	`; Movimientos del eje de posicionado.`
`N30 G1 X500 Y80 POS[U]=150FA[U]=300 F550`	`; Eje de contorneado y de posicionado.`
`N40 G74 X1=0 Z1=0`	`; Búsqueda del punto de referencia.`

2.15.1.3 Cabezal, cabezal maestro

La cinemática de la máquina determina cuál de los cabezales es el principal. Este cabezal se declara generalmente con datos de máquina como cabezal maestro.

La asignación se puede modificar con el comando de programa `SETMS (<Número de cabezal>)`. Con `SETMS` sin indicación del número de cabezal se puede volver a conmutar al cabezal maestro definido en el dato de máquina.

Para el cabezal maestro se aplican funciones especiales, como p. ej. el tallado de roscas.

Identificador de cabezal

S o S0

2.15.1.4 Ejes de máquina

Los ejes de máquina son los ejes que existen físicamente en la máquina.

El desplazamiento programado de un eje de contorneado o de un eje adicional puede tener efecto en varios ejes de máquina a través de una transformada activa en el canal (TRANSMIT, TRACYL o TRAORI).

Los ejes de máquina solo se activan directamente en el programa en casos especiales (p. ej. búsqueda de punto de referencia o desplazamiento a punto fijo).

Identificador de eje

El identificador/nombre de un eje de máquina se puede definir mediante el siguiente dato de máquina específico del CN:

MD10000 $MN_AXCONF_MACHAX_NAME_TAB (nombre del eje de máquina)

Ajuste estándar: X1, Y1, Z1, A1, B1, C1, U1, V1

Los ejes de máquina tienen además identificadores de eje fijos que pueden utilizarse siempre, con independencia del nombre definido en el dato de máquina.

AX1, AX2, …, AX<n>

2.15.1.5 Ejes de canal

Se llaman ejes de canal todos los ejes geométricos, adicionales y de máquina asignados a un canal.

Identificador de eje

El nombre/identificador específico del canal de un eje geométrico o un eje adicional se puede definir con el dato de máquina siguiente:

MD20080 $MC_AXCONF_CHANAX_NAME_TAB (nombre de eje de canal)

Ajuste estándar: X, Y, Z, A, B, C, U, V

La asignación del eje de máquina que representa a un eje geométrico o un eje adicional en el canal se define en el dato de máquina siguiente:

MD20070 $MC_AXCONF_MACHAX_USED (ejes de máquina utilizados)

2.15.1.6 Ejes de contorneado

Los ejes de contorneado describen la trayectoria y, por lo tanto, los movimientos de la herramienta en el espacio.

El avance programado se mantiene a lo largo de dicha trayectoria. Los ejes implicados en dicha trayectoria llegan simultáneamente a su posición. Generalmente se trata de los ejes geométricos.

Se predefine qué ejes van a ser ejes de contorneado y, por lo tanto, determinantes de la velocidad.

En el programa CN, pueden indicarse ejes de contorneado con `FGROUP`.

Para más información acerca de `FGROUP`, ver "Avance (G93, G94, G95, F, FGROUP, FL, FGREF) (Página 108)".

2.15.1.7 Ejes de posicionado

Los ejes de posicionado se interpolan separadamente; esto es, cada eje de posicionado tiene su propio interpolador de eje y su propio avance. Los ejes de posicionado no interpolan con los ejes de contorneado.

Los ejes de posicionado se mueven desde el programa de CN o desde el PLC. En el caso de que se mueva el mismo eje simultáneamente desde el programa de CN y desde el PLC se presenta un aviso de error.

Ejes de posicionado típicos son:

- Cargador para la carga de piezas
- Cargador para la descarga de piezas
- Almacén de herramientas/torreta revólver

Tipos

Se diferencia entre ejes de posicionado con desplazamiento simultáneo hasta el final de la secuencia o durante varias secuencias.

Ejes POS

El cambio de secuencia se realiza al final de la secuencia cuando todos los ejes de contorneado y posicionado programados en dicha secuencia hayan alcanzado sus posiciones.

Ejes POSA

Los desplazamientos de dichos ejes pueden extenderse por varias secuencias.

Ejes POSP

El movimiento de dichos ejes de posicionamiento para el desplazamiento a la posición final se realiza por tramos.

Nota

Los ejes de posicionado se pueden tratar como si fuesen ejes síncronos cuando se desplazan sin la instrucción POS/POSA.

G64 (modo Contorneado) solamente se puede aplicar a los ejes de contorneado cuando los ejes de posicionado (POS) alcanzan su posición programada antes que aquéllos.

Los ejes de contorneado programados con `POS`/`POSA` se eliminan del conjunto de ejes de contorneado para esta secuencia.

Para más información acerca de `POS`, `POSA` y `POSP`, ver "Desplazar ejes de posicionado (POS, POSA, POSP, FA, WAITP, WAITMC) (Página 116)".

2.15.1.8 Ejes síncronos

Los ejes síncronos se desplazan de forma simultánea a la trayectoria desde la posición inicial hasta la posición final programada.

El avance programado bajo `F` es válido para todos los ejes de contorneado programados en la secuencia, pero no lo es para los ejes síncronos. Los ejes síncronos necesitan el mismo tiempo que los ejes de contorneado para realizar su desplazamiento.

Un eje síncrono puede ser, por ejemplo, un eje giratorio que se desplaza de forma simultánea a la trayectoria.

2.15.1.9 Ejes de comando

Los ejes de comando se arrancan por acciones síncronas derivadas de un evento (comandos). Se pueden posicionar, arrancar y parar en forma completamente asíncrona al programa de pieza. Un eje no puede ser desplazado simultáneamente por el programa de pieza y por acciones síncronas.

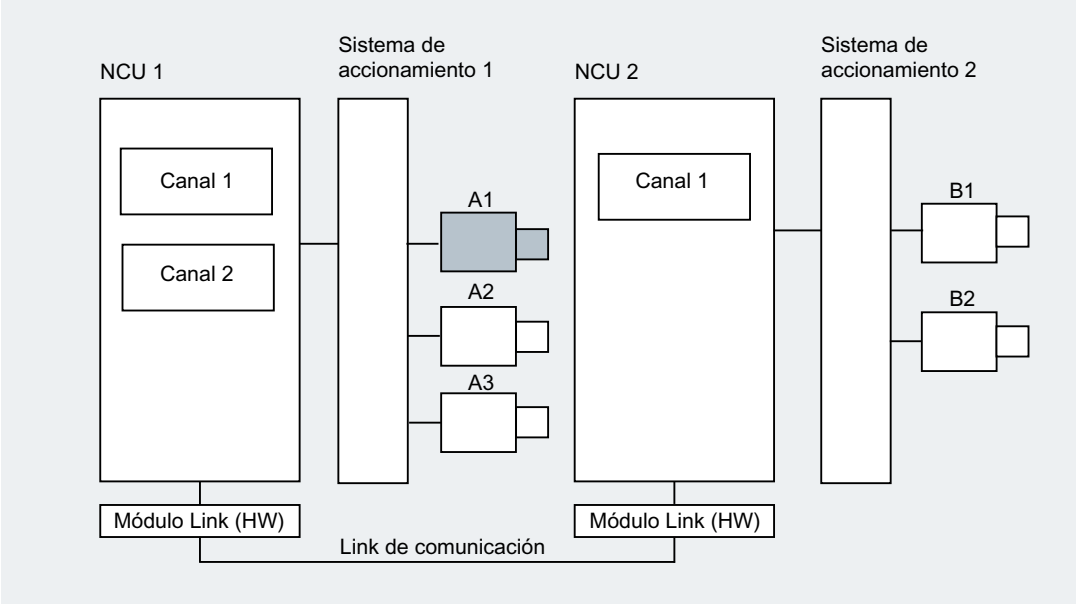
Los ejes de comando se interpolan separadamente, esto es, cada eje de comando tiene su propio interpolador de eje y su propio avance.

2.15.1.10 Ejes PLC

Los ejes PLC son desplazados en el programa básico por el PLC a través de bloques de función (módulos software) especiales y se pueden desplazar en forma asíncrona al resto de los ejes. Estos desplazamientos se realizan de forma completamente independiente a movimientos interpolados o síncronos.

2.15.1.11 Ejes lincados

Los ejes lincados o Link son ejes que están conectados físicamente a otra NCU que realiza la regulación de posición. Los ejes lincados se pueden asignar a canales dinámicos de **otra** NCU. Desde el punto de vista de una NCU determinada, los ejes lincados son ejes no locales.



El concepto de **contenedor de ejes** rige para la asignación dinámica a una determinada NCU. El intercambio de ejes con `GET` y `RELEASE` desde el programa de pieza **no** está disponible para ejes lincados.

Información adicional

Requisitos

- Las NCU implicadas NCU1 y NCU2 deberán estar acopladas mediante el módulo de lincado con comunicación de lincado rápida.
- El eje se debe configurar pertinentemente mediante datos de máquina.
- Debe estar presente la opción "Eje lincado".

Descripción

La regulación de posición se realiza en la NCU a la que el eje está físicamente unido con el accionamiento. Aquí se encuentra también la correspondiente interfaz VDI del eje. Los valores de consigna de posición para ejes lincados se generan en otra NCU y se transfieren utilizando NCU-Link.

La comunicación de lincado se encarga de la interacción entre los interpoladores y el regulador de posición o bien la interfaz PLC. Los valores de consigna calculados por los interpoladores se deben transferir al lazo de regulación de posición de la NCU propia, o bien los valores reales se deben transferir en sentido inverso.

Contenedor de ejes

Un contenedor de ejes consiste en una estructura de búfer circular de datos en la que se realiza la asignación de ejes locales y/o ejes lincados a canales. Los datos introducidos en el búfer circular se pueden **desplazar cíclicamente**.

En la imagen lógica de ejes de máquina, junto con la referencia directa a los ejes locales o los ejes lincados, la configuración de ejes lincados permite la referencia a contenedores de ejes. Una referencia de este tipo se compone de:

- Número de contenedor **y**
- Slot (puesto del búfer circular en el contenedor correspondiente)

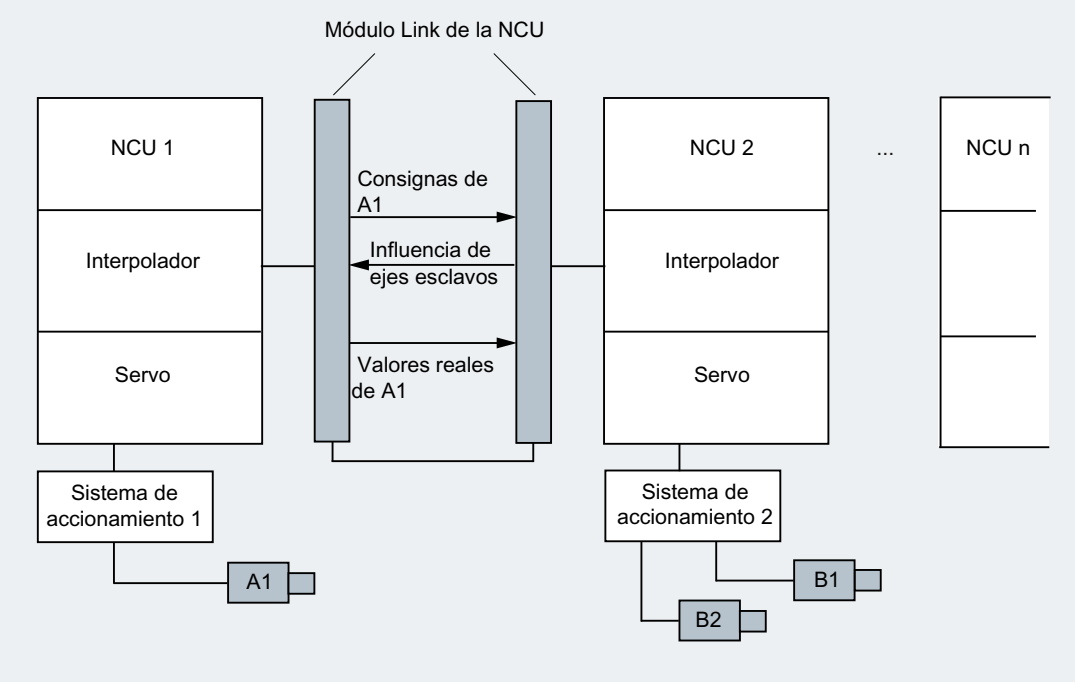
Se puede introducir como puesto de un búfer circular:

- Un eje local **o**
- Un eje lincado

Si se considera una sola NCU, las introducciones en el contenedor de ejes contienen ejes de máquina locales o ejes lincados. Las entradas en la imagen lógica de ejes de máquina (MD10002 $MN_AXCONF_LOGIC_MACHAX_TAB) de una sola NCU son fijas.

2.15.1.12 Ejes lincados guía

Un eje lincado guía es un eje interpolado por una NCU y utilizado por una o varias otras NCUs como eje guía o maestro para la conducción de ejes de seguimiento (esclavos).



Una alarma de regulador de posición de eje se distribuye a todas las demás NCU relacionadas con el eje afectado a través de un eje lincado guía.

Las NCU dependientes del eje lincado guía pueden utilizar los siguientes acoplamientos al eje lincado guía:

- Valor guía (valor guía de consigna, real, simulado)
- Arrastre
- Seguimiento tangencial
- Reductor electrónico
- Cabezal síncrono

Programación

NCU guía:

Sólo la NCU asignada físicamente al eje de valor guía puede programar movimientos de desplazamiento para este eje. Por lo demás, la programación no necesita tener en cuenta ninguna particularidad.

NCUs de ejes de seguimiento:

La programación en la NCU de los ejes de seguimiento no debe contener comandos de desplazamiento para los ejes lincados guía (eje de valor guía). Las violaciones de esta regla producen una alarma.

El eje lincado guía se activa de la forma acostumbrada a través del identificador de eje de canal. Los estados del eje lincado guía son accesibles a través de variables de sistema seleccionadas.

Información adicional

Requisitos

- Las NCU implicadas NCU1 a NCU<n> (<n> máx. 8) deberán estar acopladas mediante el módulo de lincado con comunicación de lincado rápida.
  Bibliografía:
  Manual de producto Configuración NCU
- El eje se debe configurar pertinentemente mediante datos de máquina.
- Debe estar presente la opción "Eje lincado".
- Para todas las NCU implicadas tiene que estar configurado el mismo ciclo de interpolador.

Limitaciones

- Un eje guía como eje lincado guía no puede ser un eje lincado, es decir, ser desplazado por otras NCU que no sean su NCU propia.
- Un eje guía como eje lincado guía no puede ser un eje de contenedor, es decir, ser activado alternativamente por distintas NCU.
- Un eje lincado guía no puede ser un eje guía programado de un conjunto Gantry.
- Acoplamientos con ejes lincados guía no se pueden conectar en serie en varios escalones (cascada).
- El intercambio de ejes sólo es posible dentro de la NCU propia del eje lincado guía.

Variables del sistema

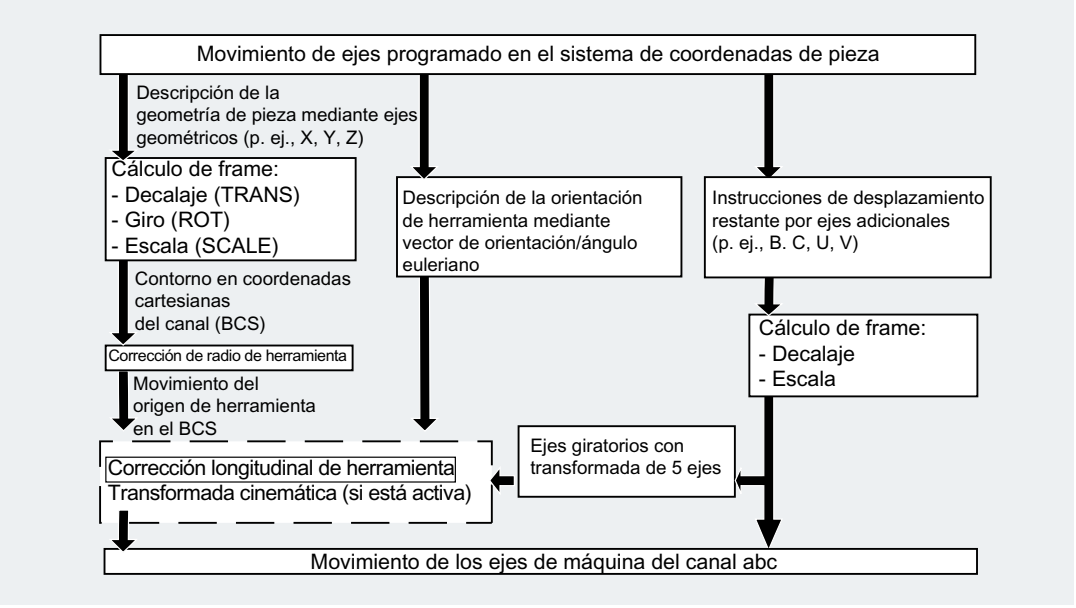
Las siguientes variables del sistema se pueden utilizar con el identificador de eje de canal del eje lincado guía:

Variable de sistema	Descripción
$AA_LEAD_SP	Valor maestro simulado, posición
$AA_LEAD_SV	Valor maestro simulado, velocidad

Si estas variables del sistema son actualizadas por la NCU del eje guía, los nuevos valores se transmiten también a las NCU que quieren desplazar ejes esclavos dependientes de este eje guía.

2.15.2 Del comando de desplazamiento al movimiento de la máquina

La siguiente figura muestra la relación entre los desplazamientos de eje programados (comandos de desplazamiento) y los movimientos de la máquina resultantes:

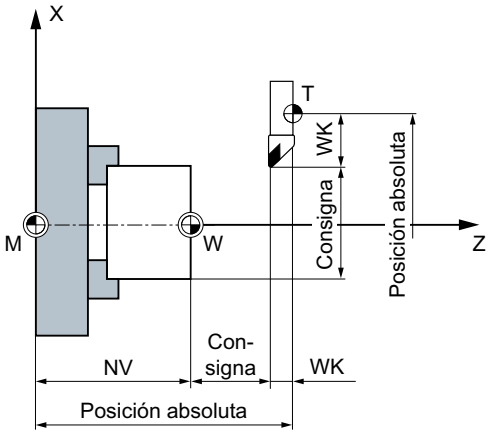


2.15.3 Cálculo del recorrido

La función de cálculo de recorrido determina el trayecto a recorrer en una secuencia, considerando todos los desplazamientos y correcciones.

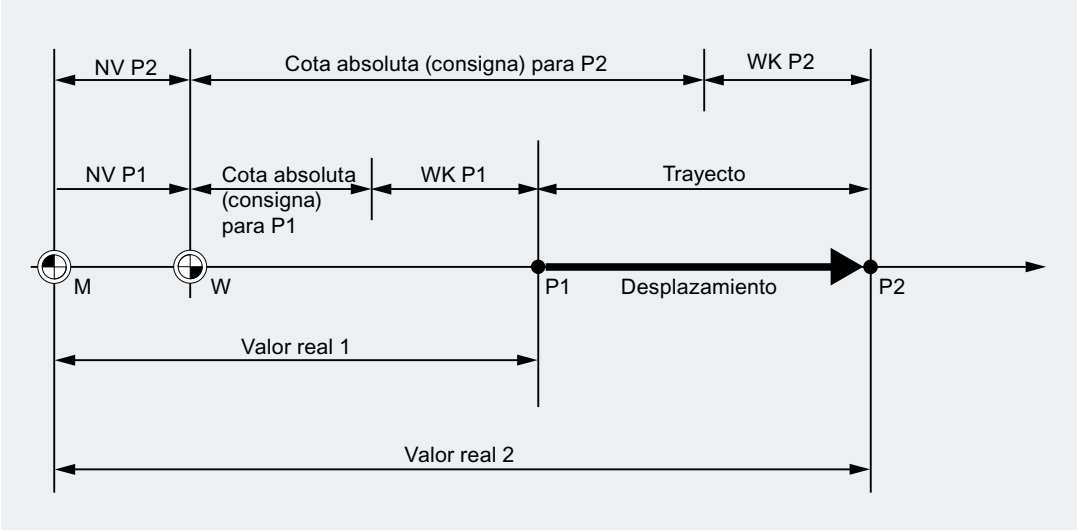
De aplicación general:

Recorrido =
consigna - valor real + decalaje de origen (DO) + corrección de herramienta



Si en una nueva secuencia de programación se programa un nuevo decalaje de origen y una nueva corrección de herramienta, entonces:

- Con acotado absoluto:
  Recorrido = (cota absoluta P2 - cota absoluta P1) + (DO P2 - DO P1) + (corr. herram. P2 - corr. herram. P1).
- Con acotado incremental:
  Recorrido = cota incremental + (DO P2 - DO P1) + (corr. herram. P2 - corr. herram. P1).



2.15.4 Direcciones

Direcciones fijas

Las direcciones tienen una configuración fija, es decir, no es posible modificar los caracteres de dirección.

En la tabla "Direcciones fijas (Página 1281)" se encuentra una lista.

Direcciones ajustables

El fabricante de la máquina herramienta puede asignarle otro nombre a estas direcciones a través de los datos de máquina (DM).

Nota

Las direcciones ajustables deben ser unívocas dentro del control, es decir, no puede utilizarse el mismo nombre de dirección para diferentes tipos de dirección (valores de eje y puntos finales, orientación de la herramienta, parámetros de interpolación, etc.).

En la tabla "Direcciones ajustables (Página 1286)" se encuentra una lista.

Direcciones activas modales/por secuencia

Las direcciones modales permanecen activas con el valor programado para el resto de secuencias sucesivas, hasta que se programe un nuevo valor bajo la misma dirección.

Las direcciones activas secuencia a secuencia sólo tienen validez en la secuencia en la que se han programado.

Ejemplo:

Código de programa	Comentarios
N10 G01 F500 X10	
N20 X10	; El avance F de N10 permanece activo hasta que se introduce otro nuevo.

Direcciones con extensión de eje

En las direcciones con extensión de eje, el nombre de éste figura entre corchetes tras la dirección que define la asignación del eje.

Ejemplo:

Código de programa	Comentarios
FA[U]=400	; Avance específico para el eje U.

Ver también la tabla "Direcciones fijas (Página 1281)".

Escritura de dirección extendida

La posibilidad de escribir direcciones extendidas permite incorporar en una sistemática numerosos ejes y cabezales.

Una dirección extendida está formada por una extensión numérica y una expresión aritmética asignada con el carácter "=". La dirección numérica extendida puede ser un número de una o dos cifras pero siempre positivo.

Sólo se permite programar direcciones extendidas para las siguientes direcciones:

Dirección	Descripción
X, Y, Z, …	Direcciones de ejes
I, J, K	Parámetro de interpolación
S	Velocidad de giro del cabezal
SPOS, SPOSA	Posición del cabezal
M	Funciones adicionales
H	funciones auxiliares
T	Número de herramienta
F	Avance

Ejemplos:

Código de programa	Comentarios
X7	; No se necesita el carácter "="; 7 es el valor; no obstante, también es posible utilizar aquí el carácter "="
X4=20	; Eje X4; se requiere el carácter "="
CR=7.3	; 2 letras; se requiere el carácter "="
S1=470	; Velocidad de giro para el 1.er cabezal: 470 r/min
M3=5	; Parada del 3.er cabezal

En las direcciones M, H y S, así como en SPOS y SPOSA, es posible sustituir la extensión numérica por una variable. En el caso de utilizar la variable, el identificador de la variable debe de ir entre corchetes.

Ejemplos:

Código de programa	Comentarios
S[SPINU]=470	; Velocidad de giro para el cabezal cuyo número se indica en la variable SPINU.
M[SPINU]=3	; Giro a derechas para el cabezal cuyo número se indica en la variable SPINU.
T[SPINU]=7	; Preselección de la herramienta para el cabezal cuyo número se indica en la variable SPINU.

2.15.5 Nombres

Los comandos según DIN 66025 se completan mediante el lenguaje de alto nivel CN con, entre otras cosas, objetos denominados.

Los objetos denominados pueden ser, p. ej.:

- Variables del sistema
- Variables definidas por el usuario
- Ejes/cabezales
- Subprogramas
- Palabras reservadas
- Metas (etiquetas) de salto
- Macros

Nota

Los descriptores tienen que ser unívocos. No se puede utilizar el mismo descriptor para objetos distintos.

Reglas de denominación

Los nombres pueden elegirse libremente de acuerdo con las siguientes reglas:

- Caracteres permitidos:
  - Letras: A ... Z, a ... z
  - Números: 0 ... 9
  - Carácter de subrayado: _
- Los dos primeros caracteres deberían ser letras o caracteres de subrayado.
- Longitud máxima:
  - Nombres de programa (Página 43): 24 caracteres
  - Nombres de eje: 8 caracteres
  - Nombres de variables: 31 caracteres

Nota

Las palabras reservadas no se pueden utilizar como identificadores.

Ciclos

Para evitar conflictos de nombres, se recomienda la siguiente definición a la hora de asignar nombres para ciclos de usuario:

Cadena de caracteres	Reservada para nombres de
• CYCLE • CUST_ • GROUP_ • _ • S_ • E_ • F_	Ciclos de SIEMENS
• CCS_	Ciclos compilados de SIEMENS
• CC_	Ciclos compilados del usuario

Ciclos de usuario

Se recomienda que los nombres de ciclos de usuario empiecen con "U_".

Variables

Encontrará información sobre la asignación de nombres de variables en los siguientes capítulos:

- Datos de sistema (Página 400)
- Definición de variables de usuario (DEF) (Página 407)

2.15.6 Constantes

Constante (gral.)

Una constante es un elemento de datos cuyo valor no varía con la ejecución de un programa; p. ej. una asignación de valores a una dirección.

Constante decimal

El valor numérico de una constante decimal se representa en el sistema decimal.

Constante INTEGER

Una constante INTEGER es un valor entero, es decir, una secuencia de cifras sin punto decimal con o sin signo.

Ejemplos:

`X10`	Asignación del valor +10 a la dirección X
`X-35`	Asignación del valor -35 a la dirección X
`X0`	Asignación del valor 0 a la dirección X **Nota:** X0 no puede sustituirse por X.

Constante REAL

Una constante REAL es una secuencia de cifras con punto decimal con o sin signo y con o sin exponente.

Ejemplos:

`X10.25`	Asignación del valor +10.25 a la dirección X
`X-10.25`	Asignación del valor -10.25 a la dirección X
`X0.25`	Asignación del valor +0.25 a la dirección X
`X.25`	Asignación del valor +0.25 a la dirección X, omitiendo el "0" antes de la coma
`X=-.1EX-3`	Asignación del valor $-0.1*10^{-3}$ a la dirección X

Nota

Si para una determinada dirección se introducen más cifras que las máximas permitidas detrás del punto decimal, se eliminan las cifras sobrantes y se redondea dicho valor por el último decimal permitido.

Constantes hexadecimales

También son posibles las constantes que se interpretan de forma hexadecimal, es decir, con base 16. Para ello se utilizan las letras A" a "F" como valores hexadecimales con los valores decimales del 10 al 15.

Los valores hexadecimales se utilizan entre comillas y comienzan con la letra inicial "H", seguida del valor hexadecimal. Se pueden utilizar caracteres separadores entre las letras y los números.

Ejemplo:

Código de programa	Comentarios
`$MC_TOOL_MANAGEMENT_MASK='H7F'`	`; Mediante asignación de la constante hexadecimal se establecen los bits del 0 al 7 en el dato de máquina.`

Nota

El límite para el máximo valor hexadecimal permitido viene definido por el rango de valores enteros permitidos para el tipo de datos utilizado.

Constantes binarias

También se pueden utilizar constantes cuya representación sea en formato binario. Para representar números de formato binario solamente se pueden utilizar las cifras "0" y "1".

Las constantes binarias se describen entre comillas y comienzan con la letra "B", seguida del valor en binario. Se pueden utilizar caracteres separadores entre las cifras.

Ejemplo:

Código de programa	Comentarios
`$MN_AUXFU_GROUP_SPEC='B10000001'`	`; Mediante asignación de constante binaria se ajustan los bits 0 y 7 en el dato de máquina.`

Nota

El límite para el máximo valor hexadecimal permitido viene definido por el rango de valores enteros permitidos para el tipo de datos utilizado.

2.15.7 Operadores y funciones de cálculo

Operadores

Operadores matemáticos

Las variable de sistema del tipo REAL o INT pueden combinarse entre sí con los siguientes operadores:

Operador	Descripción
+	Suma
-	Resta

Operador	Descripción
*	Multiplicación
/	• División en acciones síncronas.: INT/INT ⇒ **INT** • División en acciones síncronas con resultado REAL por uso de la función ITOR(): ITOR(INT)/ITOR(INT) ⇒ **REAL** • División en programas de CN: INT/INT ⇒ **REAL**
DIV	División de enteros: INT/INT ⇒ **INT**
MOD	La división por módulo (solo para tipo INT) da el resto de la división de INT Ejemplo: 3 MOD 4 = 3

Nota

Solo es posible combinar con operadores variables del mismo tipo.

Operadores de comparación

Operador	Descripción
=	igual
=/=	distinto
<	menor
>	mayor
<=	menor o igual
>=	mayor o igual

Operadores lógicos o booleanos

Operador	Descripción
NOT	NO
AND	Y
OR	O
XOR	O exclusivo

Operadores lógicos bit a bit

Operador	Descripción
B_OR	O bit a bit
B_AND	Y bit a bit
B_XOR	O exclusivo bit a bit
B_NOT	Negación bit a bit

Prioridad de los operadores

Los operadores tienen las siguientes prioridades durante el procesado en la acción síncrona (máxima prioridad: 1):

Prio.	Operadores	Descripción
1	NOT, B_NOT	Negación, negación bit a bit
2	*, /, DIV, MOD	Multiplicación, división
3	+, -	Suma, resta
4	B_AND	Y bit a bit
5	B_XOR	O exclusivo bit a bit
6	B_OR	O bit a bit
7	AND	Y
8	XOR	O exclusivo
9	OR	O
10	<<	Concatenación de caracteres, resultado tipo STRING
11	==, <>, <, >, >=, <=	Operadores de comparación

Nota

Si se usan varios operadores en una expresión se recomienda encarecidamente priorizar inequívocamente los diferentes operadores por medio de paréntesis "(…)".

Ejemplo de condición usando una expresión con varios operadores:

Código de programa

```
... WHEN ($AA_IM[X] > VALOR) AND ($AA_IM[Y] > VALOR1) DO ...
```

Funciones de cálculo

Operador	Descripción
SIN( )	Seno
COS( )	Coseno
TAN( )	Tangente
ASIN( )	Arcoseno
ACOS( )	Arcocoseno
ATAN2(,)	Arcotangente2
SQRT( )	Raíz cuadrada
ABS( )	Valor absoluto
POT( )	2.ª potencia (cuadrado)
TRUNC( )	Parte entera Precisiones en comandos de comparación ajustables con TRUNC
ROUND()	Redondeo a entero
LN( )	Logaritmo neperiano (logaritmo natural)
EXP( )	Función exponencial

Indexación

El índice de variable de sistema de tipo "Matriz de …" puede ser también una variable de sistema. En este caso el índice se evalúa también en la pasada principal usando del ciclo del interpolador.

Ejemplo

```
Código de programa
... WHEN ... DO $AC_PARAM[ $AC_MARKER[1] ] = 3
```

Limitaciones

- No se permite imbricar el indexado con más variables de sistema.
- El índice no debe formarse mediante variables de preprocesamiento. Por ejemplo, esto **no** está permitido, ya que $P_EP es una variable de preprocesamiento:
  $AC_PARAM[1] = $P_EP[$AC_MARKER[0]]

Consulte también

Funciones de cálculo (Página 453)

Operaciones (Página 453)

Preparación del trabajo 3

3.1 Programación flexible de CN

3.1.1 Variables

Las variables de las áreas de datos de sistema y datos de usuario, especialmente en combinación con las funciones de cálculo y las estructuras de control, permiten crear programas CN y ciclos con flexibilidad.

⚠ ADVERTENCIA
Daños personales y materiales por la modificación de variables
Cuando se utilizan variables en el programa CN, debe tenerse en cuenta que los operadores de la máquina o personas no autorizadas pueden modificar las variables y, con ello, influir en la ejecución del programa si cuentan con los derechos de acceso correspondientes. Como consecuencia, pueden producirse daños personales y materiales. • Para evitar los efectos negativos en la ejecución del programa que pueden derivarse de la modificación de variables, deben preverse las comprobaciones de datos pertinentes ("Input validation") en el programa CN.

- Datos de sistema
  En los datos de sistema se encuentran las variables predefinidas en el sistema. Estas variables poseen un significado definido. Las utiliza principalmente el software de sistema. El usuario puede leer y escribir estas variables en programas CN y en ciclos. Ejemplo: Datos de máquina, datos del operador, variables del sistema.
  A pesar de que el significado de un dato del sistema está predefinido, el usuario puede modificar las propiedades redefiniéndolas hasta cierto punto.
  Ver "Redefinición de datos de sistema, datos de usuario e instrucciones de programación del CN (REDEF) (Página 413)"
- Datos de usuario
  En los datos de usuario se encuentran las variables definidas por el usuario, cuyo significado solo lo determina él. El sistema no las evalúa.
  Los datos de usuario se dividen en:
  - Variables de usuario predefinidas
    Las variables de usuario predefinidas son variables ya definidas en el sistema cuyo número solo debe parametrizarse mediante datos de máquina. El usuario puede adaptar las propiedades de estas variables.
    Ver "Redefinición de datos de sistema, datos de usuario e instrucciones de programación del CN (REDEF) (Página 413)".
  - Variables definidas por el usuario
    Las variables definidas por el usuario son variables que define el usuario y que el sistema crea en el momento de la ejecución. El usuario es el único que establece su número, tipo de datos, visibilidad y todas las demás propiedades.
    Ver "Definición de variables de usuario (DEF) (Página 407)"

3.1.1.1 Datos de sistema

Los datos de sistema contienen las variables predefinidas en el sistema que, en programas CN y ciclos, permiten el acceso a la parametrización actual del control y a estados de máquina, control y proceso.

Variables de decodificación previa

Las variables de decodificación previa son datos de sistema que se leen y se escriben en el contexto de la decodificación previa, es decir, en el momento de la interpretación de la secuencia en la que está programada la variable. Las variables de decodificación previa no disparan ninguna parada de decodificación previa.

Variables de proceso principal

Las variables de proceso principal son datos de sistema que se leen o se escriben en el contexto del proceso principal, es decir, en el momento de la ejecución de la secuencia en la que está programada la variable. Las variables de proceso principal son:

- Variables que pueden programarse en acciones síncronas (lectura/escritura)
- Variables que pueden programarse en el programa CN y que disparan una parada de decodificación previa (lectura/escritura)
- Variables que pueden programarse en el programa CN y cuyo valor se determina en la decodificación previa, pero no se escribe hasta el proceso principal (de forma sincronizada con el proceso principal: sólo escritura)

Sistemática de prefijos

Para identificar los datos de sistema de una forma especial, normalmente se añade al nombre un prefijo que se compone del carácter $ seguido de una o dos letras y un guión bajo:

$ + 1. Letra	**Significado: Tipo de datos**
Datos preliminares (datos de sistema que se leen/escriben en la decodificación previa)	
$M	Datos de máquina[1)]
$S	Datos de operador, zonas protegidas[1)]
$T	Datos de gestión de herramientas
$P	Datos programados
$C	Variables de ciclos de envolvente ISO
$O	Datos de opciones
R	Parámetros R (parámetros de cálculo)[2)]
Datos de proceso principal (datos de sistema que se leen/escriben en el proceso principal)	
$$M	Datos de máquina[1)]
$$S	Datos de operador [1)]
$A	Datos actuales del proceso principal
$V	Datos del regulador de posición
$R	Parámetros R (parámetros de cálculo)[2)]
[1)] El tratamiento de los datos de máquina y de operador como variables de decodificación previa o de proceso principal depende de si se han escrito con uno o dos símbolos $. Cada usuario puede elegir la notación libremente. [2)] Al utilizar un parámetro R en el programa de pieza/ciclo como variable de decodificación previa, no se escribe ningún prefijo, p. ej. R10. Si se utilizan en una acción síncrona como variable de proceso principal, se escribe un carácter $ como prefijo, p. ej. $R10.	

2.ª letra	**Significado: Visibilidad**
N	Variable de usuario global de CN (C**N**)
C	Variable específica de canal (**C**hannel)
A	Variable específica de eje (**A**xis)

Condiciones

Excepciones de la sistemática de prefijos

Las siguientes variables de sistema difieren de la sistemática de prefijos mencionada anteriormente:

- $TC_...: La 2.ª letra C no hace referencia aquí a variables de sistema específicas de canal sino específicas del portaherramientas (TC = Tool Carrier)
- $P_ ...: Variables del sistema específicas del canal

Utilización de datos de máquina y de operador en acciones síncronas

Al utilizar datos de máquina y de operador en acciones síncronas, se puede determinar con el prefijo si el dato de máquina o de operador se lee/escribe de forma sincronizada con la decodificación previa o el proceso principal.

Si el dato permanece invariable durante el mecanizado, se puede leer de forma sincronizada con la decodificación previa. Para ello, el prefijo del dato de máquina o de operador se escribe con un carácter $:

```
ID=1 WHENEVER $AA_IM[z] < $SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Si el dato varía durante el mecanizado, se debe leer/escribir de forma sincronizada con el proceso principal. Para ello, el prefijo del dato de máquina o de operador se escribe con dos caracteres $:

```
ID=1 WHENEVER $AA_IM[z] < $$SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Nota

Escritura de datos de máquina y de operador

Al escribir un dato de máquina o de operador hay que prestar atención a que el nivel de acceso activo al ejecutar el programa de pieza/ciclo permita el acceso de escritura y que el efecto del dato sea "IMMEDIATE".

Consulte también

Variables (Página 399)

3.1.1.2 Variables de usuario predefinidas: Parámetros de cálculo específicos de canal (R)

Los parámetros de cálculo específicos de canal o parámetros R son variables de usuario predefinidas con la identificación R, definidas como matriz del tipo de datos REAL. Por motivos históricos, junto con la notación con índice de matriz, p. ej., `R[10]`, también se permite la notación sin índice de matriz, p. ej., `R10`, para los parámetros R.

Si se utilizan en acciones síncronas, debe anteponerse el carácter $, p. ej., `$R10`.

Sintaxis

En la utilización como variable de decodificación previa:

```
R<n>
R[<Expresión>]
```

En la utilización como variable de proceso principal:

```
$R<n>
$R[<Expresión>]
```

Descripción

R:	Identificador en la utilización como variable de decodificación previa, p. ej. en el programa de pieza	
$R:	Identificador en caso de uso como variable de proceso principal, p. ej., en acciones síncronas	
	Tipo:	REAL
	Rango de valores:	En notación no exponencial: ± (0.000 0001 ... 9999 9999) **Nota:** Se permite un máximo de 8 decimales.
		En notación exponencial: ± ($1*10^{-300}$ ... $1*10^{+300}$) **Nota:** • Notación: <Mantisa>EX<Exponente>, p. ej., 8.2EX-3 • Se permite un máximo de 10 caracteres, incluidos signo y punto decimal.
<n>:	Número del parámetro R	
	Tipo:	INT
	Rango de valores:	0 - MAX_INDEX **Nota** MAX_INDEX se obtiene a partir de la cantidad parametrizada de parámetros R: MAX_INDEX = (DM28050 $MN_MM_NUM_R_PARAM) - 1
<Expresión>:	Índice de matriz Como índice de matriz puede indicarse cualquier expresión mientras el resultado de la expresión pueda convertirse en el tipo de datos INT (INT, REAL, BOOL, CHAR)	

Ejemplo

Asignaciones a parámetros R y utilización de parámetros R en funciones matemáticas:

```
Código de programa          Comentarios
R0=3.5678                   ; Asignación en la decodificación previa
R[1]=-37.3                  ; Asignación en la decodificación previa
R3=-7                       ; Asignación en la decodificación previa
$R4=-0.1EX-5                ; Asignación en el proceso principal: R4 = -0.1 * 10^-5
$R[6]=1.874EX8              ; Asignación en el proceso principal: R6 = 1.874 * 10^8
R7=SIN(25.3)                ; Asignación en la decodificación previa
```

Código de programa	Comentarios
`R[R2]=R10`	`; Direccionamiento indirecto mediante parámetros R`
`R[(R1+R2)*R3]=5`	`; direccionamiento indirecto mediante expresión matem.`
`X=(R1+R2)`	`; Desplazar eje X a la posición resultante de la suma de R1 y R2`
`Z=SQRT(R1*R1+R2*R2)`	`; Desplazar eje Z a la posición raíz cuadrada(R1^2 + R2^2)`

Consulte también

Variables (Página 399)

3.1.1.3 Variables de usuario predefinidas: Parámetros de cálculo globales (RG)

Función

Además de los parámetros R específicos de canal, el usuario puede utilizar también parámetros R globales. Están presentes una vez dentro del control y pueden leerse/escribirse desde todos los canales.

Los parámetros R globales se utilizan, p. ej., para pasar información de un canal al siguiente. Otro ejemplo son los ajustes globales que deben evaluarse para todos los canales, como, p. ej., el saliente de la pieza en bruto respecto al cabezal.

Los parámetros R globales se leen y escriben a través de la interfaz de usuario o en el programa CN durante la decodificación previa. No es posible utilizarlos en acciones síncronas o ciclos tecnológicos.

Nota

Al leer y escribir parámetros R globales **no** se realiza ninguna sincronización entre canales.

Dado que la lectura y escritura se realizan durante la decodificación previa, el momento en que un valor escrito se hace efectivo en un canal partiendo de otro canal no está definido.

Ejemplo:

En el canal 1 se está ejecutando un rectificado con un parámetro R global como contador de rectificados. El canal 2 escribe en este parámetro R global un valor que provoca la cancelación del rectificado en el canal 1. Sin embargo, se siguen ejecutando todos los rectificados interpretados hasta ese momento en el canal 1 en la decodificación previa. El número de rectificados a que esto equivale no está definido y varía, entre otras cosas, en función de la carga de trabajo del canal.

La sincronización entre canales debe realizarla el propio usuario mediante una aplicación, p. ej., con marcas WAIT.

Sintaxis

Escritura en el programa CN

```
RG[<n>]=<Valor>
```

```
RG[<Expresión>]=<Valor>
```

Lectura en el programa CN

```
R...=RG[<n>]
R...=RG[<Expresión>]
```

Descripción

RG:	Nombre predeterminado de la dirección CN para parámetros R **globales** **Nota:** El nombre de la dirección CN se ajusta mediante MD15800 $MN_R_PARAM_NCK_NAME	
<n>:	Número del parámetro R global	
	Tipo:	INT
	Rango de valores:	0 ... MAX_INDEX **Nota** MAX_INDEX se obtiene a partir del número parametrizado de parámetros R globales: MAX_INDEX = (MD18156 $MN_MM_NUM_R_PARAM_NCK) - 1
<Expresión>:	Como índice de matriz puede indicarse cualquier expresión mientras el resultado de la expresión pueda convertirse en el tipo de datos INT (INT, REAL, BOOL, CHAR)	
<Valor>:	Valor del parámetro R global	
	Tipo:	REAL
	Rango de valores:	En notación no exponencial: ± (0.000 0001 ... 9999 9999) **Nota:** Se permite un máximo de 8 decimales.
		En notación exponencial: ± ($1*10^{-300}$ ... $1*10^{+300}$) **Nota:** • Notación: <Mantisa>EX<Exponente>, p. ej., 8.2EX-3 • Se permite un máximo de 10 caracteres, incluidos signo y punto decimal.

3.1.1.4 Variables de usuario predefinidas: Variables de lincado

En el marco de la función "Lincado NCU", las variables de lincado permiten el intercambio cíclico de datos entre NCU interconectadas en una red. En ese caso posibilitan un acceso específico del formato de datos a la memoria de variables de lincado. El usuario/fabricante de la máquina establece de forma específica para la instalación la memoria de variables de lincado, tanto en lo que se refiere a tamaño como a estructura de datos.

Las variables de lincado son variables de usuario globales del sistema que todas las NCU del conjunto de lincado pueden leer y escribir en programas de pieza y ciclos durante la comunicación de lincado configurada. Al contrario que las variables globales del usuario (GUD), las variables de lincado también pueden utilizarse en acciones síncronas.

En instalaciones sin lincado NCU activo, las variables de lincado pueden utilizarse de forma local para el control, junto con las variables globales del usuario (GUD), como variables globales del usuario adicionales.

Sintaxis

```
$A_DLB[<Índice>]
$A_DLW[<Índice>]
$A_DLD[<Índice>]
$A_DLR[<Índice>]
```

Descripción

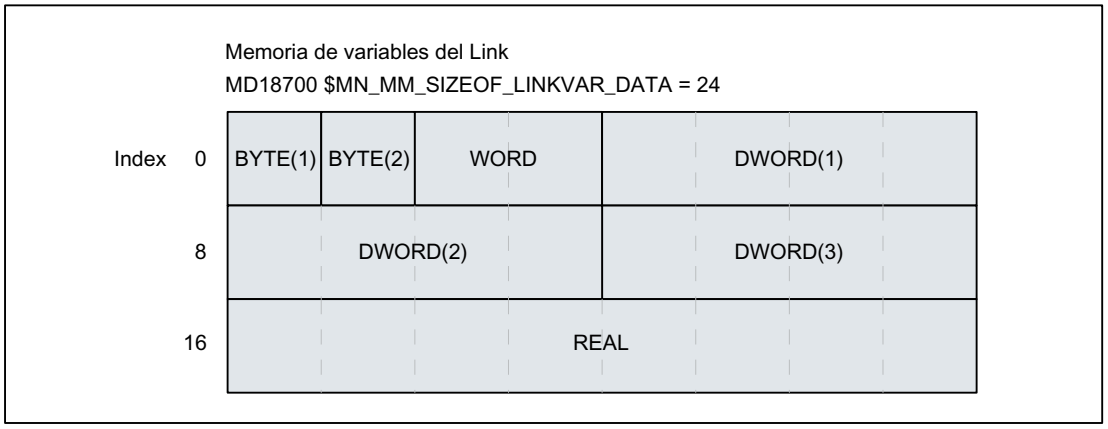
`$A_DLB`:	Variable de lincado para el formato de datos BYTE (1 byte)	
	Tipo de dato:	UINT
	Rango de valores:	0 ... 255
`$A_DLW`:	Variable de lincado para el formato de datos WORD (2 bytes)	
	Tipo de dato:	INT
	Rango de valores:	–32768...32767
`$A_DLD`:	Variable de lincado para el formato de datos DWORD (4 bytes)	
	Tipo de dato:	INT
	Rango de valores:	–2147483648...2147483647
`$A_DLR`:	Variable de lincado para el formato de datos REAL (8 bytes)	
	Tipo de dato:	REAL
	Rango de valores:	±(2,2*10^{-308}…1,8*10^{+308})
`<Índice>`:	Índice de direcciones en bytes, calculado desde el principio de la memoria de variables de lincado	
	Tipo de dato:	INT
	Rango de valores:	0 - MAX_INDEX **Nota** • MAX_INDEX se obtiene a partir del tamaño parametrizado de la memoria de variables de lincado: MAX_INDEX = (MD18700 $MN_MM_SIZEOF_LINKVAR_DATA) - 1 • Solamente pueden programarse índices para que los bytes direccionados en la memoria de variables de lincado se encuentren en un límite de formato de datos ⇒ Índice = n * bytes, con n = 0, 1, 2... – `$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, ...

Ejemplo

En la instalación de automatización hay 2 NCU (NCU1 y NCU2). A la NCU1 está conectado el eje de máquina AX2 que la NCU2 desplaza como eje lincado.

La NCU1 escribe cíclicamente el valor real de intensidad ($VA_CURR) del eje AX2 en la memoria de variables de lincado. La NCU2 lee cíclicamente el valor real de intensidad transmitido por comunicación de lincado e indica la alarma 61000 al sobrepasar el límite.

La estructura de datos en la memoria de variables de lincado se representa en la siguiente figura. El valor real de intensidad se transmite mediante el valor REAL.



NCU1

En una acción síncrona estática, la NCU1 escribe cíclicamente en el ciclo IPO el valor real de intensidad del eje AX2 en la memoria de variables de lincado mediante la variable de lincado $A_DLR[16].

Código de programa

```
N111 IDS=1 WHENEVER TRUE DO $A_DLR[16]=$VA_CURR[AX2]
```

NCU2

En una acción síncrona estática, la NCU2 lee cíclicamente en el ciclo IPO el valor real de intensidad del eje AX2 de la memoria de variables de lincado mediante la variable de lincado $A_DLR[16]. Si el valor real de intensidad es mayor que 23.0 A, se indica la alarma 61000.

Código de programa

```
N222 IDS=1 WHEN $A_DLR[16] > 23.0 DO SETAL(61000)
```

3.1.1.5 Definición de variables de usuario (DEF)

Con el comando `DEF` pueden definirse variables específicas de usuario o variables de usuario (User Data) y ocuparlas con valores.

De acuerdo con el ámbito de vigencia, es decir, el ámbito en el que la variable es visible, existen las siguientes categorías de variables de usuario:

- Variables locales de usuario (LUD)
  Las variables locales de usuario (LUD) son variables que están definidas en un programa CN que no es el programa principal en el momento de la ejecución. Se crean al llamar el programa CN y se borran con un reset al final del programa o bien con el siguiente arranque del control. Solo puede tenerse acceso a las LUD dentro del programa CN donde están definidas.
- Variables de usuario globales de programa (PUD)
  Las variables de usuario globales de programa (PUD) son variables que están definidas en un programa CN utilizado como programa principal. Se crean al llamar el programa CN y se borran con un reset al final del programa o bien con el siguiente arranque del control. Puede tenerse acceso a las PUD en el programa principal y en todos los subprogramas.

  Nota

  Disponibilidad de las variables de usuario globales de programa (PUD)

  Las variables de usuario globales de programa (PUD) definidas en el programa principal solo estarán disponibles en los subprogramas si está ajustado el siguiente dato de máquina:

  MD11120 $MN_LUD_EXTENDED_SCOPE = 1

  Con MD11120 = 0 las variables de usuario globales de programa definidas en el programa principal solo están disponibles en el programa principal.

- Variables globales de usuario (GUD)
  Las variables globales de usuario (GUD) son variables globales de CN o del canal que están definidas en un bloque de datos (SGUD, MGUD, UGUD, GUD4 ... GUD9) y que se conservan incluso después de un reset al final del programa o tras el siguiente arranque de control. Puede tenerse acceso a GUD en todos los programas CN.

Las variables de usuario deben haberse definido antes de su utilización (lectura/escritura). Se tienen que observar las siguientes reglas:

- Las GUD deben definirse en un fichero de definición, p. ej., _N_DEF_DIR/_N_UGUD_DEF.
- Las PUD y las LUD deben definirse en la parte de definición de un programa CN.
- Los datos deben definirse en una secuencia propia.
- Por cada definición de datos solamente se debe utilizar un tipo de datos.
- Por cada definición de datos pueden definirse varias variables del mismo tipo de datos.

Sintaxis

LUD y PUD

```
DEF <Tipo> <Unidad_fís> <Valores_límite> <Nombre>[<Valor_1>,
<Valor_2>, <Valor_3>]=<Valor_inic>
```

GUD

```
DEF <Ámbito> <Parada_DP> <Derechos_de_acceso> <Clase_de_datos>
<Tipo> <Unidad_fís> <Valores_límite> <Nombre>[<Valor_1>, <Valor_2>,
<Valor_3>]=<Valor_inic>
```

Descripción

DEF:	Comando para la definición de variables de usuario GUD, PUD, LUD	
<Ámbito>:	Ámbito de vigencia, sólo relevante para las GUD:	
	CN:	Variable de usuario global de CN
	CHAN:	Variable de usuario global del canal
<Parada_DP>:	Parada de decodificación previa, solo relevante para las GUD (opcional)	
	SYNR:	Parada de decodificación previa durante la lectura
	SYNW:	Parada de decodificación previa durante la escritura
	SYNRW:	Parada de decodificación previa durante la lectura/escritura
<Derechos de acceso>:	Nivel de protección para la lectura/escritura de GUD mediante el programa CN o BTSS (opcional)	
	APRP <Nivel de protección>:	Lectura: Programa CN
	APWP <Nivel de protección>:	Escritura: Programa CN
	APRB <Nivel de protección>:	Lectura: BTSS
	APWB <Nivel de protección>:	Escritura: BTSS
	<Nivel de protección>:	Rango de valores: 0 ... 7
	Ver "Atributo: derechos de acceso (APR, APW, APRP, APWP, APRB, APWB) (Página 423)"	
<Clase_de_datos>:	Asignación de clase de datos (¡solo SINUMERIK 828D!)	
	DCM:	Clase de datos M (= fabricante (manufacturer))
	DCI:	Clase de datos I (= individual)
	DCU:	Clase de datos U (= usuario)
<Tipo>:	Tipo de dato:	
	INT:	Valor entero con signo
	REAL:	Número real (LONG REAL según IEEE)
	BOOL:	Valor lógico TRUE (1)/FALSE (0)
	CHAR:	Caracteres ASCII
	STRING[<LongitudMáx>]:	Cadena de caracteres de una longitud definida
	AXIS:	Identificador de cabezal/eje
	FRAME:	Datos geométricos para una transformación de coordenadas estática
	Ver "Tipos de datos (Página 436)"	
<Unidad_fís>:	Unidad física (opcional)	
	PHU <Unidad>:	Unidad física
	Ver "Atributo: Unidad física (PHU) (Página 421)"	

<Valores límite>:	Valor límite inferior/superior (opcional)	
	LLI <Valor límite>:	Valor límite inferior (lower limit)
	ULI <Valor límite>:	Valor límite superior (upper limit)
	Ver "Atributo: Valores límite (LLI, ULI) (Página 419)"	
<Nombre>:	Nombre de la variable **Nota** • 31 caracteres como máximo • Los dos primeros caracteres deben de ser una letra y/o un guión bajo. • El carácter $ está reservado para las variables de sistema y no debe utilizarse.	
[<Valor_1>, <Valor_2>, <Valor_3>]:	Indicación de los tamaños de matriz para variables de matriz de 1 hasta 3 dimensiones como máximo (opcional) Para la inicialización de variables de matriz, ver "Definición e inicialización de variables de matriz (DEF, SET, REP) (Página 430)"	
<Valor_inic>:	Valor de inicialización (opcional) Ver "Atributo: Valor de inicialización (Página 416)" Para la inicialización de variables de matriz, ver "Definición e inicialización de variables de matriz (DEF, SET, REP) (Página 430)"	

Ejemplos

Ejemplo 1: definiciones de variables de usuario en el bloque de datos para el fabricante de la máquina

Código de programa	**Comentarios**
%_N_MGUD_DEF	; Bloque GUD: Fabricante de la máquina
$PATH=/_N_DEF_DIR	
DEF CHAN REAL PHU 24 LLI 0 ULI 10 INTENSIDAD_1, INTENSIDAD_2	
; Descripción	
; Definición de dos GUD: INTENSIDAD_1, INTENSIDAD_2	
; Ámbito de vigencia: Todo el canal	
; Tipo de dato: REAL	
; Parada DP: no programada => valor por defecto = sin parada DP	
; Unidad fís.: 24 = [A]	
; Valores límite: Low = 0.0, High = 10.0	
; Derechos de acceso: no programados => valor por defecto = 7 = posición del interruptor de llave 0	
; Valor de inicialización: no programado => valor por defecto = 0.0	
DEF NCK REAL PHU 13 LLI 10 APWP 3 APRP 3 APWB 0 APRB 2 TIEMPO_1=12, TIEMPO_2=45	
; Descripción	
; Definición de dos GUD: TIEMPO_1, TIEMPO_2	
; Ámbito de vigencia: En todo el CN	
; Tipo de dato: REAL	
; Parada DP: no programada => valor por defecto = sin parada DP	

Código de programa	Comentarios
; Unidad fís.: 13 = [s]	
; Valores límite: Low = 10.0, High = no programado => límite superior de la zona de definición	
; Derechos de acceso:	
; Programa CN: escritura/lectura = 3 = usuario final	
; BTSS: escritura = 0 = Siemens, lectura = 3 = usuario final	
; Valor de inicialización: TIEMPO_1 = 12.0, TIEMPO_2 = 45.0	
DEF NCK APWP 3 APRP 3 APWB 0 APRB 3 STRING[5] GUD5_NAME = "COUNTER"	
; Descripción	
; Definición de una GUD: GUD5_NAME	
; Ámbito de vigencia: En todo el CN	
; Tipo de dato: STRING, máx. 5 caracteres	
; Parada DP: no programada => valor por defecto = sin parada DP	
; Unidad fís.: no programada => valor por defecto = 0 = ninguna unidad fís.	
; Valores límite: no programados => límites de la zona de definición: Low = 0, High = 255	
; Derechos de acceso:	
; Programa CN: escritura/lectura = 3 = usuario final	
; BTSS: escritura = 0 = Siemens, lectura = 3 = usuario final	
; Valor de inicialización: "COUNTER"	
M30	

Ejemplo 2: variables de usuario locales y globales de programa (PUD/LUD)

Código de programa	Comentarios
PROC MAIN	; Programa principal
DEF INT VAR1	; Definición PUD
...	
SUB2	; Llamada a un subprograma
...	
M30	

Código de programa	Comentarios
PROC SUB2	; Subprograma SUB2
DEF INT VAR2	; DEFINICIÓN LUD
...	
IF (VAR1==1)	; Leer PUD
VAR1=VAR1+1	; Leer y escribir PUD
VAR2=1	; Escribir LUD
ENDIF	
SUB3	; Llamada a un subprograma
...	
M17	

Código de programa	Comentarios
PROC SUB3	; Subprograma SUB3
...	
IF (VAR1==1)	; Leer PUD
VAR1=VAR1+1	; Leer y escribir PUD
VAR2=1	; Error: LUD de SUB2 desconocido
ENDIF	
...	
M17	

Ejemplo 3: definición y utilización de variables de usuario del tipo de datos AXIS

Código de programa	Comentarios
DEF AXIS ABSCISA	; 1.er eje geométrico
DEF AXIS CABEZAL	; Cabezal
...	
IF ISAXIS(1) == FALSE GOTOF CONTINUAR	
ABSCISA = $P_AXN1	
CONTINUAR:	
...	
CABEZAL=(S1)	; 1.er cabezal
OVRA[CABEZAL]=80	; Corrección del cabezal = 80%
CABEZAL=(S3)	; 3.er cabezal

Condiciones

Variables globales de usuario (GUD)

En el contexto de la definición de variables globales de usuario (GUD) deben tenerse en cuenta los siguientes datos de máquina:

N.°	Identificador: $MN_	Descripción
11140	GUD_AREA_ SAVE_TAB	Almacenamiento adicional para bloques GUD
18118 [1)]	MM_NUM_GUD_MODULES	Número de ficheros GUD en el sistema de ficheros activo
18120 [1)]	MM_NUM_GUD_NAMES_NCK	Número de nombres GUD globales
18130 [1)]	MM_NUM_GUD_NAMES_CHAN	Número de nombres de GUD específicos de canal
18150 [1)]	MM_GUD_VALUES_MEM	Espacio en memoria para valores de GUD globales
18660 [1)]	MM_NUM_SYNACT_GUD_REAL	Número de GUD configurables del tipo de datos REAL
18661 [1)]	MM_NUM_SYNACT_GUD_INT	Número de GUD configurables del tipo de datos INT
18662 [1)]	MM_NUM_SYNACT_GUD_BOOL	Número de GUD configurables del tipo de datos BOOL

N.°	Identificador: $MN_	Descripción
18663 1)	MM_NUM_SYNACT_GUD_AXIS	Número de GUD configurables del tipo de datos AXIS
18664 1)	MM_NUM_SYNACT_GUD_CHAR	Número de GUD configurables del tipo de datos CHAR
18665 1)	MM_NUM_SYNACT_GUD_STRING	Número de GUD configurables del tipo de datos STRING

1) ¡MD de solo lectura en SINUMERIK 828D!

Uso en todos los canales de una variable de usuario global de CN del tipo de datos AXIS

Una variable de usuario global de CN del tipo de datos `AXIS` que se ha inicializado con un identificador de eje en la definición en el bloque de datos solamente se podrá utilizar en diferentes canales del CN si el eje tiene el mismo número de eje de canal en esos canales.

Si no es así, debe cargarse la variable al principio del programa CN o bien, como en el siguiente ejemplo, debe utilizarse la función AXNAME(...) (ver "Funciones de eje (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Página 904)").

Código de programa	Comentarios
`DEF NCK STRING[5] EJE="X"`	`; Definición en el bloque de datos`
`...`	
`N100 AX[AXNAME(EJE)]=111 G00`	`; Uso en el programa CN`

3.1.1.6 Redefinición de datos de sistema, datos de usuario e instrucciones de programación del CN (REDEF)

Con el comando `REDEF` pueden modificarse los atributos de datos de sistema, datos de usuario e instrucciones de programación del CN. El requisito básico para la redefinición es que ésta se ejecute después de la correspondiente definición.

En una redefinición no se pueden modificar varios atributos simultáneamente. Para cada atributo que se desee modificar debe programarse una instrucción `REDEF` propia.

Si se programan varias modificaciones de atributos a la vez, se activa siempre la última modificación.

Restablecimiento de los valores de atributo

Los atributos de derechos de acceso e instante de inicialización modificados con `REDEF` pueden restablecerse al valor por defecto reprogramando `REDEF` seguido del nombre de la variable o del comando de programación del CN:

- Derechos de acceso: Nivel de protección 7
- Instante de inicialización: sin inicialización, o se mantiene el valor actual

Atributos redefinibles

Ver "Vista general de atributos definibles y redefinibles (Página 428)".

Variables locales de usuario (PUD/LUD)

No pueden realizarse redefiniciones para variables locales de usuario (PUD/LUD).

Sintaxis

```
REDEF <Nombre> <Parada_DP>
REDEF <Nombre> <Unidad_fís.>
REDEF <Nombre> <Valores límite>
REDEF <Nombre> <Derechos de acceso>
REDEF <Nombre> <Instante_inic>
REDEF <Nombre> <Instante_inic> <Valor_inic>
REDEF <Nombre> <Clase de datos>
REDEF <Nombre>
```

Descripción

REDEF:	Comando para redefinir un determinado atributo o para restablecer los atributos "Derechos de acceso" y/o "Instante de inicialización" de variables del sistema, variables del usuario y comandos de programación de CN	
<Nombre>:	Nombre de una variable ya definida o de una instrucción de programación del CN	
<Parada_DP>:	Parada de decodificación previa	
	SYNR:	Parada de decodificación previa durante la lectura
	SYNW:	Parada de decodificación previa durante la escritura
	SYNRW:	Parada de decodificación previa durante la lectura/escritura
<Unidad_fís>:	Unidad física	
	PHU <Unidad>:	Unidad física
	Ver "Atributo: Unidad física (PHU) (Página 421)". **Nota:** no redefinible para: • Variables del sistema • Datos de usuario globales (GUD) de los tipos de datos: BOOL, AXIS, STRING, FRAME	
<Valores límite>:	Valor límite inferior/superior	
	LLI <Valor límite>:	Valor límite inferior (lower limit)
	ULI <Valor límite>:	Valor límite superior (upper limit)
	Ver "Atributo: Valores límite (LLI, ULI) (Página 419)". **Nota:** no redefinible para: • Variables del sistema • Datos de usuario globales (GUD) de los tipos de datos: BOOL, AXIS, STRING, FRAME	

<Derechos de acceso>:	Derechos de acceso para la lectura/escritura mediante el programa de pieza o BTSS	
	APX <Nivel de protección>:	Ejecución: Elemento de lenguaje CN
	APRP <Nivel de protección>:	Lectura: Programa de pieza
	APWP <Nivel de protección>:	Escritura: Programa de pieza
	APRB <Nivel de protección>:	Lectura: BTSS
	APWB <Nivel de protección>:	Escritura: BTSS
	<Nivel de protección>:	Rango de valores: 0 ... 7
	Ver "Atributo: derechos de acceso (APR, APW, APRP, APWP, APRB, APWB) (Página 423)".	
<Instante_inic>:	Instante en el que se reinicializa la variable	
	INIPO:	Power On
	INIRE:	Fin del programa principal, reset de CN o Power On
	INICF:	NEWCONF o fin del programa principal, reset de CN o Power On
	PRLOC:	Fin del programa principal, reset de CN tras una modificación local o Power On
	Ver "Atributo: Valor de inicialización (Página 416)".	
<Valor_inic>:	Valor de inicialización Al redefinir el valor de inicialización también se debe indicar siempre el instante de la inicialización (ver <Instante_inic>). Ver "Atributo: Valor de inicialización (Página 416)". Para la inicialización de variables de matriz, ver "Definición e inicialización de variables de matriz (DEF, SET, REP) (Página 430)". **Nota** No redefinible para variables del sistema, excepto datos del operador.	
<Clase_de_datos>:	Asignación de clase de datos (¡solo SINUMERIK 828D!)	
	DCM:	Clase de datos M (= fabricante (manufacturer))
	DCI:	Clase de datos I (= individual)
	DCU:	Clase de datos U (= usuario)

Ejemplo

Redefiniciones de la variable de sistema $TC_DPCx en el bloque de datos para fabricante de máquina

Código de programa

```
%_N_MGUD_DEF                                ; Bloque GUD: Fabricante de la máquina
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)
```

```
Código de programa
N800 REDEF $TC_DPC1
N900 REDEF $TC_DPC4
M30
```

Acerca de N100:	Derecho de acceso de escritura: BTSS = nivel de protección 2, programa de pieza = nivel de protección 3
Acerca de N200:	Unidad física [%]
Acerca de N300:	valor límite inferior = 0, valor límite superior = 200
Acerca de N400:	La variable de matriz se inicializa con los cuatro valores al realizar un PowerOn.
Acerca de N800/ N900:	Restablecimiento de los valores de atributo "Derechos de acceso" y/o "Instante de inicialización"

Nota

Uso de ficheros ACCESS

Si se utilizan ficheros ACCESS, la redefinición de los derechos de acceso debe desplazarse de_N_MGUD_DEF a _N_MACCESS_DEF.

Condiciones

Granularidad

Una redefinición hace referencia siempre a toda la variable identificada unívocamente mediante un nombre. No es posible, p. ej., asignar diferentes valores de atributo para elementos de matriz individuales en variables de matriz.

3.1.1.7 Atributo: Valor de inicialización

Definición (DEF) de variables de usuario

En la definición puede especificarse un valor de inicialización para las siguientes variables de usuario:

- Variables globales de usuario (GUD)
- Variables de usuario globales de programa (PUD)
- Variables locales de usuario (LUD)

Redefinición (REDEF) de variables de sistema y de usuario

En la redefinición puede especificarse un valor de inicialización para las siguientes variables:

- Datos de sistema
  - Datos del operador
- Datos de usuario
  - Parámetros R
  - Variables de acción síncrona ($AC_MARKER, $AC_PARAM, $AC_TIMER)
  - GUD de acción síncrona (SYG_xy[ ], con x=R, I, B, A, C, S e y=S, M, U, 4, ..., 9)
  - Parámetros EPS
  - OEM de datos de herramienta
  - OEM de datos de almacén
  - Variables globales de usuario (GUD)

Instante de la reinicialización

En la redefinición se puede indicar el instante en el que se debe reinicializar la variable, es decir, ajustar la variable otra vez al valor de inicialización.

- INIPO (PowerOn)
  La variable se reinicializa con Power On.
- INIRE (Reset)
  La variable se reinicializa con reset de CN, reset GMO, fin del programa de pieza (M02/M30) o Power On.
- INICF (NEWCONF)
  Con la función "Activar los datos de máquina", la variable se reinicializa mediante HMI, un comando de programa de pieza NEWCONF o un reset de CN, reset de BAG, fin del programa de pieza (M02/M30) o Power On.
- PRLOC (modificación local de programa)
  Entonces la variable sólo se reinicializará con un reset de CN, un reset GMO o con el fin del programa de pieza (M02/M30) si se ha modificado en el marco del programa de pieza actual.
  El atributo PRLOC sólo se puede utilizar en relación con los datos de operador programables (ver la tabla siguiente).

Tabla 3-1 Datos de operador programables

Número	Identificador	Comando 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

Número	Identificador	Comando G[1)]
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) Con este comando G se activa el dato de operador		

Condiciones

Valor de inicialización: variables globales de usuario (GUD)

- Para las variables globales de usuario (GUD) con el ámbito de vigencia CN, solamente se puede especificar INIPO (Power On) como instante de la inicialización.
- Para las variables globales de usuario (GUD) con el ámbito de vigencia CHAN, además de INIPO (Power On) se pueden especificar INIRE (Reset) o INICF (NEWCONF) como instante de la inicialización.
- En las variables globales de usuario (GUD) con el ámbito de vigencia CHAN y el instante de inicialización INIRE (Reset) o INICF (NEWCONF), en el caso de un reset de CN, reset de BAG y "Activar los datos de máquina", las variables se reinicializan solo en aquellos canales en los que se han disparado los sucesos mencionados.

Valor de inicialización: tipo de datos FRAME

Para variables del tipo de datos FRAME no se puede indicar ningún valor de inicialización. Las variables del tipo de datos FRAME se inicializan siempre implícitamente con el frame por defecto.

Valor de inicialización: tipo de datos CHAR

Para las variables del tipo de datos CHAR, en lugar del código ASCII (0...255) también se puede programar el carácter ASCII correspondiente entre comillas, p. ej. "A"

Valor de inicialización: tipo de datos STRING

En variables del tipo de datos STRING, la cadena de caracteres debe escribirse entre comillas, p. ej., ...= "MÁQUINA_1"

Valor de inicialización: tipo de datos AXIS

Para variables del tipo de datos `AXIS`, en el caso de la dirección extendida, el identificador de eje debe escribirse entre paréntesis, p. ej., ...=(X3)

Valor de inicialización: Variable del sistema

Para la variable del sistema no se pueden especificar valores de inicialización específicos del usuario mediante una redefinición. Los valores de inicialización de las variables del sistema están especificados en el sistema de forma fija. No obstante, mediante la redefinición se puede modificar el instante (`INIRE`, `INICF`) en el que se reinicializa la variable del sistema.

Valor de inicialización implícito: tipo de datos AXIS

Para variables del tipo de datos `AXIS` se utiliza el siguiente valor de inicialización implícito:

- Datos de sistema: "primer eje geométrico"
- GUD de acción síncrona (nombre: SYG_A*), PUD, LUD:
  Identificador de eje del dato de máquina: DM20082
  $MC_AXCONF_CHANAX_DEFAULT_NAME

Valor de inicialización implícito: datos de herramienta y de almacén

Para los datos de herramienta y de almacén, los valores de inicialización pueden especificarse mediante el siguiente dato de máquina: DM17520 $MN_TOOL_DEFAULT_DATA_MASK

Nota

Sincronización

La sincronización de sucesos que desencadenan la reinicialización de una variable global con la lectura de esta variable en otro lugar es responsabilidad exclusiva del usuario/fabricante de la máquina.

Consulte también

Variables (Página 399)

3.1.1.8 Atributo: Valores límite (LLI, ULI)

Los valores límite superior e inferior de la zona de definición sólo pueden especificarse para los siguientes tipos de datos:

- INT
- REAL
- CHAR

Definición (DEF) de variables de usuario: valores límite y valores de inicialización implícitos

Si al definir una variable de usuario de uno de los tipos de datos antes mencionados no se define ningún valor de inicialización explícito, la variable se ajusta al valor de inicialización implícito del tipo de datos:

- INT: 0
- REAL: 0.0
- CHAR: 0

Si el valor de inicialización implícito está fuera de la zona de definición establecida mediante los valores límite programados, la variable se inicializa con el valor límite que está más próximo al valor de inicialización implícito:

- Valor de inicialización implícito **<** valor límite inferior (LLI) ⇒
  valor de inicialización **=** valor límite inferior
- Valor de inicialización implícito **>** valor límite superior (ULI) ⇒
  valor de inicialización **=** valor límite superior

Ejemplos:

Código de programa	**Comentarios**

`DEF REAL GUD1`	`; Valor límite inferior = límite de la zona de definición`
	`; Valor límite superior = límite de la zona de definición`
	`; No se ha programado ningún valor de inicialización`
	`; => valor de inicialización implícito = 0.0`

`DEF REAL LLI 5.0 GUD2`	`; Valor límite inferior = 5.0`
	`; Valor límite superior = límite de la zona de definición`
	`; => valor de inicialización = 5.0`

`DEF REAL ULI -5 GUD3`	`; Valor límite inferior = límite de la zona de definición`
	`; Valor límite superior = -5.0`
	`; => valor de inicialización = -5.0`

Redefinición (REDEF) de variables de usuario: Valores límite y valores reales actuales

Si al redefinir los valores límite de una variable de usuario éstos se modifican de tal manera que el valor real actual se sitúa fuera de la zona nueva de definición, se produce una alarma y los valores límite no se adoptan.

Nota

Redefinición (REDEF) de variables de usuario

Al redefinir los valores límite de una variable de usuario, hay que presentar atención a que los siguientes valores se modifiquen con coherencia:

- Valores límite
- Valor real
- Valor de inicialización al efectuar la redefinición y la reinicialización automática debido a INIPO, INIRE o INICF

Consulte también

Variables (Página 399)

3.1.1.9 Atributo: Unidad física (PHU)

Una unidad física solamente puede especificarse para variables de los siguientes tipos de datos:

- INT
- REAL

Unidades físicas programables (PHU)

La unidad física se especifica como un número en coma fija: `PHU <Unidad>`

Se pueden programar las siguientes unidades físicas:

<Unidad>	Descripción	Unidad física
0	Ninguna unidad física	-
1	Posición lineal o angular 1)2)	[mm], [pulgadas], [grados]
2	Posición lineal 2)	[mm], [pulgadas]
3	Posición angular	[grados]
4	Velocidad lineal o angular 1)2)	[mm/min], [pulgadas/min], [rpm]
5	Velocidad lineal 2)	[mm/min]
6	Velocidad angular	[rpm]
7	Aceleración lineal o angular 1)2)	[m/s^2], [pulgadas/s^2], [r/s^2]
8	Aceleración lineal 2)	[m/s^2], [pulgadas/s^2]
9	Aceleración angular	[r/s^2]
10	Jerk lineal o angular 1)2)	[m/s^3], [pulgadas/s^3], [r/s^3]
11	Jerk lineal 2)	[m/s^3], [pulgadas/s^3]

<Unidad>	Descripción	Unidad física
12	Jerk angular	[r/s^3]
13	Tiempo	[s]
14	Ganancia del regulador de posición	[16.667/s]
15	Avance por vuelta 2)	[mm/vuelta], [pulgadas/vuelta]
16	Compensación de temperatura 1)2)	[mm], [pulgadas]
18	Fuerza	[N]
19	Masa	[kg]
20	Momento de inercia 3)	[kgm^2]
21	Porcentaje	[%]
22	Frecuencia	[Hz]
23	Tensión	[V]
24	Intensidad	[A]
25	Temperatura	[°C]
26	Ángulo	[grados]
27	Kv	[1000/min]
28	Posición lineal o angular 3)	[mm], [pulgadas], [grados]
29	Velocidad de corte 2)	[m/min], [pies/min]
30	Velocidad periférica 2)	[m/s], [pies/s]
31	Resistencia	[Ohm]
32	Inductancia	[mH]
33	Par 3)	[Nm]
34	Constante de par 3)	[Nm/A]
35	Ganancia del regulador de intensidad	[V/A]
36	Ganancia del regulador de velocidad 3)	[Nm/(rad*s)]
37	Velocidad de giro	[rpm]
42	Potencia	[kW]
43	Corriente, baja	[µA]
46	Par, bajo 3)	[µNm]
48	Por mil	-
49	-	[Hz/s]
65	Caudal	[l/min]
66	Presión	[bar]
67	Volumen 3)	[cm^3]
68	Ganancia del proceso 3)	[mm/(V*min)]
69	Ganancia del proceso del regulador de fuerza	[N/V]
155	Paso de rosca 3)	[mm/vuelta], [pulgadas/vuelta]
156	Cambio del paso de rosca 3)	[mm/vuelta / vuelta], [pulgadas/vuelta / vuelta]
1) La unidad física depende del tipo de eje: eje lineal o giratorio.		

<Unidad>	Descripción	Unidad física
2) Conmutación del sistema de medida G70/G71 (pulgadas/métrico) Tras una conmutación del sistema base (MD10240 $MN_SCALING_SYSTEM_IS_METRIC) con G70/G71, en accesos de escritura/lectura a variables de usuario y de sistema con indicación de longitud **no se realiza** una conversión de los valores (valor real, valor por defecto y valores límite) G700/G710 (pulgadas/métrico) Tras una conmutación del sistema base (MD10240 $MN_SCALING_SYSTEM_IS_METRIC) con G700/G710, en accesos de escritura/lectura a variables de usuario y de sistema con indicación de longitud **se realiza** una conversión de los valores (valor real, valor por defecto y valores límite)		
3) La variable **no** se convierte automáticamente al sistema de medida actual del CN (pulgadas/métrico). La conversión es responsabilidad exclusiva del usuario/fabricante de la máquina.		

Nota

Rebase de niveles por conversión de formato

El formato de almacenamiento interno para todas las variables de usuario (GUD/PUD/LUD) con unidades físicas con indicación de longitud es métrico. Una utilización excesiva de estas variables en el proceso principal del NCK, p. ej. en acciones síncronas, puede provocar un rebase del tiempo de cálculo del nivel de interpolador, alarma 4240, durante la conmutación del sistema de medida.

Nota

Compatibilidad de unidades

Durante la utilización de variables (asignación, comparación, cálculo, etc.) no se comprueba la compatibilidad de las unidades implicadas. Si en algún caso resulta necesario realizar una conversión, ésta será responsabilidad exclusiva del usuario/fabricante de la máquina.

Consulte también

Variables (Página 399)

3.1.1.10 Atributo: derechos de acceso (APR, APW, APRP, APWP, APRB, APWB)

Nombre

El nombre de los atributos de acceso AP... se compone de:

1. A: **A**ccess
2. P: **P**rotection
3. R / W: **R**ead / **W**rite
4. P / B: **P**rogram / **B**TSS

Derechos de acceso/niveles de protección

Con los derechos de acceso se corresponden los siguientes niveles de protección que deben indicarse durante la programación:

Derecho de acceso	Nivel de protección
Contraseña del sistema	0
Contraseña del fabricante de la máquina	1
Contraseña de servicio técnico	2
Contraseña del usuario final	3
Interruptor de llave posición 3	4
Interruptor de llave posición 2	5
Interruptor de llave posición 1	6
Interruptor de llave posición 0	7

Definición (DEF) de datos de usuario

Pueden definirse derechos de acceso (APR.../APW...) para los siguientes datos:

- Datos globales del usuario (GUD)

Redefinición (REDEF) de datos de sistema y de usuario

Pueden redefinirse derechos de acceso (APR.../APW...) para los siguientes datos:

- Datos de sistema
  - Datos de máquina

    Nota

    Redefinición de los derechos de lectura de datos de máquina

    El nivel de protección para la lectura de datos de máquina solo se puede ajustar con la palabra clave APR conjuntamente para el programa de pieza y BTSS.

    Las palabras clave APRP y APRB no se soportan durante la redefinición de los derechos de lectura y generan el aviso de alarma 12490 "El derecho de acceso APRP/APRB <nivel de protección> no se ha ajustado".

  - Datos del operador
  - Variable de sistema
  - Datos de proceso
  - Datos de almacén
  - Datos de herramienta
- Datos de usuario
  - Parámetros R
  - Variables de acción síncrona ($AC_MARKER, $AC_PARAM, $AC_TIMER)
  - GUD de acción síncrona (SYG_xy[ ], con x=R, I, B, A, C, S e y=S, M, U, 4, ..., 9)
  - Parámetros EPS
  - OEM de datos de herramienta
  - OEM de datos de almacén
  - Variables globales de usuario (GUD)

    Nota

    Durante la redefinición el derecho de acceso puede asignarse libremente a una variable situada entre el nivel de protección más bajo, 7, y el nivel de protección propio, p. ej. 1 (fabricante de la máquina).

Redefinición (REDEF) de comandos de programación CN

El derecho de acceso o de ejecución (APX) puede redefinirse para los siguientes comandos de programación CN:

- Comandos G/condiciones de desplazamiento (Página 174)
- Funciones predefinidas (Página 1340)
- Llamadas de subprograma predefinidas

- Instrucción DO para acciones síncronas
- Identificador de programa de ciclos
  El ciclo debe estar guardado en un directorio de ciclos y contener una instrucción PROC.

Derechos de acceso relativos a programas CN y ciclos (APRP, APWP)

Los diferentes derechos de acceso tienen los siguientes efectos para el acceso desde un programa CN o ciclo:

- APRP 0/APWP 0
  - Al ejecutar el programa CN debe estar definida la contraseña de sistema.
  - El ciclo debe estar guardado en el directorio _N_CST_DIR (Sistema).
  - Para el directorio _N_CST_DIR, el derecho de ejecución debe estar ajustado a Sistema en el MD11160 $MN_ACCESS_EXEC_CST.
- APRP 1/APWP 1 o bien APRP 2/APWP 2
  - Al ejecutar el programa CN debe estar definida la contraseña del fabricante de la máquina o de servicio técnico.
  - El ciclo debe estar guardado en el directorio _N_CMA_DIR (Fabricante de la máquina) o _N_CST_DIR.
  - Para los directorios _N_CMA_DIR o _N_CST_DIR, los derechos de ejecución deben estar ajustados como mínimo a Fabricante de la máquina en los datos de máquina MD11161 $MN_ACCESS_EXEC_CMA o MD11160 $MN_ACCESS_EXEC_CST.
- APRP 3/APWP 3
  - Al ejecutar el programa CN debe estar definida la contraseña del usuario final.
  - El ciclo debe estar guardado en el directorio _N_CUS_DIR (Usuario), _N_CMA_DIR o _N_CST_DIR.
  - Para los directorios _N_CUS_DIR, _N_CMA_DIR o _N_CST_DIR, los derechos de ejecución deben estar ajustados como mínimo a Usuario final en los datos de máquina MD11162 $MN_ACCESS_EXEC_CUS, MD11161 $MN_ACCESS_EXEC_CMA o MD11160 $MN_ACCESS_EXEC_CST.
- APRP 4...7/APWP 4...7
  - Al ejecutar el programa CN debe estar ajustada la posición 3 ... 0 del interruptor de llave.
  - El ciclo debe estar guardado en el directorio _N_CUS_DIR, _N_CMA_DIR o _N_CST_DIR.
  - Para los directorios _N_CUS_DIR, _N_CMA_DIR o _N_CST_DIR, los derechos de ejecución deben estar ajustados como mínimo a la correspondiente posición del interruptor de llave en los datos de máquina MD11162 $MN_ACCESS_EXEC_CUS, MD11161 $MN_ACCESS_EXEC_CMA o MD11160 $MN_ACCESS_EXEC_CST.

Derechos de acceso relativos a BTSS (APRB, APWB)

Los derechos de acceso (APRB, APWB) limitan el acceso a las variables de sistema y de usuario mediante BTSS para todos los componentes de sistema (HMI, PLC, equipos externos, servicios EPS, etc.) en igual medida.

Nota

Derechos de acceso locales de HMI

Si se realizan modificaciones de derechos de acceso de datos de sistema hay que vigilar que estas modificaciones sean coherentes con los derechos de acceso definidos mediante mecanismos de HMI.

Atributos de acceso APR/APW

Por motivos de compatibilidad, los atributos APR y APW se asocian implícitamente a los atributos APRP/APRB y APWP/APWB:

- APR x ⇒ APRP x APRB x
- APW y ⇒ APWP y APWB y

Derechos de acceso mediante ficheros ACCESS

Al utilizar ficheros ACCESS para la asignación de derechos de acceso, las redefiniciones de derechos de acceso para datos de sistema, datos de usuario y comandos de programación CN solo deben realizarse en ficheros ACCESS. Los datos globales del usuario (GUD) son una excepción. Para estos, la redefinición de los derechos de acceso deberá seguir realizándose en los correspondientes ficheros de definición *_DEF.

Para un derecho de acceso homogéneo, los datos de máquina para los derechos de ejecución y el derecho de acceso de los correspondientes directorios deben adaptarse de una forma coherente.

De ahí se obtiene la siguiente forma de proceder básica:

1. Creación de los ficheros de definición necesarios:
   - _N_DEF_DIR/_N_SACCESS_DEF
   - _N_DEF_DIR/_N_MACCESS_DEF
   - _N_DEF_DIR/_N_UACCESS_DEF
2. Parametrización del derecho de escritura para los ficheros de definición al valor necesario para la redefinición:
   - MD11170 $MN_ACCESS_WRITE_SACCESS = <Nivel de protección>
   - MD11171 $MN_ACCESS_WRITE_MACCESS = <Nivel de protección>
   - MD11172 $MN_ACCESS_WRITE_UACCESS = <Nivel de protección>

3. Para los accesos a los elementos protegidos desde los ciclos, deben adaptarse los derechos de ejecución y de escritura de los directorios de ciclos _N_CST_DIR, _N_CMA_DIR y _N_CST_DIR:
   Derechos de ejecución
   - MD11160 $MN_ACCESS_EXEC_CST = <Nivel de protección>
   - MD11161 $MN_ACCESS_EXEC_CMA = <Nivel de protección>
   - MD11162 $MN_ACCESS_EXEC_CUS = <Nivel de protección>

   Derechos de escritura
   - MD11165 $MN_ACCESS_WRITE_CST = <Nivel de protección>
   - MD11166 $MN_ACCESS_WRITE_CMA = <Nivel de protección>
   - MD11167 MN_ACCESS_WRITE_CUS = <Nivel de protección>

   El derecho de ejecución debe ajustarse como mínimo al mismo nivel de protección que el nivel máximo del elemento utilizado.
   El derecho de escritura debe ajustarse como mínimo al mismo nivel de protección que el derecho de ejecución.
4. Los derechos de escritura de los directorios de ciclos locales de HMI deben ajustarse al mismo nivel de protección que los de los directorios de ciclos locales de CN.

Llamadas de subprograma en ficheros ACCESS

Para la siguiente estructuración de la protección de acceso pueden llamarse también subprogramas (identificación SPF o MPF) en los ficheros ACCESS. Los subprogramas heredan entonces los derechos de ejecución del fichero ACCESS desde el que se llama.

Nota

En los ficheros ACCESS sólo pueden redefinirse los derechos de acceso. Todos los demás atributos se deben seguir redefiniendo o programando en los correspondientes ficheros de definición.

Consulte también

Variables (Página 399)

3.1.1.11 Vista general de atributos definibles y redefinibles

Las siguientes tablas muestran los atributos que se pueden definir (`DEF`) y/o redefinir (`REDEF`) y para qué tipos de datos.

Datos de sistema

Tipo de datos	Valor inic.	Valores límite	Unidad fís.	Derechos de acceso	Clase de datos (solo 828D)
Datos de máquina	---	---	---	`REDEF`	`REDEF`
Datos del operador	`REDEF`	---	---	`REDEF`	---

Tipo de datos	Valor inic.	Valores límite	Unidad fís.	Derechos de acceso	Clase de datos (solo 828D)
Datos de FRAME	---	---	---	REDEF	---
Datos de proceso	---	---	---	REDEF	---
Comp. de error de paso de husillo (EEC)	---	---	---	REDEF	---
Compensación de la flexión (CEC)	---	---	---	REDEF	---
Compensación del error de cuadrante (QEC)	---	---	---	REDEF	---
Datos de almacén	---	---	---	REDEF	---
Datos de herramienta	---	---	---	REDEF	---
Zonas protegidas	---	---	---	REDEF	---
Portaherramientas orientables	---	---	---	REDEF	---
Cadenas cinemáticas	---	---	---	REDEF	---
Zonas de protección 3D	---	---	---	REDEF	---
Limitación del campo de trabajo	---	---	---	REDEF	---

Datos de usuario

Tipo de datos	Valor inic.	Valores límite	Unidad fís.	Derechos de acceso	Clase de datos
Parámetros R	REDEF	REDEF	REDEF	REDEF	---
Variable de acción síncrona ($AC_...)	REDEF	REDEF	REDEF	REDEF	---
GUD de acción síncrona (SYG_...)	REDEF	REDEF	REDEF	REDEF	---
Parámetros EPS	REDEF	REDEF	REDEF	REDEF	---
OEM de datos de herramienta	REDEF	REDEF	REDEF	REDEF	---
OEM de datos de almacén	REDEF	REDEF	REDEF	REDEF	---
Variables globales del usuario (GUD)	DEF/REDEF	DEF	DEF	DEF/REDEF	DEF/REDEF
Variables locales del usuario (PUD/LUD)	DEF	DEF	DEF	---	---

Consulte también

Variables (Página 399)

3.1.1.12 Definición e inicialización de variables de matriz (DEF, SET, REP)

Una variable de usuario puede definirse como una matriz (array) de 1 a 3 dimensiones como máximo:

- 1 dimensión: `DEF <Tipo de datos> <Nombre de la variable>[<n>]`
- 2 dimensiones: `DEF <Tipo de datos> <Nombre de la variable>[<n>,<m>]`
- 3 dimensiones: `DEF <Tipo de datos> <Nombre de la variable>[<n>,<m>,<o>]`

Nota

Las variables de usuario del tipo de datos STRING pueden definirse como una matriz de 2 dimensiones como máximo.

Tipos de datos

Las variables de usuario pueden definirse como matrices para los siguientes tipos de datos: BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME

Asignación de valores a elementos de matriz

Las asignaciones de valores a elementos de matriz pueden realizarse en los momentos siguientes:

- en la definición de matrices (valores de inicialización);
- durante la ejecución del programa.

En este caso, los valores se pueden asignar:

- indicando explícitamente un elemento de matriz;
- indicando explícitamente un elemento de matriz como elemento inicial e indicando una lista de valores (`SET`);
- indicando explícitamente un elemento de matriz como elemento inicial e indicando un valor y su frecuencia de repetición (`REP`).

Nota

No se pueden asignar valores de inicialización a variables de usuario del tipo de datos FRAME.

Sintaxis (`DEF`)

```
DEF <Tipo de datos> <Nombre de la variable>[<n>,<m>,<o>]
DEF  STRING[<Longitud del string>] <Nombre de la variable>[<n>,<m>]
```

Sintaxis (DEF...=SET...)

Utilización de una lista de valores:

- En la definición:
  ```
  DEF <Tipo de datos> <Nombre de la
  variable>[<n>,<m>,<o>]=SET(<Valor1>,<Valor2>,...)
  ```
  con el mismo significado que:
  ```
  DEF <Tipo de datos> <Nombre de la
  variable>[<n>,<m>,<o>]=(<Valor1>,<Valor2>,...)
  ```

  Nota

  En la inicialización mediante una lista de valores la indicación de `SET` es opcional.

- En una asignación de valores:
  ```
  <Nombre de la variable>[<n>,<m>,<o>]=SET(<VALOR1>,<Valor2>,...)
  ```

Sintaxis (DEF...=REP...)

Utilización de un valor con repetición

- En la definición:
  ```
  DEF <Tipo de datos> <Nombre de la
  variable>[<n>,<m>,<o>]=REP(<Valor>)
  DEF <Tipo de datos> <Nombre de la
  variable>[<n>,<m>,<o>]=REP(<Valor>,<Cantidad de elementos de la
  matriz>)
  ```
- En una asignación de valores:
  ```
  <Nombre de la variable>[<n>,<m>,<o>]=REP(<Valor>)
  <Nombre de la variable>[<n>,<m>,<o>]=REP(<Valor>,<Cantidad de
  elementos de la matriz>)
  ```

Descripción

`DEF`:	Comando para la definición de variables	
`<Tipo de datos>`:	Tipo de datos de las variables	
	Rango de valores: • En variables del sistema: BOOL, CHAR, INT, REAL, STRING, AXIS • En variables GUD o LUD: BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME	
`<Longitud del string>`:	Cantidad máxima de caracteres en el tipo de datos STRING	
`<Nombre de la variable>`:	Nombre de variable	
`[<n>,<m>,<o>]`:	Tamaños o índices de matriz	
`<n>`:	Tamaño o índice de matriz para la 1.ª dimensión	
	Tipo:	INT (en variables del sistema también AXIS)
	Rango de valores:	Tamaño máx. de matriz: 65535 Índice de matriz: 0 ≤ n ≤ 65534

<m>:	Tamaño o índice de matriz para la 2.ª dimensión	
	Tipo:	INT (en variables del sistema también AXIS)
	Rango de valores:	Tamaño máx. de matriz: 65535 Índice de matriz: 0 ≤ m ≤ 65534
<o>:	Tamaño o índice de matriz para la 3.ª dimensión	
	Tipo:	INT (en variables del sistema también AXIS)
	Rango de valores:	Tamaño máx. de matriz: 65535 Índice de matriz: 0 ≤ o ≤ 65534
SET:	Asignación de valores mediante la lista de valores indicada	
(<Valor1>,<Valor2>,...):	Lista de valores	
REP:	Asignación de valores mediante el <Valor> indicado	
<Valor>:	Valor con el que deben describirse los elementos de matriz durante la inicialización con REP.	
<Cantidad de elementos de la matriz>:	Cantidad de elementos de la matriz que deben escribirse con el <Valor> indicado. Para el resto de elementos de la matriz, se aplica lo siguiente en función del momento: • Inicialización en la definición de matrices: → Los elementos de matriz restantes se escriben con cero • Asignación durante la ejecución del programa: → Los valores actuales de los elementos de matriz se mantienen invariables. Si el parámetro no está programado, todos los elementos de matriz se escriben con el <Valor>. Si el parámetro es igual a cero, en función del momento se aplica lo siguiente: • Inicialización en la definición de matrices: → Todos los elementos se preasignarán en cero • Asignación durante la ejecución del programa: → Los valores actuales de los elementos de matriz se mantienen invariables.	

Índice de matriz

El orden implícito de los elementos de matriz, p. ej., en una asignación de valores mediante SET o REP, se realiza con la iteración del índice de matriz de derecha a izquierda.

Ejemplo: Inicialización de una matriz de 3 dimensiones con 24 elementos:

```
DEF INT MATRIZ[2,3,4] = REP(1,24)
  MATRIZ[0,0,0] = 1   1. elemento de matriz
  MATRIZ[0,0,1] = 1   2. elemento de matriz
  MATRIZ[0,0,2] = 1   3. elemento de matriz
  MATRIZ[0,0,3] = 1   4. elemento de matriz
  ...
  MATRIZ[0,1,0] = 1   5. elemento de matriz
  MATRIZ[0,1,1] = 1   6. elemento de matriz
```

```
...
MATRIZ[0,2,3] = 1    12. elemento de matriz
MATRIZ[1,0,0] = 1    13. elemento de matriz
MATRIZ[1,0,1] = 1    14. elemento de matriz
...
MATRIZ[1,2,3] = 1    24. elemento de matriz
```

correspondientemente:

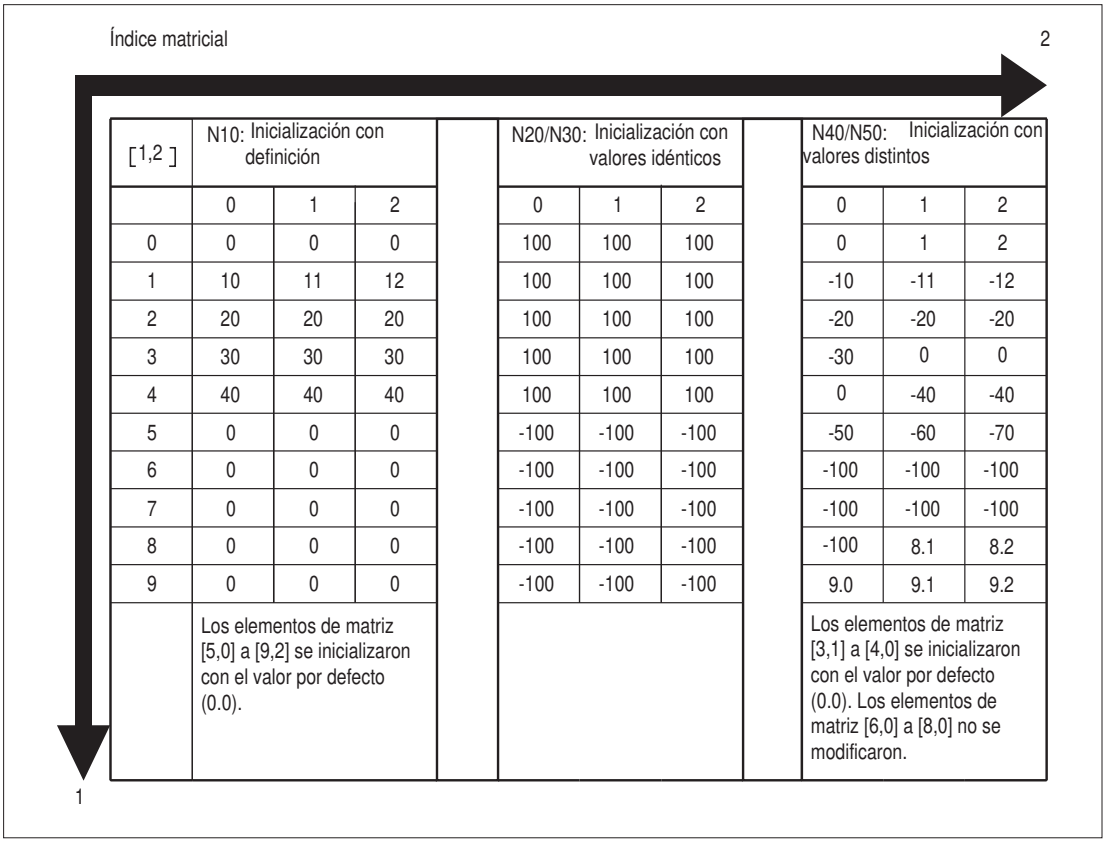
```
FOR n=0 TO 1
  FOR m=0 TO 2
    FOR o=0 TO 3
      MATRIZ[n,m,o] = 1
    ENDFOR
  ENDFOR
ENDFOR
```

Ejemplo: Inicialización de todos los elementos de matriz

Para la ocupación actual, ver figura.

Código de programa

```
N10 DEF REAL MATRIZ1[10,3]=SET(0,0,0,10,11,12,20,20,20,30,30,30,40,40,40,)
N20 MATRIZ1[0,0] = REP(100)
N30 MATRIZ1[5,0] = REP(-100)
N40 MATRIZ1[0,0]=SET(0,1,2,-10,-11,-12,-20,-20,-20,-30, , , ,-40,-40,-50,-60,-70)
N50 MATRIZ1[8,1]=SET(8.1,8.2,9.0,9.1,9.2)
```



[1,2]	N10: Inicialización con definición			N20/N30: Inicialización con valores idénticos			N40/N50: Inicialización con valores distintos		
	0	1	2	0	1	2	0	1	2
0	0	0	0	100	100	100	0	1	2
1	10	11	12	100	100	100	-10	-11	-12
2	20	20	20	100	100	100	-20	-20	-20
3	30	30	30	100	100	100	-30	0	0
4	40	40	40	100	100	100	0	-40	-40
5	0	0	0	-100	-100	-100	-50	-60	-70
6	0	0	0	-100	-100	-100	-100	-100	-100
7	0	0	0	-100	-100	-100	-100	-100	-100
8	0	0	0	-100	-100	-100	-100	8.1	8.2
9	0	0	0	-100	-100	-100	9.0	9.1	9.2
	Los elementos de matriz [5,0] a [9,2] se inicializaron con el valor por defecto (0.0).						Los elementos de matriz [3,1] a [4,0] se inicializaron con el valor por defecto (0.0). Los elementos de matriz [6,0] a [8,0] no se modificaron.		

Consulte también

Definición e inicialización de variables de matriz (DEF, SET, REP) (Página 430)

Variables (Página 399)

Información adicional (SET)

Inicialización en la definición

- Empezando por el 1.er elemento de matriz, se inicializan tantos elementos de matriz con los valores de la lista de valores como elementos haya programados en la lista de valores.
- Los elementos de matriz sin valores indicados explícitamente en la lista de valores (huecos en la lista de valores) se ocupan con 0.
- Las variables del tipo de datos AXIS no permiten huecos en la lista de valores.
- Si la lista de valores contiene más valores que elementos de matriz hay definidos, se muestra una alarma.

Asignación de valores en la ejecución del programa

Durante la asignación de valores en la ejecución del programa, se aplican las reglas descritas anteriormente en la definición. Adicionalmente existen las posibilidades siguientes:

- Como elementos de la lista de valores también están permitidas expresiones.
- La asignación de valores empieza con el índice de matriz programado. De esta manera, se pueden asignar valores a elementos de matriz determinados.

Ejemplo:

Código de programa	**Comentarios**
`DEF INT ARRAY[5,5]`	`; Definición de la matriz`
`ARRAY[0,0]=SET(1,2,3,4,5)`	`; Asignación de valores a los 5 primeros elementos de matriz [0,0] - [0,4]`
`ARRAY[0,0]=SET(1,2, , ,5)`	`; Asignación de valores con hueco a los 5 primeros elementos de matriz [0,0] - [0,4], elementos de matriz [0,2] y [0,3] = 0`
`ARRAY[2,3]=SET(VARIABLE,4*5.6)`	`; Asignación de valores con variable y expresión a partir del índice de matriz [2,3]: [2,3] = VARIABLE [2,4] = 4 * 5.6 = 22.4`

Información adicional (REP)

Inicialización en la definición

- Todos los elementos de matriz (o las cantidades de estos indicadas opcionalmente) se inicializan con el valor (constante) indicado.
- Las variables del tipo de datos FRAME no pueden inicializarse.

Ejemplo:

Código de programa	**Comentarios**
`DEF REAL varName[10]=REP(3.5,4)`	`; Definición de la matriz e inicialización de elementos de matriz de [0] a [3] con el valor 3,5`

Asignación de valores en la ejecución del programa

Durante la asignación de valores en la ejecución del programa, se aplican las reglas descritas anteriormente en la definición. Adicionalmente existen las posibilidades siguientes:

- Como elementos de la lista de valores también están permitidas expresiones.
- La asignación de valores empieza con el índice de matriz programado. De esta manera, se pueden asignar valores a elementos de matriz determinados.

Ejemplos:

Código de programa	**Comentarios**
`DEF REAL varName[10]`	`; Definición de la matriz`
`varName[5]=REP(4.5,3)`	`; Elementos de matriz de [5] a [7] = 4,5`
`R10=REP(2.4,3)`	`; Parámetros R de R10 a R12 = 2,4`
`DEF FRAME FRM[10]`	`; Definición de la matriz`

Código de programa	Comentarios
`FRM[5]=REP(CTRANS(X,5))`	`; Elementos de matriz de [5] a [9] = CTRANS(X,5)`

Consulte también

Definición e inicialización de variables de matriz (DEF, SET, REP) (Página 430)

3.1.1.13 Tipos de datos

En el CN están disponibles los siguientes tipos de datos:

Tipo de datos	Descripción	Rango
INT	Valor entero con signo	-2147483648 ... +2147483647
REAL	Número real (LONG REAL según IEEE)	±(~2,2*10^{-308} … ~1,8*10^{+308})
BOOL	Valores binarios TRUE (1) y FALSE (0)	1, 0
CHAR	Caracteres ASCII	Código ASCII 0 … 255
STRING	Cadena de caracteres de una longitud definida	Máximo 200 caracteres (sin caracteres especiales)
AXIS	Identificador de cabezal/eje	Identificador de eje de canal
FRAME	Datos geométricos para una transformación de coordenadas estática (desplazar, girar, escalar, invertir en simetría especular)	---

Conversiones de tipos de datos implícitas

Son posibles las siguientes conversiones de tipos de datos y se realizan implícitamente en asignaciones y transferencias de parámetros:

de ↓/ a →	REAL	INT	BOOL
REAL	x	o	&
INT	x	x	&
BOOL	x	x	x
x: posible sin limitaciones o: posible pérdida de datos por rebase del rango de valores ⇒ alarma; redondeo: valor decimal ≥ 0,5 ⇒ redondear al alza, valor decimal < 0,5 ⇒ redondear a la baja &: valor ≠ 0 ⇒ TRUE, valor == 0 ⇒ FALSE			

Consulte también

Variables (Página 399)

3.1.1.14 Mínimo, máximo y rango de variables (MINVAL, MAXVAL, BOUND)

Con los comandos MINVAL y MAXVAL pueden compararse entre sí los valores de dos variables. Como resultado se devuelve el valor menor (con MINVAL) o el valor mayor (con MAXVAL).

Con el comando BOUND se puede comprobar si el valor de una variable de prueba se encuentra dentro de un rango de valores definido.

Sintaxis

```
<Valor menor>=MINVAL(<Variable1>,<Variable2>)
<Valor mayor>=MAXVAL(<Variable1>,<Variable2>)
<Valor de retorno>=<BOUND>(<Mínimo>,<Máximo>,<Variable de prueba>)
```

Descripción

MINVAL:	Determina el valor **menor** de dos variables (<Variable1>, <Variable2>)
<Valor menor>:	Variable de resultado para el comando MINVAL Se ajusta al valor de variable menor.
MAXVAL:	Determina el valor **mayor** de dos variables (<Variable1>, <Variable2>)
<Valor mayor>:	Variable de resultado para el comando MAXVAL Se ajusta al valor de variable mayor.
BOUND:	Comprueba si una variable (<Variable de prueba>) está situada dentro de un rango de valores definido.
<Mínimo>:	Variable que define el valor mínimo del rango de valores
<Máximo>:	Variable que define el valor máximo del rango de valores
<Valor de retorno>:	Variable de resultado para el comando BOUND Si el valor de las variables de prueba está situado dentro del rango de valores definido, la variable de resultado se ajustará al valor de las variables de prueba. Si el valor de las variables de prueba es mayor que el valor máximo, la variable de resultado se ajustará al valor máximo de la zona de definición. Si el valor de las variables de prueba es menor que el valor mínimo, la variable de resultado se ajustará al valor mínimo de la zona de definición.

Nota

MINVAL, MAXVAL y BOUND se pueden programar también en acciones síncronas.

Nota

Comportamiento en caso de igualdad

En caso de igualdad, con MINVAL/MAXVAL se suministra este mismo valor. Con BOUND se devolverá el valor de las variables que se deben comprobar.

Ejemplo

Código de programa	Comentarios
DEF REAL rVar1=10.5, rVar2=33.7, rVar3, rVar4,	rVar5, rValMin, rValMax, rRetVar
rValMin=MINVAL(rVar1,rVar2)	; rValMin toma el valor 10.5.
rValMax=MAXVAL(rVar1,rVar2)	; rValMax toma el valor 33.7.
rVar3=19.7	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 está situado dentro de los límites, rRetVar toma el valor 19.7.
rVar3=1.8	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 está situado por debajo del límite mínimo, rRetVar toma el valor 10.5.
rVar3=45.2	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 está situado por encima del límite máximo, rRetVar toma el valor 33.7.

3.1.1.15 Comprobar la presencia de una variable (ISVAR)

Con la función predefinida ISVAR se puede comprobar si se conoce en el CN una variable de sistema o de usuario (p. ej., dato de máquina, dato de operador, variable del sistema, variable general como GUD).

Variable

La variable que se va a consultar se estructura de la siguiente manera:

Variable adimensional:	<Variable>
Variable unidimensional sin índice de matriz:	<Variable>[ ]
Variable unidimensional con índice de matriz n:	<Variable>[<n>]
Variable bidimensional sin índice de matriz:	<Variable>[,]
Variable bidimensional con índices de matriz n y m:	<Variable>[<n>,<m>]

Sintaxis

```
<Resultado>=ISVAR(<Variable>[<n>,<m>])
```

Descripción

<Resultado>:	Valor de retorno		
	Tipo de dato:	BOOL	
	Rango de valores:	1	Variable existente
		0	Variable desconocida
ISVAR:	Comprueba si se conoce en el CN la variable de sistema/usuario indicada.		
<Variable>:	Nombre de las variables de sistema/usuario		
	Tipo de dato:	STRING	

<n>:	Índice de matriz en la primera dimensión (**opcional**)	
	Tipo de dato:	INT
<m>:	Índice de matriz en la segunda dimensión (**opcional**)	
	Tipo de dato:	INT

Conforme al parámetro de transferencia se realizan las siguientes comprobaciones:

- Si el nombre es conocido
- Si la variable es una matriz
- Si se trata de una matriz unidimensional o bidimensional
- Si el índice de matriz en cuestión está dentro del rango permitido

Si todas estas comprobaciones son positivas, se devuelve TRUE (1).

Si una de las comprobaciones es negativa o se produce un error de sintaxis, se devuelve FALSE (0).

Ejemplos

Código de programa	**Comentarios**
DEF INT VAR1	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VAR1")	; IS_VAR es en este caso TRUE.

Código de programa	**Comentarios**
DEF REAL VARARRAY[10,10]	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VARARRAY[,]")	; IS_VAR es en este caso TRUE, es una matriz bidimensional.
N20 IS_VAR=ISVAR("VARARRAY")	; IS_VAR es TRUE, la variable existe.
N30 IS_VAR=ISVAR("VARARRAY[8,11]")	; IS_VAR es FALSE, el índice de matriz no se admite.
N40 IS_VAR=ISVAR("VARARRAY[8,8")	; IS_VAR es FALSE, falta "]" (error de sintaxis).
N50 IS_VAR=ISVAR("VARARRAY[,8]")	; IS_VAR es TRUE, el índice de matriz se admite.
N60 IS_VAR=ISVAR("VARARRAY[8,]")	; IS_VAR es TRUE, el índice de matriz se admite.

Código de programa	**Comentarios**
DEF BOOL IS_VAR=FALSE	
N100 IS_VAR=ISVAR("$MC_GCODE_RESET_VALUES[1]"	; El parámetro de transferencia es un dato de máquina, IS_VAR es TRUE.

Código de programa	**Comentarios**
DEF BOOL IS_VAR=FALSE	

Código de programa	Comentarios
N10 IS_VAR=ISVAR("$P_EP")	; IS_VAR es en este caso TRUE.
N20 IS_VAR=ISVAR("$P_EP[X]")	; IS_VAR es en este caso TRUE.

3.1.1.16 Leer valores de atributo/tipo de datos (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP)

Con las funciones predefinidas GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM y GETVARDFT pueden leerse los valores de atributo de variables de sistema/usuario, y con GETVARTYP puede leerse el tipo de datos de una variable de sistema/usuario.

Leer unidad física

Sintaxis:
`<Resultado>=GETVARPHU(<Nombre>)`

Significado:

`<Resultado>`:	Valor numérico de la unidad física		
	Tipo de dato:	INT	
	Rango de valores:	ver tabla en "Atributo: Unidad física (PHU) (Página 421)"	
		En caso de fallo:	
		- 2	El nombre de variable indicado no está asignado a ningún parámetro de sistema ni a ninguna variable de usuario
`GETVARPHU`:	Lectura de la unidad física de una variable de sistema/usuario		
`<Nombre>`:	Nombre de las variables de sistema/usuario		
	Tipo de dato:	STRING	

Ejemplo:

Supongamos que el CN contiene la siguiente variable GUD:

DEF CHAN REAL PHU 42 LLI 0 ULI 10000 electric

Código de programa	Comentarios
DEF INT result=0	
result=GETVARPHU("electric")	; Determina la unidad física de las variables GUD.
IF (result < 0) GOTOF error	

Como resultado se devuelve el valor 42. Esto corresponde a la unidad física [kW].

Nota

GETVARPHU permite comprobar, p. ej., si en una asignación de variables a = b, las dos variables tienen las unidades físicas esperadas.

Derecho de acceso de lectura

Sintaxis:
`<Resultado>=GETVARAP(<Nombre>,<Acceso>)`

Significado:

`<Resultado>`:	Nivel de protección para el <acceso> indicado		
	Tipo de dato:	INT	
	Rango de valores:	0 ... 7	Ver "Atributo: derechos de acceso (APR, APW, APRP, APWP, APRB, APWB) (Página 423)".
		En caso de fallo:	
		- 1	No se puede escribir (solo es relevante con los tipos de acceso "WP" y "WB")
		- 2	El nombre de variable indicado no está asignado a ningún parámetro de sistema ni a ninguna variable de usuario
		- 3	Valor erróneo para el parámetro <Acceso>
`GETVARAP`:	Lectura del derecho de acceso a una variable de sistema/usuario		
`<Nombre>`:	Nombre de las variables de sistema/usuario		
	Tipo de dato:	STRING	
`<Acceso>`:	Tipo de acceso		
	Tipo de dato:	STRING	
	Rango de valores:	`"RP"`	Lectura mediante programa de pieza
		`"WP"`	Escritura mediante programa de pieza
		`"RB"`	Lectura mediante BTSS
		`"WB"`	Escritura mediante BTSS

Ejemplo:

Código de programa	**Comentarios**
`DEF INT result=0`	
`result=GETVARAP("$TC_MAP8","WB")`	`; Determina la protección de acceso para el parámetro de sistema "Posición de almacén" en relación con la escritura mediante BTSS.`
`IF (result < 0) GOTOF error`	

Como resultado se devuelve el valor 7. Esto corresponde a la posición 0 del interruptor de llave (= no hay protección de acceso).

Nota

Con GETVARAP es posible, p. ej., crear un programa verificador que compruebe los derechos de acceso esperados por la aplicación.

Leer valores límite

Sintaxis:
`<Estado>=GETVARLIM(<Nombre>,<Valor límite>,<Resultado>)`

Significado:

`<Estado>`:	Estado de la función		
	Tipo de dato:	INT	
	Rango de valores:	1	o. k.
		-1	Ningún valor límite definido (para variables del tipo AXIS, STRING, FRAME)
		-2	El nombre de variable indicado no está asignado a ningún parámetro de sistema ni a ninguna variable de usuario
		-3	Valor erróneo para el parámetro <Valor límite>
`GETVARLIM`:	Lectura del valor límite inferior/superior de una variable de sistema/usuario		
`<Nombre>`:	Nombre de las variables de sistema/usuario		
	Tipo de dato:	STRING	
`<Valor límite>`:	Indica qué valor límite debe leerse		
	Tipo de dato:	CHAR	
	Rango de valores:	`"L"`	Límite inferior
		`"U"`	Límite superior
`<Resultado>`:	Devolución del valor límite		
	Tipo de dato:	VAR REAL	

Ejemplo:

Código de programa	**Comentarios**
`DEF INT state=0`	
`DEF REAL result=0`	
`state=GETVARLIM("$MA_MAX_AX_VELO","L",result)`	`; Determina el valor límite inferior para MD32000 $MA_MAX_AX_VELO.`
`IF (result < 0) GOTOF error`	

Lectura del valor dimensional

Sintaxis:
`<Resultado>=GETVARDIM(<Nombre>,<Índice>)`

Significado:

`<Resultado>`:	Tamaño de la dimensión indicada a través del parámetro <Índice>	
	Tipo de dato:	INT
`GETVARDIM`:	Lectura del tamaño de la 1.ª, 2.ª o 3.ª dimensión de la matriz de una variable de sistema/usuario	
`<Nombre>`:	Nombre de las variables de sistema/usuario	
	Tipo de dato:	STRING

<Índice>:	Índice de matriz		
	Tipo de dato:	INT	
	Rango de valores:	1 ... 3	
		1	Índice para la 1.ª dimensión de la matriz
		2	Índice para la 2.ª dimensión de la matriz
		3	Índice para la 3.ª dimensión de la matriz

Ejemplo:

Código de programa	**Comentarios**
N5 DEF REAL myReal[5,4]	
N10 R1=GETVARDIM("myReal",1)	; Determina el tamaño de la 1.ª dimensión de la matriz.
	; Resultado: R1 = 5
N15 R2=GETVARDIM("myReal",2)	; Determina el tamaño de la 2.ª dimensión de la matriz.
	; Resultado: R2 = 4

Leer valor estándar

Sintaxis:

```
<Estado>=GETVARDFT(<Nombre>,<Resultado>[,<Índice_1>,<Índice_2>,
<Índice_3>])
```

Significado:

<Estado>:	Estado de la función		
	Tipo de dato:	INT	
	Rango de valores:	1	o. k.
		-1	No hay ningún valor estándar disponible (p. ej., porque la definición del tipo de datos de las variables de resultado no es adecuada para el tipo de datos de las variables de sistema/usuario)
		-2	El nombre de variable indicado no está asignado a ningún parámetro de sistema ni a ninguna variable de usuario
		-3	Valor erróneo para el parámetro <Índice_1>, dimensión inferior a uno (⇒ sin matriz, pero con variable escalar)
		-4	Valor erróneo para el parámetro <Índice_2>
		-5	Valor erróneo para el parámetro <Índice_3>
GETVARDFT:	Lectura del valor estándar de una variable de sistema/usuario		
<Nombre>:	Nombre de las variables de sistema/usuario		
	Tipo de dato:	STRING	

<Resultado>:	Devolución del valor estándar	
	Tipo de dato:	VAR REAL (al leer el valor estándar de las variables de los tipos INT, REAL, BOOL, AXIS)
		VAR STRING (al leer el valor estándar de las variables de los tipos STRING y CHAR)
		VAR FRAME (al leer el valor estándar de las variables del tipo FRAME)
<Índice_1>:	Índice para la 1.ª dimensión de la matriz (opcional)	
	Tipo de dato:	INT
	No programado significa = 0	
<Índice_2>:	Índice para la 2.ª dimensión de la matriz (opcional)	
	Tipo de dato:	INT
	No programado significa = 0	
<Índice_3>:	Índice para la 3.ª dimensión de la matriz (opcional)	
	Tipo de dato:	INT
	No programado significa = 0	

Ejemplo:

Código de programa	**Comentarios**
DEF INT state=0	
DEF REAL resultR=0	; Variable para el registro de los valores estándar de los tipos INT, REAL, BOOL, AXIS.
DEF FRAME resultF=0	; Variable para el registro de los valores estándar del tipo FRAME
IF (GETVARTYP("$MA_MAX_AX_VELO") <> 4) GOTOF error	
state=GETVARDFT("$MA_MAX_AX_VELO",resultR, AXTOINT(X))	; Determina el valor estándar del eje "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)	

Leer tipo de datos

Sintaxis:

<Resultado>=GETVARTYP(<Nombre>)

Significado:

<Resultado>:	Tipo de datos de las variables de sistema/usuario indicadas		
	Tipo de dato:	INT	
	Rango de valores:	1	= BOOL
		2	= CHAR
		3	= INT
		4	= REAL
		5	= STRING
		6	= AXIS
		7	= FRAME
		En caso de fallo:	
		< 0	El nombre de variable indicado no está asignado a ningún parámetro de sistema ni a ninguna variable de usuario
GETVARTYP:	Lectura del tipo de dato de una variable de sistema/usuario		
<Nombre>:	Nombre de las variables de sistema/usuario		
	Tipo de dato:	STRING	

Ejemplo:

```
Código de programa                  Comentarios
DEF INT result=0
DEF STRING name="R"
result=GETVARTYP(name)              ; Determina el tipo de parámetro R.
IF (result < 0) GOTOF error
```

Como resultado se devuelve el valor 4. Esto corresponde al tipo de datos REAL.

3.1.1.17 Conversiones de tipos posibles

Los valores numéricos constantes, variables o expresiones asignadas a una variable deben de ser compatibles con el tipo de la variable. En el caso de que esto sea así, la variable toma automáticamente el valor asignado.

Posibles conversiones de tipos

a	REAL	INT	BOOL	CHAR	STRING	AXIS	FRAME
de							
REAL	sí	sí*	sí[1])	sí*	–	–	–
INT	sí	sí	sí[1])	sí[2])	–	–	–
BOOL	sí	sí	sí	sí	sí	–	–
CHAR	sí	sí	sí[1])	sí	sí	–	–
STRING	–	–	sí[4])	sí[3])	sí	–	–
AXIS	–	–	–	–	–	sí	–
FRAME	–	–	–	–	–	–	sí

Aclaraciones

* En la conversión de tipos de REAL a INT se redondea hacia arriba con una fracción de >=0.5; de lo contrario, se redondea hacia abajo (ver función ROUND)

1) El valor <> 0 equivale a TRUE, el valor == 0 equivale a FALSE

2) Si el valor se sitúa en el margen numérico admisible

3) Con sólo 1 carácter

4) Longitud del string 0 = >FALSE, de lo contrario TRUE

Nota

Si a la hora de convertir un valor, dicho valor es mayor que el rango permitido, automáticamente se genera un mensaje de error.

Si se mezclan varios tipos de variables en una expresión, automáticamente se realiza una conversión de tipo. Las conversiones de tipos también son posibles en las acciones síncronas, ver apartado "Acciones síncronas de desplazamiento, conversión implícita de tipos".

3.1.2 Programación indirecta

3.1.2.1 Programación indirecta de direcciones

En la programación indirecta de direcciones (Página 1280), la dirección ampliada (<Índice>) se sustituye por una variable del tipo adecuado.

Nota

La programación indirecta de direcciones no es posible para:

- N (Número de secuencia)
- L (subprograma)
- Direcciones ajustables
  (p. ej. X[1] en lugar de X1 no está permitido)

Sintaxis

```
<DIRECCIÓN>[<Índice>]
```

Descripción

<DIRECCIÓN>[...]:	Dirección fija con ampliación (índice)
<Índice>:	Variable, p. ej. para número de cabezal, eje...

Ejemplos

Ejemplo 1: programación indirecta de un número de cabezal

Programación directa:

Código de programa	Comentarios
S1=300	; Velocidad de 300 r/min para el cabezal con el número 1.

Programación indirecta:

Código de programa	Comentarios
DEF INT N_CABEZAL=1	; Definición de las variables del tipo INT y asignación de valores.
S[SPINU]=300	; Velocidad de 300 r/min para el cabezal cuyo número está guardado en la variable SPINU (en este ejemplo, el cabezal con el número 1).

Ejemplo 2: programación indirecta de un eje

Programación directa:

Código de programa	Comentarios
FA[U]=300	; Avance 300 para el eje "U".

Programación indirecta:

Código de programa	Comentarios
DEF AXIS AXVAR2=U	; Definición de una variable del tipo AXIS y asignación de valores.
FA[AXVAR2]=300	; Avance 300 para el eje cuyo nombre de dirección está guardado en la variable con el nombre AXVAR2.

Ejemplo 3: programación indirecta de un eje

Programación directa:

Código de programa	Comentarios
$AA_MM[X]	; Leer medida del palpador (MKS) del eje "X".

Programación indirecta:

Código de programa	Comentarios
DEF AXIS AXVAR3=X	; Definición de una variable del tipo AXIS y asignación de valores.
$AA_MM[AXVAR3]	; Leer medida del palpador (MKS) del eje cuyo nombre está guardado en la variable AXVAR3.

Ejemplo 4: programación indirecta de un eje

Programación directa:

Código de programa
X1=100 X2=200

Programación indirecta:

Código de programa	Comentarios
DEF AXIS AXVAR1 AXVAR2	; Definición de dos variables del tipo AXIS.
AXVAR1=(X1) AXVAR2=(X2)	; Asignación de los nombres de eje.
AX[AXVAR1]=100 AX[AXVAR2]=200	; Desplazamiento de los ejes cuyos nombres de dirección están guardados en las variables con los nombres AXVAR1 y AXVAR2.

Ejemplo 5: programación indirecta de un eje

Programación directa:

Código de programa
G2 X100 I20

Programación indirecta:

Código de programa	Comentarios
DEF AXIS AXVAR1=X	; Definición de una variable del tipo AXIS y asignación de valores.
G2 X100 IP[AXVAR1]=20	; Programación indirecta de la indicación de centro para el eje cuyo nombre de dirección está guardado en la variable con el nombre AXVAR1

Ejemplo 6: programación indirecta de elementos de matriz

Programación directa:

Código de programa	Comentarios
DEF INT MATRIZ1[4,5]	; Definición de la matriz 1.

Programación indirecta:

Código de programa	Comentarios
DEFINE DIM1 AS 4	; Al definir las dimensiones de la matriz debe indicarse el tamaño de la matriz como valor fijo.
DEFINE DIM2 AS 5	
DEF INT MA- TRIZ[DIM1,DIM2]	
MATRIZ[DIM1-1,DIM2-1]=5	

Ejemplo 7: llamada indirecta de subprograma

Código de programa	Comentarios
CALL "L" << R10	; Llamada del programa cuyo número se encuentra en R10 (concatenación).

3.1.2.2 Programación indirecta de comandos G

La programación indirecta de comandos G (Página 1293) permite una programación de ciclos efectiva.

Sintaxis

```
G[<Grupo>]=<Número>
```

Descripción

G[...]:	Comando G con ampliación (índice)	
<Grupo>:	Parámetro de índice: Grupo G	
	Tipo:	INT
<Número>:	Variable para el número de comando G	
	Tipo:	INT o REAL

Nota

Por lo general, solo pueden programarse indirectamente comandos G que no definen la sintaxis.

De los comandos G que definen la sintaxis, solo pueden utilizarse los del grupo G 1.
Los comandos G que definen la sintaxis de los grupos G 2, 3 y 4 no se pueden utilizar.

Nota

En la programación indirecta de comandos G no se admiten funciones aritméticas. Un cálculo necesario del número de comando G se tiene que realizar en una línea propia del programa de pieza antes de la programación indirecta de comandos G.

Ejemplos

Ejemplo 1: Decalaje de origen ajustable (grupo G 8)

Código de programa	Comentarios
N1010 DEF INT INT_VAR	
N1020 INT_VAR=2	
...	
N1090 G[8]=INT_VAR G1 X0 Y0	; G54
N1100 INT_VAR=INT_VAR+1	; Cálculo del comando G
N1110 G[8]=INT_VAR G1 X0 Y0	; G55

Ejemplo 2: Selección del plano (grupo G 6)

Código de programa	Comentarios
N2010 R10=$P_GG[6]	; Leer comando G activo del grupo G 6
...	
N2090 G[6]=R10	

3.1.2.3 Programación indirecta de atributos de posición (GP)

Los atributos de posición, como, p. ej., la programación incremental o absoluta de la posición del eje, pueden programarse indirectamente como variables junto con la palabra reservada GP.

Aplicación

La programación indirecta de atributos de posición se utiliza en **ciclos de sustitución**, ya que aquí se obtiene la siguiente ventaja con respecto a la programación de atributos de posición como palabra reservada (p. ej., IC, AC):

Gracias a la programación indirecta como variable, no se necesita **ninguna** instrucción CASE que se derive hacia todos los atributos de posición posibles.

Sintaxis

```
<COMANDO DE POSICIONAMIENTO>[<Eje/cabezal>]=
GP(<Posición>,<Atributo de posición>)
<Eje/cabezal>=GP(<Posición>,<Atributo de posición>)
```

Descripción

`<COMANDO DE POSICIONAMIENTO>[]`:	Los siguientes comandos de posicionamiento pueden programarse junto con la palabra reservada GP:
	`POS`, `POSA`, `SPOS`, `SPOSA`
	También son posibles:
	• todos los identificadores de eje/cabezal existentes en el canal:
	`<Eje/cabezal>`
	• identificador de eje/cabezal variable `AX`

`<Eje/cabezal>`:	Eje/cabezal que debe posicionarse
`GP()`:	Palabra reservada para el posicionamiento
`<Posición>`:	Parámetro 1 Posición de eje/cabezal como constante o variable
`<Atributo de posición>`:	Parámetro 2 Atributo de posición (p. ej. modo de aproximación a la posición) como variable (p. ej. $P_SUB_SPOSMODE) o como palabra reservada (IC, AC...)

Los valores suministrados por las variables tienen el siguiente significado:

Valor	Descripción	Admisible en:
0	Ninguna modificación del atributo de posición	
1	AC	POS, POSA,SPOS, SPOSA,AX, dirección de eje
2	IC	POS, POSA,SPOS, SPOSA,AX, dirección de eje
3	DC	POS, POSA,SPOS, SPOSA,AX, dirección de eje
4	ACP	POS, POSA,SPOS, SPOSA,AX, dirección de eje

Valor	Descripción	Admisible en:
5	ACN	POS, POSA,SPOS, SPOSA,AX, dirección de eje
6	OC	-
7	PC	-
8	DAC	POS, POSA,AX, dirección de eje
9	DIC	POS, POSA,AX, dirección de eje
10	RAC	POS, POSA,AX, dirección de eje
11	RIC	POS, POSA,AX, dirección de eje
12	CAC	POS, POSA
13	CIC	POS, POSA
14	CDC	POS, POSA
15	CACP	POS, POSA
16	CACN	POS, POSA

Ejemplo

En caso de acoplamiento síncrono activo de cabezales entre el cabezal maestro S1 y el cabezal esclavo S2, se llamará al siguiente ciclo de sustitución para el posicionamiento de los cabezales mediante el comando `SPOS` en el programa principal.

El posicionamiento se realiza mediante la instrucción en N2230:
```
SPOS[1]=GP($P_SUB_SPOSIT,$P_SUB_SPOSMODE)
SPOS[2]=GP($P_SUB_SPOSIT,$P_SUB_SPOSMODE)
```

La posición que se debe alcanzar se lee de la variable del sistema $P_SUB_SPOSIT, el modo de aproximación a la posición se lee de la variable del sistema $P_SUB_SPOSMODE.

Código de programa	Comentarios
N1000 PROC LANG_SUB DISPLOF SBLOF	
...	
N2100 IF($P_SUB_AXFCT==2)	
N2110	; Sustitución del comando SPOS/SPOSA/M19 durante un acoplamiento síncrono activo de cabezales
N2185 DELAYFSTON	; Inicio del área Stop-Delay
N2190 COUPOF(S2,S1)	; Desactivar acoplamiento síncrono de cabezales
N2200	; Posicionar cabezal maestro y cabezal esclavo
N2210 IF($P_SUB_SPOS==TRUE) OR ($P_SUB_SPOSA==TRUE)	
N2220	; Posicionar cabezal con SPOS:
N2230 SPOS[1]=GP($P_SUB_SPOSIT, $P_SUB_SPOSMODE)	
SPOS[2]=GP($P_SUB_SPOSIT, $P_SUB_SPOSMODE)	
N2250 ELSE	
N2260	; Posicionar cabezal con M19:

Código de programa	Comentarios
N2270 M1=19 M2=19	; Posicionar cabezal maestro y cabezal esclavo
N2280 ENDIF	
N2285 DELAYFSTOF	; Fin del área Stop-Delay
N2290 COUPON(S2,S1)	; Activar acoplamiento síncrono de cabezales
N2410 ELSE	
N2420	; Consulta de otras sustituciones
...	
N3300 ENDIF	
...	
N9999 RET	

Condiciones

La programación indirecta de atributos de posición no es posible en acciones síncronas.

3.1.2.4 Programación indirecta de líneas de programa de pieza (EXECSTRING)

El comando de programa de pieza `EXECSTRING` permite ejecutar como línea de programa de pieza una variable string previamente creada.

Sintaxis

`EXECSTRING` se programa en una línea propia del programa de pieza:
`EXECSTRING (<Variable string>)`

Descripción

`EXECSTRING`:	Comando para ejecutar una variable string como línea de programa de pieza
`<Variable string>`:	Variable del tipo STRING que contiene la línea del programa de pieza que realmente debe ejecutarse

Nota

Con `EXECSTRING` se pueden consignar, con la excepción de las estructuras de control (Página 488), todos los construcciones de programa de pieza que se pueden programar en el **cuerpo** de un programa de pieza. De esta manera se excluyen las instrucciones `PROC` y `DEF` y, en general, la utilización en ficheros INI y DEF.

Ejemplo

Código de programa	Comentarios
N100 DEF STRING[100] MY_BLOCK	; Definición de las variables string para el registro de las líneas de programa de pieza que deben ejecutarse.
N110 DEF STRING[10] MFCT1="M7"	
...	
N200 EXECSTRING(MFCT1 << "M4711")	; Ejecutar línea de programa de pieza "M7 M4711".
...	
N300 R10=1	
N310 MY_BLOCK="M3"	
N320 IF(R10)	
N330 MY_BLOCK = MY_BLOCK << MFCT1	
N340 ENDIF	
N350 EXECSTRING(MY_BLOCK)	; Ejecutar línea de programa de pieza "M3 M7".

3.1.3 Operaciones

3.1.3.1 Funciones de cálculo

Operador/función de cálculo	Descripción
+	Suma
–	Resta
*	Multiplicación
/ [1)]	División [1)]
DIV [1)]	División entera [1)]
MOD [1)]	División módulo (da como resultado el resto de una división entera) [1)]
:	Operador de concatenación para variables de tipo FRAME
SIN( )	Seno
COS( )	Coseno
TAN( )	Tangente
ASIN( )	Arcoseno
ACOS( )	Arcocoseno
ATAN2(,) [1)]	Arcotangente2 [1)]
SQRT( )	Raíz cuadrada
ABS( )	Valor absoluto
POT( )	2.ª potencia (cuadrado)

TRUNC( )	Parte entera Precisiones en comandos de comparación ajustables con TRUNC (ver "Corrección de precisión en caso de errores de comparación (TRUNC) (Página 458)")
ROUND()	Redondeo a un entero
LN( )	Logaritmo neperiano (logaritmo natural)
EXP( )	Exponencial
MINVAL( )	Valor menor de dos variables (ver "Mínimo, máximo y rango de variables (MINVAL, MAXVAL, BOUND) (Página 437)")
MAXVAL( )	Valor mayor de dos variables (ver "Mínimo, máximo y rango de variables (MINVAL, MAXVAL, BOUND) (Página 437)")
BOUND( )	Valor de variable que está en un rango de valores definido (ver "Mínimo, máximo y rango de variables (MINVAL, MAXVAL, BOUND) (Página 437)")
CTRANS()	Decalaje
CROT ()	Rotación
CSCALE ()	Factor de escala
CMIRROR ()	Simetría
1) Ver apartado "Ejemplos"	

Programación

En funciones de cálculo se utiliza la notación matemática estándar. Las prioridades para la ejecución de dichas operaciones se indican mediante paréntesis. Los ángulos utilizados para cálculos trigonométricos y sus funciones inversas se toman en grados (ángulo recto = 90°).

Ejemplos

División: /

(tipo REAL) = tipo INT o tipo REAL) / (tipo INT o tipo REAL);

Ejemplo: 3 / 4 = 0.75

División entera: DIV

(tipo INT) = (tipo INT o tipo REAL) / (tipo INT o tipo REAL);

Ejemplo: 7 DIV 4.1 = 1

División módulo (da como resultado el resto de una división entera): MOD

(tipo REAL) = (tipo INT o tipo REAL) MOD (tipo INT o tipo REAL);

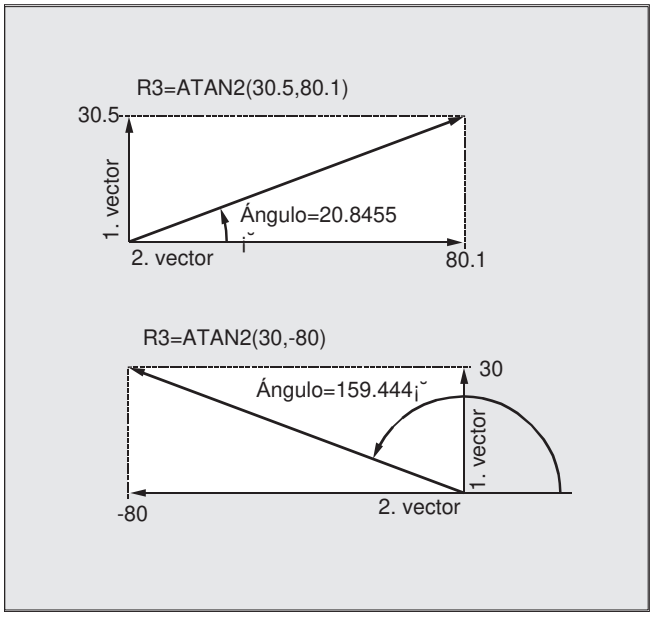
Ejemplo: 7 MOD 4.1 = 2.9

Arcotangente2: ATAN2

La función de cálculo ATAN2 calcula el ángulo respecto al origen del vector formado por dos componentes vectoriales orientadas a lo largo de los ejes de coordenadas.

El resultado se encuentra dentro del rango de los cuatro cuadrantes (-180° < 0 < +180°).

La referencia angular siempre se basa en el 2.º valor en la dirección positiva.



Ejemplos de programación

Código de programa	**Comentarios**
R1=R1+1	; Nuevo valor R1 = antiguo valor R1 + 1
R1=R2+R3 R4=R5-R6 R7=R8*R9	
R10=R11/R12 R13=SIN(25.3)	
R14=R1*R2+R3	; Multiplicaciones y divisiones se evalúan antes que sumas y restas.
R14=(R1+R2)*R3	; Primero se calculan las expresiones entre paréntesis.
R15=SQRT(POT(R1)+POT(R2))	; Primero se evalúan los paréntesis más internos: R15 = raíz cuadrada((R1^2 + R2^2))
RESFRAME=FRAME1:FRAME2 FRAME3=CTRANS(…):CROT(…)	; Operación lógica FRAME con operador de concatenación Asignación de valores a un componente de FRAME

3.1.3.2 Operaciones de comparación y operaciones lógicas

Las **operaciones de comparación** pueden utilizarse, p. ej., para formular una condición de salto. También se puede realizar la comparación de expresiones complejas.

Las operaciones de comparación se utilizan para variables del tipo `CHAR, INT, REAL` y `BOOL`. En variables del tipo `CHAR` se realiza una comparación de los valores codificados. Para variables del tipo `STRING`, `AXIS` y `FRAME` se pueden realizar comparaciones del tipo: **==** y **<>**, que también pueden utilizarse en acciones síncronas para operaciones con variables del tipo `STRING`.

El resultado de las operaciones de comparación siempre es del tipo `BOOL`.

Los**operadores lógicos** se utilizan para combinar valores lógicos.

Las operaciones lógicas sólo pueden aplicarse a variables del tipo `BOOL`. Mediante una conversión interna de tipos de variables también se pueden utilizar con los tipos de datos `CHAR, INT` y `REAL`.

En operaciones lógicas (booleanas) se debe tener en cuenta que para los tipos de datos `BOOL, CHAR, INT` y `REAL`:

- 0 significa: FALSE
- Distinto de 0 significa: TRUE

Operadores lógicos binarios

Con las variables del tipo `CHAR` e `INT` se pueden realizar también operaciones lógicas binarias. En caso necesario, se realiza una conversión automática de los tipos de variables.

Programación

Operador de comparación	Descripción
==	Igual que
<>	Distinto
>	Mayor que
<	Menor que
>=	Mayor o igual
<=	Menor o igual

Operador lógico	Descripción
AND	Y
OR	O
NOT	Negación
XOR	O exclusiva

Operador lógico bit a bit	Descripción
B_AND	Y binario
B_OR	O binario
B_NOT	Negación binaria
B_XOR	O exclusivo binario

Nota

Se pueden utilizar paréntesis con expresiones aritméticas para definir la secuencia de ejecución de todos los operadores; esto permite anular las reglas normales de prioridad.

Nota

Entre los operadores y operandos lógicos deben insertarse espacios.

Nota

El operador `B_NOT` se refiere sólo a un operando. Éste se encuentra detrás del operador.

Ejemplos

Ejemplo 1: Operadores de comparación

```
IF R10>=100 GOTOF DESTINO
```

o bien

```
R11=R10>=100
IF R11 GOTOF DESTINO
```

El resultado de la comparación `R10>=100` se almacena temporalmente en el parámetro `R11`.

Ejemplo 2: Operadores lógicos

```
IF (R10<50) AND ($AA_IM[X]>=17.5) GOTOF DESTINO
```

o bien

```
IF NOT R10 GOTOB INICIO
```

`NOT` hace referencia sólo a un operando.

Ejemplo 3: Operadores lógicos binarios

```
IF $MC_RESET_MODE_MASK B_AND 'B10000' GOTOF ACT_PLANE
```

3.1.3.3 Prioridad de las operaciones

Cada operador tiene una prioridad asignada. Al evaluar una expresión aritmética se realizan primeramente las operaciones con mayor orden de prioridad. En el caso de que se encuentren varias operaciones con el mismo orden de prioridad, éstas se ejecutan secuencialmente de izquierda a derecha.

Se pueden utilizar paréntesis con expresiones aritméticas para definir la secuencia de ejecución de todos los operadores; esto permite anular las reglas normales de prioridad.

Prioridades de los operadores

De mayor a menor

1.	NOT, B_NOT	Negación, negación binaria
2.	*, /, DIV, MOD	Multiplicación, división
3.	+, –	Suma, resta
4.	B_AND	Y binario
5.	B_XOR	O exclusivo binario
6.	B_OR	O binario
7.	AND	Y

8.	XOR	O exclusivo
9.	OR	O
10.	<<	Concatenación de caracteres, resultado tipo STRING
11.	==, <>, >, <, >=, <=	Operadores de comparación

Nota

El operador de concatenación ":" para frames no debe aparecer con otros operadores en una misma expresión. Por lo tanto no es necesario definir un nivel de prioridad para dicho operador.

Ejemplo de instrucción If:

```
If (otto==10) and (anna==20) gotof end
```

3.1.3.4 Corrección de precisión en caso de errores de comparación (TRUNC)

El comando TRUNC recorta el operando multiplicado con un factor de precisión.

Precisión ajustable en comandos de comparación

Los datos de telegrama de pieza del tipo REAL se representan a nivel interno en el formato IEEE con 64 bits. Debido a esta forma de representación, los números decimales se reflejan de forma imprecisa, lo cual puede producir resultados inesperados en una comparación con valores calculatorios ideales.

Igualdad relativa

Para evitar que las imprecisiones causadas por la forma de representación alteren el flujo del programa, la comprobación en los comandos de comparación no se efectúa con vistas a la igualdad absoluta, sino a la igualdad relativa.

Sintaxis

Corrección de precisión en caso de errores de comparación

```
TRUNC (R1*1000)
```

Descripción

TRUNC:	Supresión de decimales

Igualdad relativa tenida en cuenta de 10^{-12} con

- Igualdad: (==)
- Desigualdad: (<>)
- Mayor-igual: (>=)
- Menor-igual: (<=)
- Mayor/menor: (><) con igualdad absoluta

- Mayor: (>)
- Menor: (<)

Compatibilidad

Por motivos de compatibilidad, puede desactivarse la comprobación de la igualdad relativa con (>) y (<) definiendo el dato de máquina DM10280 $MN_ PROG_FUNCTION_MASK bit0 = 1.

Nota

Por las razones citadas, las comparaciones con datos del tipo REAL muestran generalmente una cierta imprecisión. En caso de desviaciones inaceptables se tiene que recurrir al cálculo de ENTEROS, multiplicando el operando con un factor de precisión y recortándolo después con TRUNC.

Acciones síncronas

El comportamiento descrito de los comandos de comparación también se aplica en acciones síncronas.

Ejemplos

Ejemplo 1: Consideraciones de precisión

Código de programa	Comentarios
N40 R1=61.01 R2=61.02 R3=0.01	; Asignación de los valores iniciales
N41 IF ABS(R2-R1) > R3 GOTOF ERROR	; El salto se ejecutaría hasta ahora
N42 M30	; Fin del programa
N43 ERROR: SETAL(66000)	
R1=61.01 R2=61.02 R3=0.01	; Asignación de los valores iniciales
R11=TRUNC(R1*1000) R12=TRUNC(R2*1000) R13=TRUNC(R3*1000)	; Corrección de precisión
IF ABS(R12-R11) > R13 GOTOF ERROR	; El salto ya no se ejecuta
M30	; Fin del programa
ERRORES: SETAL(66000)	

Ejemplo 2: formar y evaluar el cociente de ambos operandos

Código de programa	Comentarios
R1=61.01 R2=61.02 R3=0.01	; Asignación de los valores iniciales
IF ABS((R2-R1)/R3)-1) > 10EX-5 GOTOF ERROR	; El salto no se ejecuta
M30	; Fin del programa
ERRORES: SETAL(66000)	

3.1.3.5 Redondeo (ROUNDUP)

La función "ROUNDUP" permite redondear valores de entrada del tipo REAL (valor fraccionario con punto decimal) al siguiente valor entero superior.

Sintaxis

```
ROUNDUP(<Valor>)
```

Descripción

ROUNDUP:	Comando para redondear un valor de entrada
<Valor>:	Valor de entrada del tipo REAL

Nota

Los valores de entrada del tipo INTEGER (un número entero) se devuelven sin modificar.

Ejemplos

Ejemplo 1: distintos valores de entrada y sus resultados de redondeo

Ejemplo	Resultado de redondeo
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

Ejemplo 2: ROUNDUP en el programa CN

Código de programa

```
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 Operaciones con cadenas de caracteres

Operaciones de cadena

Además de las clásicas operaciones "Asignación" y "Comparación", son posibles las siguientes operaciones de cadena:

- Conversión de tipos a STRING (AXSTRING) (Página 461)
- Conversión de tipos de STRING (NUMBER, ISNUMBER, AXNAME) (Página 462)
- Concatenación de cadenas (<<) (Página 463)
- Conversión minúsculas/mayúsculas (TOLOWER, TOUPPER) (Página 464)
- Definir la longitud de una cadena (STRLEN) (Página 465)
- Búsqueda de caracteres/cadena en una cadena (INDEX, RINDEX, MINDEX, MATCH) (Página 466)
- Selección de una cadena parcial (SUBSTR) (Página 467)
- Lectura y escritura de caracteres individuales (Página 468)
- Formatear cadena (SPRINT) (Página 469)

Significado especial del carácter 0

El carácter 0 se interpreta internamente como delimitador final de una cadena. Si se sustituye un carácter por 0, se acorta la cadena.

Ejemplo:

Código de programa	**Comentarios**
`DEF STRING[20] STRG="Eje . parado"`	
`STRG[6]="X"`	
`MSG(STRG)`	`; Emite el aviso "Eje X parado".`
`STRG[6]=0`	
`MSG(STRG)`	`; Emite el aviso "Eje".`

3.1.4.1 Conversión de tipos a STRING (AXSTRING)

La función "Conversión de tipo a STRING" permite utilizar variables de distintos tipos como componente de un aviso (MSG).

Tienen lugar al utilizar el operador << implícitamente para los tipos de datos INT, REAL, CHAR y BOOL (ver "Concatenación de cadenas (<<) (Página 463)").

Un valor INT se convertirá en una forma legible normal. Para los valores REAL se introducirán hasta 10 caracteres asociados a las correspondientes posiciones decimales.

El comando `AXSTRING` permite convertir variables del tipo AXIS a STRING.

Sintaxis

```
<STRING_ERG> = << <Cualquier tipo>
<STRING_ERG> = AXSTRING(<Identificador de eje>)
```

Descripción

<STRING_ERG>:	Variable para el resultado de la conversión de tipo	
	Tipo:	STRING
<Cualquier tipo>:	Tipos de variables INT, REAL, CHAR, STRING y BOOL	
AXSTRING:	El comando AXSTRING proporciona el identificador de eje indicado como cadena.	
<Identificador de eje>:	Variable para el identificador de eje	
	Tipo:	AXIS

Nota

Las variables FRAME no se pueden convertir.

3.1.4.2 Conversión de tipos de STRING (NUMBER, ISNUMBER, AXNAME)

El comando NUMBER convierte STRING a REAL. Es posible comprobar la convertibilidad con el comando ISNUMBER.

El comando AXNAME convierte una cadena en el tipo de datos AXIS.

Sintaxis

```
<REAL_ERG>=NUMBER("<String>")
<BOOL_ERG>=ISNUMBER("<String>")
<AXIS_ERG>=AXNAME("<String>")
```

Descripción

NUMBER:	El comando NUMBER devuelve la cifra representada por el <String> como valor REAL.	
<String>:	Variable del tipo STRING que debe convertirse	
<REAL_ERG>:	Variable para el resultado de la conversión de tipo con NUMBER	
	Tipo:	REAL
ISNUMBER:	El comando ISNUMBER permite verificar si el <String> puede convertirse en un número válido.	

<BOOL_ERG>:	Variable para el resultado de la consulta con ISNUMBER		
	Tipo:	BOOL	
	Valor:	TRUE	ISNUMBER proporciona el valor TRUE cuando el <String> representa un número REAL de acuerdo a las reglas del idioma.
		FALSE	Si ISNUMBER proporciona el valor FALSE, con la llamada de NUMBER con el mismo <String> se activa una alarma.
AXNAME:	El comando AXNAME convierte el <String> indicado en un identificador de eje. **Nota:** si el <String> no puede asignarse a ningún identificador de eje configurado, se desencadenará una alarma.		
<AXIS_ERG>:	Variable para el resultado de la conversión de tipo con AXNAME		
	Tipo:	AXIS	

Ejemplo

```
Código de programa                          Comentarios
DEF BOOL BOOL_ERG
DEF REAL REAL_ERG
DEF AXIS AXIS_ERG
BOOL_ERG=ISNUMBER("1234.9876Ex-7")          ; BOOL_ERG == TRUE
BOOL_ERG=ISNUMBER("1234XYZ")                ; BOOL_ERG == FALSE
REAL_ERG=NUMBER("1234.9876Ex-7")            ; REAL_ERG == 1234.9876Ex-7
AXIS_ERG=AXNAME("X")                        ; AXIS_ERG == X
```

3.1.4.3 Concatenación de cadenas (<<)

La función "Concatenación de cadenas" permite formar una cadena a partir de distintos componentes.

La concatenación se efectúa mediante el operador "**<<**". Este operador genera para todas las combinaciones de los tipos de variables básicas CHAR, BOOL, INT, REAL y STRING una variable del tipo de destino STRING. En el caso de que sea necesario realizar una conversión de tipo, ello se realizará siguiendo las reglas prefijadas.

Sintaxis

```
<Cualquier tipo> << <Cualquier tipo>
```

Descripción

<Cualquier tipo>:	Variable del tipo CHAR, BOOL, INT, REAL o STRING
<< :	Operador para la concatenación de variables (<Cualquier tipo>) para forma una cadena de caracteres compuesta (tipo STRING). Este operador también está disponible solo como variante "unaria". De este modo es posible ejecutar una conversión de tipo explícita a STRING (no para FRAME y AXIS): << <Cualquier tipo>

Por ejemplo, es posible componer de este modo un aviso o un comando a partir de listas de textos y añadir parámetros (p. ej., un nombre de bloque):
`MSG(STRG_TAB[LOAD_IDX]<<BAUSTEIN_NAME)`

Nota

Los resultados intermedios de la concatenación de cadenas no deben superar la longitud máxima de cadena.

Nota

Los tipos FRAME y AXIS no pueden utilizarse junto con el operador "<<".

Ejemplos

Ejemplo 1: concatenación de cadenas (strings)

```
DEF INT IDX=2
DEF REAL VALUE=9.654
DEF STRING[20] STRG="INDEX:2"
IF STRG=="Índice:"<<IDX GOTOF NO_MSG
MSG("Índice:"<<IDX<<"/valor:"<<VALUE)          ; Visualización:
                                               "Índice:2/valor:9.654"
NO_MSG:
```

Ejemplo 2: conversión de tipos explícita con <<

```
DEF REAL VALUE=3.5
<<VALUE                  ; La variable indicada del tipo REAL se convierte al tipo
                         STRING.
```

3.1.4.4 Conversión minúsculas/mayúsculas (TOLOWER, TOUPPER)

La función "Conversión minúsculas/mayúsculas" permite convertir todas las letras de una cadena de caracteres para que se representen uniformemente.

Sintaxis

```
<STRING_ERG>=TOUPPER("<String>")
<STRING_ERG>=TOLOWER("<String>")
```

Descripción

TOUPPER:	El comando TOUPPER permite convertir todas las letras de una cadena de caracteres en letras **mayúsculas**.	
TOLOWER:	El comando TOLOWER permite convertir todas las letras de una cadena de caracteres en letras **minúsculas**.	
<String>:	Cadena de caracteres que debe convertirse	
	Tipo:	STRING
<STRING_ERG>:	Variable para el resultado de la conversión	
	Tipo:	STRING

Ejemplo

Como también existe la posibilidad de que el usuario introduzca datos a través de la interfaz de usuario, se puede obtener una representación uniforme utilizando letras minúsculas o mayúsculas:

Código de programa

```
DEF STRING [29] STRG
...
IF "LEARN.CNC"==TOUPPER(STRG) GOTOF LOAD_LEARN
```

3.1.4.5 Definir la longitud de una cadena (STRLEN)

El comando STRLEN permite definir la longitud de una cadena de caracteres.

Sintaxis

```
<INT_ERG>=STRLEN("<STRING>")
```

Descripción

STRLEN:	El comando STRLEN define la longitud de la cadena de caracteres indicada. Se devuelve el número de caracteres que, contados desde el principio de la cadena de caracteres, no son caracteres 0.	
<String>:	Cadena de caracteres cuya longitud debe determinarse	
	Tipo:	STRING
<INT_ERG>:	Variable para el resultado de la determinación	
	Tipo:	INT

Ejemplo

La función en combinación con el acceso a un carácter individual permite determinar el final de una cadena de caracteres:

```
Código de programa
IF (STRLEN(BAUSTEIN_NAME)>10) GOTOF ERROR
```

3.1.4.6 Búsqueda de caracteres/cadena en una cadena (INDEX, RINDEX, MINDEX, MATCH)

Esta funcionalidad permite buscar caracteres individuales, así como cadenas de caracteres dentro de un string. El resultado de la función indica en qué posición del string en el que se realiza la búsqueda se encuentra el carácter/la cadena de caracteres que se desea localizar.

Sintaxis

`INT_ERG=INDEX(STRING,CHAR)` ; **tipo de resultado: INT**

`INT_ERG=RINDEX(STRING,CHAR)` ; **tipo de resultado: INT**

`INT_ERG=MINDEX(STRING,STRING)` ; **tipo de resultado: INT**

`INT_ERG=MATCH(STRING,STRING)` ; **tipo de resultado: INT**

Semántica

Funciones de búsqueda: Devuelven la posición de la cadena (primer parámetro) en la que la búsqueda ha tenido resultados. Si no se pueden encontrar el carácter o la cadena, el valor devuelto será -1. El primer carácter tiene la posición 0.

Descripción

`INDEX`:	Busca el carácter indicado como segundo parámetro (desde el principio) en el primer parámetro.
`RINDEX`:	Busca el carácter indicado como segundo parámetro (desde el final) en el primer parámetro.
`MINDEX`:	Equivale a la función INDEX, excepto en que se emite una lista de caracteres (como cadena) de la que se devuelve el índice del primer carácter encontrado.
`MATCH`:	Busca una cadena dentro de una cadena.

De este modo es posible dividir las cadenas según criterios determinados, como en posiciones con espacios o por el carácter separador de ruta ("/").

Ejemplo

División de una entrada en un nombre de ruta y de bloque

Código de programa	Comentarios
`DEF INT PFADIDX, PROGIDX`	
`DEF STRING[26] ENTRADA`	
`DEF INT LISTIDX`	

Código de programa	Comentarios
ENTRADA = "/_N_MPF_DIR/_N_EXECU- TE_MPF"	
LISTIDX = MINDEX (ENTRADA, "M,N,O,P") + 1	; En LISTIDX se devuelve 3 como valor, puesto que "N" es el primer carácter del parámetro ENTRADA de la lista de selección (desde el principio).
PFADIDX = INDEX (ENTRADA, "/") +1	; Por tanto: PFADIDX = 1
PROGIDX = RINDEX (ENTRADA, "/") +1	; Por tanto: PROGIDX = 12
	; Con la función SUBSTR introducida en la siguiente sección es posible dividir la variable ENTRADA en los componentes ;"Ruta";y "Bloque":
VARIABLE = SUBSTR (ENTRADA, PFADIDX, PROGIDX-PFADIDX-1)	; devuelve "_N_MPF_DIR"
VARIABLE = SUBSTR (ENTRADA, PROGIDX)	; devuelve "_N_EXECUTE_MPF"

3.1.4.7 Selección de una cadena parcial (SUBSTR)

La función SUBSTRING permite leer una parte cualquiera de una cadena de caracteres.

Sintaxis

```
<STRING_ERG>=SUBSTR(<Cadena_de_caracteres>,<Índice>,<Longitud>)
<STRING_ERG>=SUBSTR(<Cadena_de_caracteres>,<Índice>)
```

Descripción

SUBSTR:	La función proporciona una cadena de caracteres parcial a partir de <Cadena_de_caracteres>, partiendo del <Índice> con la <Longitud> indicada. Si no se indica el parámetro <Longitud>, la función entrega una parte de la cadena de caracteres partiendo del <Índice> y hasta el final de la cadena.
<Índice>:	Posición inicial de la cadena de caracteres parcial dentro de la cadena de caracteres. Si la posición inicial se encuentra detrás del final de la cadena, se devuelve una cadena vacía (" "). Primer carácter de la cadena: Índice = 0 Rango de valores: 0 ... (longitud de la cadena - 1)
<Longitud>:	Longitud de la cadena parcial. Si se indica una longitud demasiado grande, solamente se devuelve la cadena parcial hasta el final de la cadena. Rango de valores: 1 ... (longitud de la cadena - 1)

Ejemplo

Código de programa	Comentarios
DEF STRING[29] RESULT	
; 1	
; 0123456789012345678	
RESULT = SUBSTR("ACUSE: 10 a 99", 10, 2)	; ERG == "10"

Código de programa	Comentarios
`RESULT = SUBSTR("ACUSE: 10 a 99", 10)`	`; ERG == "10 a 99"`

3.1.4.8 Lectura y escritura de caracteres individuales

Dentro de una cadena de caracteres pueden leerse y escribirse caracteres individuales.

Para ello, se deben observar las siguientes condiciones marginales:

- solo posible para variables definidas por el usuario, no para variables del sistema
- caracteres individuales de una cadena se transfieren solamente con llamadas de subprograma "call by value"

Sintaxis

```
<Carácter>=<Cadena_de_caracteres>[<Índice>]
<Carácter>=<Cadena_Array>[<Array_Índice>,<Índice>]
<Cadena_de_caracteres>[<Índice>]=<Carácter>
<Cadena_Array>[<Array_Índice>,<Índice>]=<Carácter>
```

Descripción

`<Cadena_de_caracteres>`:	Cualquier cadena de caracteres
`<Carácter>`:	Variable del tipo CHAR
`<Índice>`:	Posición del carácter dentro de la cadena de caracteres. Primer carácter de la cadena: Índice = 0 Rango de valores: 0 ... (longitud de la cadena - 1)

Ejemplos

Ejemplo 1: variable Aviso

Código de programa	Comentarios
`;  0123456789`	
`DEF STRING [50] AVISO = "El eje n ha alcanzado su posición"`	
`AVISO [6] = "X"`	
`MSG (AVISO)`	`; "El eje X ha alcanzado la posición"`

Ejemplo 2: evaluación de una variable de sistema

Código de programa	Comentarios
`DEF STRING [50] STRG`	`; Memoria intermedia para variable de sistema`
`...`	
`STRG = $P_MMCA`	`; Cargar variable de sistema`
`IF STRG[0] == "E" GOTO ...`	`; Evaluación de la variable de sistema`

Ejemplo 3: transferencia de parámetros "call by value" y "call by reference"

Código de programa	**Comentarios**
; 0123456	
DEF STRING[50] STRG = "Eje X"	
DEF CHAR CHR	
...	
EXTERN UP_VAL(EJE)	; Definición de subprograma con parámetro "call by value"
EXTERN UP_REF(VAR EJE)	; Definición de subprograma con parámetro "call by ref."
...	
UP_VAL(STRG[6])	; Transferencia de parámetros "by value"
...	
CHR = STRG[6]	; Guardar en búfer
UP_REF(CHR)	; Transferencia de parámetros "by reference"

3.1.4.9 Formatear cadena (SPRINT)

La función predefinida SPRINT permite formatear cadenas de caracteres y procesarlas, p. ej., para la salida a equipos externos (ver también "Process DataShare - Salida a un equipo/fichero externo (EXTOPEN, WRITE, EXTCLOSE): (Página 1031)").

Sintaxis

```
"<Resultado_cadena>"=SPRINT("<Formato_cadena>",<Valor_1>,<Valor_2>,..., <Valor_n>)
```

Descripción

SPRINT:	Identificador de una función predefinida que devuelve un valor del tipo STRING.
"<Formato_cadena>":	Cadena de caracteres que contiene partes fijas y variables. Las partes variables se definen mediante el carácter de control de formato % seguido de una descripción del formato.
<Valor_1>,<Valor_2>,…,<Valor_n>:	Valor en forma de constante o variable de CN que se inserta en el lugar en el que se encuentra el enésimo carácter de control de formato %, de acuerdo con la descripción de formato de <Formato_cadena>.
"<Resultado_cadena>":	Cadena de caracteres formateada (máx. 400 bytes)

Descripciones de formato disponibles

%B:	Conversión a la cadena "TRUE" si el valor que debe convertirse:
	• es distinto de 0;
	• no es una cadena vacía (en el caso de valores de cadena).
	Conversión a la cadena "FALSE" si el valor que debe convertirse:
	• es igual a 0;
	• es una cadena vacía.
	Ejemplo:
	`N10 DEF BOOL BOOL_VAR=1`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF BOOL_VAR:%B", BOOL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF BOOL_VAR:TRUE".

%C:	Conversión a un carácter ASCII.
	Ejemplo:
	`N10 DEF CHAR CHAR_VAR="X"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF CHAR_VAR:%C",CHAR_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF CHAR_VAR:X".

%D:	Conversión a una cadena con un valor entero (INTEGER).
	Ejemplo:
	`N10 DEF INT INT_VAR=123`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF INT_VAR:%D",INT_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF INT_VAR:123".

%<m>D:	Conversión a una cadena con un valor entero (INTEGER). La cadena tiene una longitud mínima de <m> caracteres. Las posiciones vacías se rellenan con espacios a la izquierda.
	Ejemplo:
	`N10 DEF INT INT_VAR=-123`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF INT_VAR:%6D",INT_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF INT_VAR:xx-123" (en el ejemplo, "x" representa un espacio).

%F:	Conversión a una cadena con un número con 6 decimales. Los decimales se redondean o se rellenan con 0, según proceda.
	Ejemplo:
	`N10 DEF REAL REAL_VAR=-1.2341234EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%F",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR: -1234.123400".

%<m>F:	Conversión a una cadena con un número con 6 decimales y una longitud total de al menos <m> caracteres. Los decimales se redondean o se rellenan con 0, según proceda. Las posiciones vacías se rellenan con espacios a la izquierda hasta alcanzar la longitud total <m>.
	Ejemplo:
	`N10 DEF REAL REAL_VAR=-1.23412345678EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15F",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR: xxx-1234.123457" (en el ejemplo, "x" representa un espacio).

%.<n>F:	Conversión a una cadena con un número con <n> decimales. Los decimales se redondean o se rellenan con 0, según proceda.
	Ejemplo:
	`N10 DEF REAL REAL_VAR=-1.2345678EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3F",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:-1234.568".

%<m>.<n>F:	Conversión a una cadena con un número con <n> decimales y una longitud total de al menos <m> caracteres. Los decimales se redondean o se rellenan con 0, según proceda. Las posiciones vacías se rellenan con espacios a la izquierda hasta alcanzar la longitud total <m>.
	Ejemplo:
	`N10 DEF REAL REAL_VAR=-1.2341234567890EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%10.2F",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:xx-1234.12" (en el ejemplo, "x" representa un espacio).

%E:	Conversión a una cadena con un número decimal en representación exponencial. El formato normalizado para guardar la mantisa consiste en un dígito entero y 6 decimales. Los decimales se redondean o se rellenan con 0, según proceda. El exponente comienza por la palabra reservada "EX". Le sigue el signo ("+" o "-") y un número de dos o tres dígitos.
	Ejemplo:
	`N10 DEF REAL REAL_VAR=-1234.567890`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%E",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:-1.234568EX+03".

%<m>E:	Conversión a una cadena con un número decimal en representación exponencial de una longitud total de al menos <m> caracteres. Los caracteres vacíos se rellenan con espacios a la izquierda. El formato normalizado para guardar la mantisa consiste en un dígito entero y 6 decimales. Los decimales se redondean o se rellenan con 0, según proceda. El exponente comienza por la palabra reservada "EX". Le sigue el signo ("+" o "-") y un número de dos o tres dígitos.
	Ejemplo:
	`N10 DEF REAL REAL_VAR=-1234.5`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%20E",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:xxxxxx-1.234500EX+03" (en el ejemplo, "x" representa un espacio).

%.<n>E:	Conversión a una cadena con un número decimal en representación exponencial. El formato normalizado para guardar la mantisa consiste en un dígito entero y <n> decimales. Los decimales se redondean o se rellenan con 0, según proceda. El exponente comienza por la palabra reservada "EX". Le sigue el signo ("+" o "-") y un número de dos o tres dígitos.
	Ejemplo:
	`N10 DEF REAL REAL_VAR=-1234.5678`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.2E",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:-1.23EX+03".

%<m>.<n>E:	Conversión a una cadena con un número decimal en representación exponencial de una longitud total de al menos <m> caracteres. Los caracteres vacíos se rellenan con espacios a la izquierda. El formato normalizado para guardar la mantisa consiste en un dígito entero y <n> decimales. Los decimales se redondean o se rellenan con 0, según proceda. El exponente comienza por la palabra reservada "EX". Le sigue el signo ("+" o "-") y un número de dos o tres dígitos.
	Ejemplo:
	`N10 DEF REAL REAL_VAR=-1234.5678`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.2E", REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:xx-1.23EX+03" (en el ejemplo, "x" representa un espacio).

%G:	Conversión a una cadena con un número decimal en representación decimal o exponencial, según el rango de valores: si el valor absoluto del número que debe mostrarse es inferior a 1.0EX-04 o superior/igual a 1.0EX+06, se elige la representación exponencial; en caso contrario, la representación decimal. Como máximo se muestran seis posiciones significativas; dado el caso, el número se redondea.
	Ejemplo en representación decimal:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:0.000123457".
	Ejemplo en representación exponencial:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+06`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:1.23457EX+06".

%<m>G:	Conversión a una cadena con un número decimal en representación decimal o exponencial, según el rango de valores (igual que %G). La cadena tiene una longitud total de al menos <m> caracteres. Los caracteres vacíos se rellenan con espacios a la izquierda.
	Ejemplo en representación decimal:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:xxxx0.000123457" (en el ejemplo, "x" representa un espacio).
	Ejemplo en representación exponencial:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+06`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:xxx1.23457EX+06" (en el ejemplo, "x" representa un espacio).

%.<n>G:	Conversión a una cadena con un número decimal en representación decimal o exponencial, según el rango de valores. Como máximo se muestran <n> posiciones significativas; dado el caso, el número se redondea. Si el valor absoluto del número que debe mostrarse es inferior a 1.0EX-04 o superior/igual a 1.0EX(+<n>), se elige la representación exponencial; en caso contrario, la representación decimal.
	Ejemplo en representación decimal:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:0.000123".
	Ejemplo en representación exponencial:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+03`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT = SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:1.23EX+03".

%<m>.<n>G:	Conversión a una cadena con un número decimal en representación decimal o exponencial, según el rango de valores (igual que %.<n>G). La cadena tiene una longitud total de al menos <m> caracteres. Los caracteres vacíos se rellenan con espacios a la izquierda.
	Ejemplo en representación decimal:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX-04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:xxx0.0001235" (en el ejemplo, "x" representa un espacio).
	Ejemplo en representación exponencial:
	`N10 DEF REAL REAL_VAR=1.234567890123456EX+04`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF REAL_VAR:xx1.235EX+06" (en el ejemplo, "x" representa un espacio).

%.<n>P:	Conversión de un valor REAL a un valor INTEGER teniendo en cuenta <n> decimales. El valor INTEGER se expresa como número binario de 32 bits. Si el valor que debe convertirse no puede representarse con 32 bits, la ejecución se interrumpe y se emite una alarma.
	Dado que una secuencia de bytes generada con la instrucción de formato %.<n>P también puede contener ceros binarios, la cadena completa generada ya no se ajusta a las convenciones del tipo de datos de CN STRING. En consecuencia, no se puede guardar en una variable de tipo STRING ni se puede seguir ejecutando con los comandos de cadena del lenguaje CN. La única posibilidad de uso es la transferencia de parámetros al comando WRITE con salida a un equipo externo correspondiente (ver ejemplo a continuación).
	En cuanto <Formato_cadena> contiene una descripción de formato del tipo %P, toda la cadena, a excepción del número binario generado con %.<n>P, se expresa en código de caracteres ASCII, ISO (DIN6024) o EIA (RS244) de acuerdo con MD10750 $MN_SPRINT_FORMAT_P_CODE. Si se programa un carácter no convertible, el mecanizado se interrumpe y se emite una alarma.
	Ejemplo:
	`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`
	`...  ; Tratamiento de errores`
	`N200 WRITE(ERROR,EXT_DEVICE,SPRINT("INTEGER BINARY CODED:%.3P",REAL_VAR)`
	`N210 IF ERROR <> 0`
	`… ; Tratamiento de errores`
	Resultado: la cadena "INTEGER BINARY CODED: 'H0001E23A'" se transfiere al equipo de salida /ext/dev/1. El valor hexadecimal 0x0001E23A equivale al valor decimal 123450.

%<m>.<n>P:	Conversión, según el ajuste del dato de máquina MD10751 $MN_SPRINT_FORMAT_P_DECIMAL, de un valor REAL a una cadena con:
	• un número entero de <m> + <n> dígitos **o bien**
	• un número decimal con un máximo de <m> dígitos antes de la coma y exactamente <n> decimales.
	Como en la descripción de formato %.<n>P, toda la cadena se guarda en el código de caracteres definido por MD10750 $MN_SPRINT_FORMAT_P_CODE.
	Conversión con MD10751 = 0:
	El valor REAL se convierte a una cadena con un número entero de <m> + <n> dígitos. Los decimales se redondean a <n> dígitos o se rellenan con 0, según proceda. Las posiciones vacías antes de la coma se rellenan con espacios. El signo menos se añade a la izquierda, mientras que el signo más se sustituye por un espacio.
	Ejemplo:
	`N10 DEF REAL REAL_VAR=-123.45`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("PUNCHED TAPE FORMAT:%5.3P",REAL_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "PUNCHED TAPE FORMAT:-xx123450" (en el ejemplo, "x" representa un espacio).
	Conversión con MD10751 = 1:
	El valor REAL se convierte a una cadena con un número decimal con un máximo de <m> dígitos antes de la coma y exactamente <n> decimales. Según proceda, los dígitos antes de la coma se cortan y los decimales se redondean o se rellenan con 0. Si <n> es igual a 0, el separador de decimales también se suprime.
	Ejemplo:
	`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)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "PUNCHED TAPE FORMAT:-123.450 VAR2:23".

%S:	Inserción de una cadena.
	Ejemplo:
	`N10 DEF STRING[16] STRING_VAR="ABCDEFG"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%S",STRING_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF STRING_VAR:ABCDEFG".

%<m>S:	Inserción de una cadena de al menos <m> caracteres. Las posiciones vacías se rellenan con espacios.
	Ejemplo:
	`N10 DEF STRING[16] STRING_VAR="ABCDEFG"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10S",STRING_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF STRING_VAR:xxxABCDEFG" (en el ejemplo, "x" representa un espacio).

%.<n>S:	Inserción de <n> caracteres de una cadena (comenzando por el primer carácter).
	Ejemplo:
	`N10 DEF STRING[16] STRING_VAR="ABCDEFG"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%.3S",STRING_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF STRING_VAR:ABC".

%<m>.<n>S:	Inserción de <n> caracteres de una cadena (comenzando por el primer carácter). La cadena generada tiene una longitud total de al menos <m> caracteres. Las posiciones vacías se rellenan con espacios.
	Ejemplo:
	`N10 DEF STRING[16] STRING_VAR="ABCDEFG"`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10.5S", STRING_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "CONTENT OF STRING_VAR:xxxxxABCDE" (en el ejemplo, "x" representa un espacio).

%X:	Conversión de un valor INTEGER a una cadena en representación hexadecimal.
	Ejemplo:
	`N10 DEF INT INT_VAR='HA5B8'`
	`N20 DEF STRING[80] RESULT`
	`N30 RESULT=SPRINT("INTEGER HEXADECIMAL:%X",INT_VAR)`
	Resultado: la variable de tipo cadena RESULT se describe con la cadena de caracteres "INTEGER HEXADECIMAL:A5B8".

Nota

La propiedad del lenguaje CN de no diferenciar entre mayúsculas y minúsculas en identificadores y palabras reservadas también es válida para las descripciones de formato. En consecuencia, se puede programar en mayúsculas o minúsculas sin diferencia funcional.

Posibilidades de combinación

En la siguiente tabla se ofrece información sobre las combinaciones posibles entre tipos de datos de CN y descripciones de formato. Se aplican las reglas para la conversión implícita de tipos de datos (ver "Tipos de datos (Página 436)").

	Tipos de datos de CN						
	BOOL	CHAR	INT	REAL	STRING	AXIS	FRAME
%B	+	+	+	+	+	-	-
%C	-	+	-	-	+	-	-
%D	+	+	+	+	-	-	-
%F	-	-	+	+	-	-	-
%E	-	-	+	+	-	-	-
%G	-	-	+	+	-	-	-
%S	-	+	-	-	+	-	-
%X	+	+	+	-	-	-	-
%P	-	-	+	+	-	-	-

Nota

La tabla indica que los tipos de datos de CN AXIS y FRAME no pueden utilizarse directamente en la función SPRINT. Sin embargo, es posible:

- Convertir, con la función AXSTRING, el tipo de datos AXIS a una cadena que pueda procesarse luego con SPRINT.
- Leer los distintos valores del tipo de datos FRAME mediante el acceso a componentes frame. De este modo se obtiene un dato del tipo REAL que puede procesarse con SPRINT.

3.1.5 saltos y derivaciones del programa

3.1.5.1 Salto atrás hasta el inicio del programa (GOTOS)

El comando `GOTOS` permite saltar atrás hasta el principio de un programa principal o subprograma para repetir dicho programa.

Mediante datos de máquina puede configurarse que, con cada salto atrás hasta el inicio del programa:

- el tiempo de ejecución del programa se ajuste a "0";
- el cómputo de piezas se incremente el valor "1".

Sintaxis

```
GOTOS
```

Descripción

`GOTOS`:	Instrucción de salto hasta el inicio del programa.	
	La ejecución se controla mediante la señal de interfaz CN/PLC. DB21, ... DBX384.0 (Controlar bifurcación de programa)	
	Valor:	Significado:
	0	Sin salto atrás hasta el inicio del programa. La ejecución del programa continúa con la secuencia del programa de pieza que sigue a `GOTOS`.
	1	Salto atrás hasta el inicio del programa. Se repite el programa de pieza.

Condiciones

- GOTOS desencadena internamente una STOPRE (parada de decodificación previa).
- En un programa de pieza con definiciones de datos (variables LUD) con GOTOS se salta a la primera secuencia de programa que sigue a la sección de definición, es decir, las definiciones de datos no se ejecutan de nuevo. Con ello, las variables definidas mantienen el valor alcanzado en la secuencia GOTOS y no se restablecen al valor estándar programado en la sección de definición.
- El comando GOTOS no está disponible en las acciones síncronas ni en los ciclos tecnológicos.

Ejemplo

Código de programa	**Comentarios**
N10 ...	; Inicio del programa.
...	
N90 GOTOS	; Salto al inicio del programa.
...	

3.1.5.2 Saltos de programa a marcas de salto (GOTOB, GOTOF, GOTO, GOTOC)

Es posible fijar marcas de salto (etiquetas) en un programa a las que se puede saltar desde otros puntos del mismo programa con los comandos GOTOF, GOTOB, GOTO oGOTOC. La ejecución del programa continúa con la instrucción inmediatamente posterior a la marca de salto. De este modo es posible realizar derivaciones dentro del programa.

Además de las marcas de salto, también son posibles números de secuencia principal y de secuencia auxiliar como destinos del salto.

Si, antes de la instrucción de salto, hay formulada una condición para el salto (IF ...), el salto del programa tendrá lugar sólo si se cumple la condición para el salto.

Sintaxis

```
GOTOB <Destino del salto>
IF <Condición para el salto> == TRUE GOTOB <Destino del salto>

GOTOF <Destino del salto>
IF <Condición para el salto> == TRUE GOTOF <Destino del salto>

GOTO <Destino del salto>
IF <Condición para el salto> == TRUE GOTO <Destino del salto>

GOTOC <Destino del salto>
IF <Condición para el salto> == TRUE GOTOC <Destino del salto>
```

Descripción

GOTOB:	Instrucción de salto con destino de salto hacia el principio del programa.	
GOTOF:	Instrucción de salto con destino de salto hacia el final del programa.	
GOTO:	Instrucción de salto con búsqueda de destino de salto. La búsqueda se realiza en primer lugar hacia el final del programa y, a continuación, hacia el principio del programa.	
GOTOC:	Efecto idéntico que con GOTO con la diferencia de que se suprime la alarma 14080 "Destino del salto no encontrado". Esto significa que, la ejecución del programa no se interrumpe en caso de que la búsqueda del destino de salto no dé ningún resultado, sino que continúa con la siguiente línea de programa del comando GOTOC.	
<Destino del salto>:	Parámetro de destino de salto Posibles indicaciones:	
	<Marca de salto>:	El destino del salto es la marca de salto definida en el programa con un nombre definido por el usuario:<Marca de salto>:
	<Número de secuencia>:	El destino del salto es un número de secuencia principal o auxiliar (p. ej.: 200, N300)
	Variable del tipo STRING:	Variable Destino del salto. La variable corresponde a una marca de salto o un número de secuencia.
IF:	Palabra reservada para formular la condición de salto. La condición para el salto permite utilizar todos los operadores de comparación y lógicos (resultado: TRUE o FALSE). El salto del programa se realiza si el resultado de la operación es TRUE.	

Nota

Marcas de salto (etiquetas)

Las marcas de salto se encuentran siempre al principio de la secuencia. Cuando existe un número de programa, la marca de salto se encuentra inmediatamente a continuación del número de la secuencia.

Para la denominación de marcas de salto deben respetarse las siguientes reglas:

- Número de caracteres:
  - como mínimo 2
  - como máximo 32
- Caracteres permitidos:
  - Letras
  - Números
  - Caracteres de subrayado
- Los dos primeros caracteres deben ser letras o caracteres de subrayado.
- Al final del nombre de la marca de salto se deben introducir dos puntos (":").

Condiciones

- El destino del salto solamente puede ser una secuencia cuya marca de salto o número de secuencia se encuentre **dentro** de dicho programa.
- La instrucción de salto sin condición de salto debe programarse en una secuencia separada. Esta limitación no se aplica en el caso de instrucciones de salto con condiciones de salto. Aquí pueden formularse varias instrucciones de salto en una secuencia.
- En programas con instrucciones de salto sin condiciones de salto, el final del programa `M2`/`M30` no debe encontrarse forzosamente al final del programa.

Ejemplos

Ejemplo 1: saltos a marcas de salto

```
Código de programa                Comentarios
N10…
N20 GOTOF Label_1                 ; Salto hacia el final del programa, hasta
                                  ; la marca de salto "Label_1".
N30 …
N40 Label_0: R1=R2+R3             ; Marca de salto "Label_0" fijada.
N50 …
N60 Label_1:                      ; Marca de salto "Label_1" fijada.
N70 …
N80 GOTOB Label_0                 ; Salto hacia el principio del programa
                                  ; hasta la marca de salto "Label_0".
N90 …
```

Ejemplo 2: salto indirecto a número de secuencia

```
Código de programa                Comentarios
IF <Condición> == TRUE
   R10=100                        ; Asignar el destino del salto
ELSE
   R10=110                        ; Asignar el destino del salto
ENDIF
; Salto hacia el final del programa a la secuencia cuyo número figura en R10
N10 GOTOF "N"<<R10
...
N90 ...
N100 ...                          ; Destino del salto
N110 ...
...
```

Ejemplo 3: salto a un destino de salto variable

```
Código de programa                Comentarios
DEF STRING[20] DESTINO
```

Código de programa	Comentarios
IF <Condición> == TRUE	
DESTINO = "Marca1"	; Asignar el destino del salto
ELSE	
DESTINO = "Marca2"	; Asignar el destino del salto
ENDIF	
; Salto hacia el final del programa, hasta el destino de salto variable "Contenido de DESTINO"	
GOTOF DESTINO	
Marca1: T="Broca1"	; Destino del salto 1
...	
Marca2: T="Broca2"	; Destino del salto 2
...	

Ejemplo 4: salto con condición de salto

Código de programa	Comentarios
N40 R1=30 R2=60 R3=10 R4=11 R5=50 R6=20	; Asignación de los valores iniciales
N41 LA1: G0 X=R2*COS(R1)+R5 Y=R2*SIN(R1)+R6	; Marca de salto LA1
N42 R1=R1+R3 R4=R4-1	
; SI condición para el salto == TRUE	
; ENTONCES salto hacia el inicio del programa hasta la marca de salto LA1	
N43 IF R4>0 GOTOB LA1	
N44 M30	; Fin del programa

3.1.5.3 Bifurcación de programa (CASE ... OF ... DEFAULT ...)

La función CASE ofrece la posibilidad de comprobar el valor actual (tipo: INT) de una variable o una función de cálculo, así como de saltar a diferentes lugares del programa en función del resultado.

Sintaxis

```
CASE(<expresión>) OF <Constante_1> GOTOF <Destino del salto_1>
<Constante_2> GOTOF <Destino del salto_2> ... DEFAULT GOTOF <Destino
del salto_n>
```

Descripción

CASE:	Instrucción de salto	
<Expresión>:	Variable o función de cálculo	
OF:	Palabra reservada para formular las derivaciones de programa condicionadas	
<Constante_1>:	Primer valor constante indicado para la variable o la función de cálculo	
	Tipo:	INT

<Constante_2>:	Segundo valor constante indicado para la variable o la función de cálculo	
	Tipo:	INT
DEFAULT:	En el caso de que la variable o la función de cálculo no tomen ninguno de los valores constantes indicados, se puede definir un destino de salto con la instrucción DEFAULT. **Nota:** Si no está programada la instrucción DEFAULT, el destino de salto en estos casos será la secuencia posterior a la instrucción CASE.	
GOTOF:	Instrucción de salto con destino de salto hacia el final del programa. En lugar de GOTOF pueden programarse también todos los demás comandos GOTO (ver el tema "Saltos de programa a marcas de salto").	
<Destino del salto_1>:	Se deriva a este destino de salto cuando el valor de las variables o de la función de cálculo corresponde a la primera constante indicada. El destino de salto se puede indicar como sigue:	
	<Marca de salto>:	El destino del salto es la marca de salto definida en el programa con un nombre definido por el usuario: <Marca de salto>:
	<Número de secuencia>:	El destino del salto es un número de secuencia principal o auxiliar (p. ej.: 200, N300)
	Variable del tipo STRING:	Variable Destino del salto. La variable corresponde a una marca de salto o un número de secuencia.
<Destino del salto_2>:	Se deriva a este destino de salto cuando el valor de las variables o de la función de cálculo corresponde a la segunda constante indicada.	
<Destino del salto_n>:	Se deriva a este destino de salto cuando el valor de las variables no adopta ninguno de los valores constantes indicados.	

Ejemplo

Código de programa

```
...
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
...
```

La instrucción CASE de N30 define las siguientes opciones de derivación de programa:

1. Si el valor de la función de cálculo VAR1+VAR2-VAR3 = 7, salta a la secuencia con la definición de marcas de salto "Label_1" (→ N40).
2. Si el valor de la función de cálculo VAR1+VAR2-VAR3 = 9, salta a la secuencia con la definición de marcas de salto "Label_2" (→ N50).
3. Si el valor de la función de cálculo VAR1+VAR2-VAR3 no es ni 7 ni 9, salta a la secuencia con la definición de marcas de salto "Label_3" (→ N60).

3.1.6 Repetición de secciones de programa (REPEAT, REPEATB, ENDLABEL, P)

La función de repetición de secciones de programa permite repetir, en los puntos y formas que se deseen, partes ya escritas.

Las líneas o áreas del programa que deben repetirse se identifican con marcas de salto (etiquetas).

Nota

Marcas de salto (etiquetas)

Las marcas de salto se encuentran siempre al principio de la secuencia. Cuando existe un número de programa, la marca de salto se encuentra inmediatamente a continuación del número de la secuencia.

Para la denominación de marcas de salto deben respetarse las siguientes reglas:

- Número de caracteres:
  - como mínimo 2
  - como máximo 32
- Caracteres permitidos:
  - Letras
  - Números
  - Caracteres de subrayado
- Los dos primeros caracteres deben ser letras o caracteres de subrayado.
- Al final del nombre de la marca de salto se deben introducir dos puntos (":").

Sintaxis

1. Repetir una sola línea de programa:

```
<Marca de salto>: ...
...
REPEATB <Marca de salto> P=<n>
...
```

2. Repetir un área del programa situada entre la marca de salto y la instrucción REPEAT:

```
<Marca de salto>: ...
...
REPEAT <Marca de salto> P=<n>
...
```

3. Repetir un área situada entre dos marcas de salto:

```
<Marca de salto inicial>: ...
...
<Marca de salto final>: ...
...
```

```
REPEAT <Marca de salto inicial> <Marca de salto final>
P=<n>
...
```

Nota

No es posible acotar la instrucción REPEAT con las dos marcas de salto. Si la <Marca de salto inicial> se encuentra delante de la instrucción REPEAT y no se llega a la <Marca de salto final> antes de la instrucción REPEAT, se realiza la repetición entre la <Marca de salto inicial> y la instrucción REPEAT.

4. Repetir un área situada entre la marca de salto y ENDLABEL:

```
<Marca de salto>: ...
...
ENDLABEL: ...
...
REPEAT <Marca de salto> P=<n>
...
```

Nota

No es posible acotar la instrucción REPEAT con la <Marca de salto> y ENDLABEL. Si la <Marca de salto> se encuentra delante de la instrucción REPEAT y no se llega a ENDLABEL antes de la instrucción REPEAT, se realiza la repetición entre la <Marca de salto> y la instrucción REPEAT.

Descripción

REPEATB:	Comando para repetir una línea de programa
REPEAT:	Comando para repetir una área de programa
<Marca de salto>:	La <Marca de salto> identifica: • la línea de programa que debe repetirse (con REPEATB) o bien • el principio del área de programa que debe repetirse (con REPEAT)
	La línea de programa identificada con la <Marca de salto> puede estar delante o detrás de la instrucción REPEAT/REPEATB. La búsqueda comienza en sentido al inicio del programa. Si no se encuentra la marca de salto en esta dirección, se busca en dirección al final de programa. **Excepción:** Si el área de programa situada entre la marca de salto y la instrucción REPEAT debe repetirse (ver punto 2 del apartado Sintaxis), la línea de programa identificada con la <Marca de salto> debe situarse **delante** de la instrucción REPEAT, ya que en este caso **solo** se busca en dirección al inicio del programa. Si la línea con la <Marca de salto> tiene más instrucciones, éstas se vuelven a ejecutar en cada repetición.

ENDLABEL:	Palabra reservada que marca el final de una área de programa que debe repetirse. Si la línea con ENDLABEL tiene más instrucciones, éstas se vuelven a ejecutar en cada repetición. ENDLABEL puede utilizarse varias veces en el programa.
P:	Dirección para indicar el número de repeticiones
<n>:	Número de repeticiones de secciones de programa
	Tipo: INT
	La sección de programa que debe repetirse se repite <n> veces. Tras la última repetición, el programa continúa ejecutándose a partir de la línea siguiente a la línea REPEAT/REPEATB. **Nota:** Si no está indicado ningún P=<n>, la sección de programa que debe repetirse se repite exactamente una vez.

Ejemplos

Ejemplo 1: repetir una sola línea de programa

```
Código de programa                  Comentarios
N10 POSITION1: X10 Y20
N20 POSITION2: CYCLE(0,,9,8)        ; Ciclo de posicionado
N30 ...
N40 REPEATB POSITION1 P=5           ; Ejecutar cinco veces la secuencia N10.
N50 REPEATB POSITION2               ; Ejecutar una vez la secuencia N20.
N60 ...
N70 M30
```

Ejemplo 2: repetir un área del programa situada entre la marca de salto y la instrucción REPEAT

```
Código de programa              Comentarios
N5 R10=15
N10 Begin: R10=R10+1            ; Ancho
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            ; Ejecutar cuatro veces el área comprendida entre N10 y
                                N70.
N90 Z10
N100 M30
```

Ejemplo 3: repetir un área situada entre dos marcas de salto

Código de programa	Comentarios
N5 R10=15	
N10 Begin: R10=R10+1	; Ancho
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	; Ejecutar tres veces el área comprendida entre N10 y N70.
N110 Z10	
N120 M30	

Ejemplo 4: repetir un área situada entre la marca de salto y ENDLABEL

Código de programa	Comentarios
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	; Ejecutar tres veces el área comprendida entre N110 y N120.
N140 REPEAT BEGIN2 P=2	; Ejecutar dos veces el área comprendida entre N50 y N80.
N150 M100	
N160 REPEAT BEGIN1 P=2	; Ejecutar dos veces el área comprendida entre N20 y N80.
N170 Z10	
N180 X0 Y0	
N190 M30	

Ejemplo 5: fresado, mecanizar posición de taladrado con diferentes tecnologías

Código de programa	Comentarios
N10 CENTRADORA()	; Cargar broca de puntear.

Código de programa	Comentarios
N20 POS_1:	; Posiciones de taladrado 1
N30 X1 Y1	
N40 X2	
N50 Y2	
N60 X3 Y3	
N70 ENDLABEL:	
N80 POS_2:	; Posiciones de taladrado 2
N90 X10 Y5	
N100 X9 Y-5	
N110 X3 Y3	
N120 ENDLABEL:	
N130 BROCA()	; Cambiar broca y ciclo de taladrado.
N140 MACHO(6)	; Cargar macho de roscar M6 y ciclo de roscado.
N150 REPEAT POS_1	; Repetir una vez la sección del programa comprendida entre POS_1 y ENDLABEL.
N160 BROCA()	; Cambiar broca y ciclo de taladrado.
N170 MACHO(8)	; Cargar macho de roscar M8 y ciclo de roscado.
N180 REPEAT POS_2	; Repetir una vez la sección del programa comprendida entre POS_2 y ENDLABEL.
N190 M30	

Información adicional

- Las repeticiones de secciones de programa pueden imbricarse. Cada llamada ocupa un nivel de subprograma.
- Si durante la ejecución de una repetición de sección hay programada `M17` o `RET`, se interrumpe dicha repetición. En tal caso, el programa continúa ejecutándose a partir de la secuencia que sigue a la línea `REPEAT`.
- En la indicación actual del programa, la repetición de la sección de programa se indica como nivel de subprograma propio.
- Si durante la ejecución de una sección del programa se activa una interrupción de nivel, el programa continúa ejecutándose tras la instrucción de llamada de la sección del programa. Ejemplo:

Código de programa	Comentarios
N5 R10=15	
N10 BEGIN: R10=R10+1	; Ancho
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	; Interrupción de nivel
N50 X=-R10	
N60 Y=-R10	
N70 END: Z10	
N80 Z10	

Código de programa	Comentarios
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	
N120 Z10	; Continuar con la ejecución del programa.
N130 M30	

- Es posible combinar estructuras de control y repeticiones de secciones del programa. Sin embargo, no puede haber solapamientos. Una repetición de sección de programa conviene que esté situada en una ramificación de estructura de control o una estructura de control dentro de una repetición de sección de programa.
- Si se mezclan saltos y repeticiones de secciones de programa, las secuencias se ejecutan de forma secuencial pura. Por ejemplo, si se salta desde una repetición de sección de programa, la ejecución continúa hasta que se encuentre el fin de sección programado. Ejemplo:

Código de programa
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

Nota

La instrucción `REPEAT` se debería situar detrás de las secuencias de desplazamiento.

3.1.7 Estructuras de control

De forma estándar, el control numérico ejecuta las secuencias de CN en el orden programado.

Es posible variar este orden mediante la programación de bloques y bucles de programa alternativos. La programación de estas estructuras de control se realiza con las palabras reservadas IF, ELSE, ENDIF, LOOP, FOR, WHILE y REPEAT.

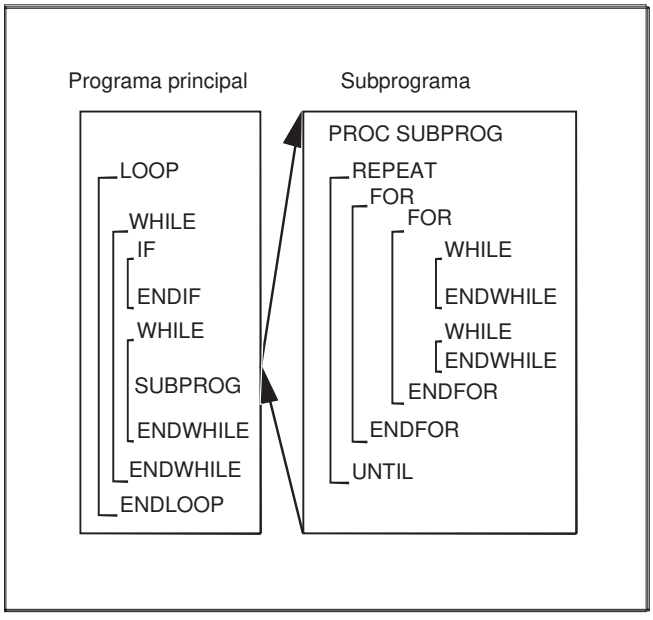
ATENCIÓN
Error de programación Solamente se pueden utilizar estructuras de control dentro del bloque de programa que contiene las instrucciones. Definiciones en la parte cabecera del programa no pueden ejecutarse condicional o repetidamente. No se deberán solapar con macros las palabras reservadas para estructuras de control ni tampoco las metas de los saltos. Al definir las macros no se comprueba este punto.

Efecto

Las estructuras de control no pueden usarse para varios programas.

Profundidad de imbricado

Dentro de cada llamada a un subprograma es posible realizar un nivel de imbricación de hasta 16 estructuras de control.



Comportamiento en cuanto a tiempo de ejecución

El tiempo necesario para la ejecución de un determinado programa en el modo Intérprete, el estándar, se puede acortar sensiblemente mediante la utilización de saltos en vez de estructuras de control.

Al utilizar ciclos precompilados, no hay diferencia entre saltos y estructuras de control.

Visualización de secuencia actual para bucles

Si dentro de un bucle se ejecutan solo secuencias de tratamiento previo, en la visualización actual de secuencia se muestra la última secuencia del proceso principal **antes** del bucle.

Para que las secuencias de decodificación ejecutadas sean visibles también en la visualización actual, p. ej., para fines de diagnóstico, debe activarse la decodificación secuencia a secuencia SBL2.

Bucles sin secuencia de proceso principal

Si, dentro de un bucle de programa, no se programa ninguna secuencia del proceso principal, el bucle se ejecutará en la decodificación previa hasta que se cumpla la condición de bucle.

Esto provoca una sobreutilización de la capacidad, que puede perjudicar la visualización.

La solución sería insertar el comando STOPRE o un tiempo de espera G04 de 0 segundos **en** el bucle.

Condiciones

- Las secuencias que contengan elementos de estructuras de control no pueden ser opcionales.
- Las marcas de salto (etiquetas) no están permitidas en las secuencias con elementos de estructuras de control.
- Las estructuras de control se ejecutan de forma interpretativa. El sistema, al reconocer el final de un bucle, buscará automáticamente el inicio del bucle, atendiendo a las estructuras de control halladas. Por ello, en el modo Intérprete no se realiza una comprobación completa de la estructura de bloques de un programa.
- Se aconseja no utilizar de forma mixta saltos y estructuras de control.
- Durante el preprocesamiento previo de ciclos se puede comprobar la imbricación correcta de estructuras de control.

3.1.7.1 Instrucción condicional y bifurcación (IF, ELSE, ENDIF)

Instrucción condicional: IF - bloque de programa - ENDIF

En caso de instrucción condicional, el bloque de programa situado entre `IF` y `ENDIF` se ejecuta únicamente si se cumple la condición.

Bifurcación: IF - Bloque de programa_1 - ELSE - Bloque de programa_2 - ENDIF

En caso de bifurcación, se ejecuta siempre uno solo de los dos bloques de programa.

Si se cumple la condición, se ejecuta el bloque de programa_1 situado entre `IF` y `ELSE`.

Si **no** se cumple la condición, se ejecuta el bloque de programa_2 situado entre `ELSE` y `ENDIF`.

Nota

ELSE en acciones síncronas

La palabra clave ELSE también puede programarse en acciones síncronas. Esto permite ampliar una acción síncrona con acciones que deben ejecutarse en caso de no cumplirse la condición.

Sintaxis

Instrucción condicional

```
IF <Condición>
   Programmblock                          ; Ejecución en caso de <condición> == TRUE
ENDIF
```

Bifurcación

```
IF <Condición>
   Programmblock_1                        ; Ejecución en caso de <condición> == TRUE
ELSE
   Programmblock_2                        ; Ejecución en caso de <condición> == FAL-
                                          SE
ENDIF
```

Descripción

`IF`:	Inicia la instrucción condicional o bifurcación.
`ELSE`:	Inicia el bloque de programa alternativo.
`ENDIF`:	Señaliza el final de la instrucción condicional o bifurcación.
`<Condición>`:	Expresión lógica cuya evaluación da como resultado TRUE o FALSE.

Ejemplo: Subprograma de cambio de herramienta

Código de programa	**Comentarios**
`PROC L6`	`Rutina de cambio de herramienta`
`N500 DEF INT TNR_AKTUELL`	`Variable para número T activo`
`N510 DEF INT TNR_VORWAHL`	`Variable para número T predefini-do`
	`Determinar la herramienta actual`
`N520 STOPRE`	
`N530 IF $P_ISTEST`	`En el modo de test del progra-ma...`
`N540  TNR_AKTUELL = $P_TOOLNO`	`...se lee la herramienta "ac-tual" en el contexto de programa.`

Código de programa	Comentarios
N550 ELSE	De lo contrario...
N560 TNR_AKTUELL = $TC_MPP6[9998,1]	...se lee la herramienta del cabezal.
N570 ENDIF	
N580 GETSELT(TNR_VORWAHL)	Leer el número T de la herramienta predefinida del cabezal.
N590 IF TNR_AKTUELL <> TNR_VORWAHL	Si la herramienta predefinida no es todavía la herramienta actual...
N600 G0 G40 G60 G90 SUPA X450 Y300 Z300 D0	... Posicionamiento en el punto de cambio de herramienta...
N610 M206	...y ejecución del cambio de herramienta.
N620 ENDIF	
N630 M17	

3.1.7.2 Bucle de programa sin fin (LOOP, ENDLOOP)

El bucle sin fin se utiliza para programas que se deban ejecutar constantemente. Al final del bucle siempre se realiza un salto hacia atrás hasta el principio del bucle.

Sintaxis

```
LOOP
...
ENDLOOP
```

Descripción

LOOP:	Inicia el bucle sin fin.
ENDLOOP:	Marca el final del bucle y salta atrás hasta el inicio del bucle.

Ejemplo

Código de programa
...
LOOP
MSG("Sin filos de herramienta activos")
M0
STOPRE
ENDLOOP
...

3.1.7.3 Bucle contador (FOR ... TO ..., ENDFOR)

El bucle contador se utiliza cuando se debe realizar una determinada cantidad de pasadas.

Sintaxis

```
FOR <Variable> = <Valor inicial> TO <Valor final>
...
ENDFOR
```

Descripción

FOR:	Inicia el bucle contador.	
ENDFOR:	Marca el final del bucle y realiza un salto atrás hasta el inicio del bucle, mientras no se haya alcanzado el valor final del recuento.	
<Variable>:	Variable de contador que se incrementa en cada pasada el valor "1", desde su valor inicial hasta su valor final.	
	Tipo	INT o REAL **Nota:** El tipo REAL se toma cuando se programan, p. ej., parámetros R para un bucle contador. Si la variable de contador es del tipo REAL su valor se redondea a un valor entero.
<Valor inicial>:	Valor inicial del recuento Condición: El valor inicial deberá ser menor que el valor final.	
<Valor final>:	Valor final del recuento	

Ejemplos

Ejemplo 1: variable INTEGER o parámetro R como variable de contador

Variable INTEGER como variable de contador:

Código de programa	**Comentarios**
DEF INT iVARIABLE1	
R10=R12-R20*R1 R11=6	
FOR iVARIABLE1 = R10 TO R11	; Variable de contador = variable INTEGER
R20=R21*R22+R33	
ENDFOR	
M30	

Parámetro R como variable de contador:

Código de programa	**Comentarios**
R11=6	
FOR R10=R12-R20*R1 TO R11	; Variable de contador = parámetro R (variable real)

Código de programa	Comentarios
R20=R21*R22+R33	
ENDFOR	
M30	

Ejemplo 2: mecanizado de un número de piezas determinado

Código de programa	Comentarios
DEF INT NÚMERO PIEZAS	; Variable definida del tipo INT con el nombre "NÚMERO PIEZAS".
FOR NÚMERO PIEZAS = 0 TO 100	; Inicia el bucle contador. La variable "NÚMERO PIEZAS" se incrementa desde el valor inicial "0" hasta el valor final "100".
G01 …	
ENDFOR	; Fin del bucle contador.
M30	

3.1.7.4 Bucle de programa con la condición en el inicio del bucle (WHILE, ENDWHILE)

En un bucle WHILE, la condición se encuentra en el inicio del bucle. Mientras se cumpla la condición se ejecutará el bucle WHILE.

Sintaxis

```
WHILE <Condición>
...
ENDWHILE
```

Descripción

WHILE:	Inicia el bucle de programa.
ENDWHILE:	Marca el final del bucle y salta atrás hasta el inicio del bucle.
<Condición>:	Condición que debe cumplirse para que se ejecute el bucle WHILE.

Ejemplo

Código de programa	Comentarios
...	
WHILE $AA_IW[EJE TALADRADO] > -10	; Llamada del bucle WHILE con la siguiente condición: la consigna WKS para el eje de taladrado debe ser mayor que -10.
G1 G91 F250 AX[EJE TALADRADO] = -1	
ENDWHILE	
...	

3.1.7.5 Bucle de programa con la condición al final del bucle (REPEAT, UNTIL)

En un bucle REPEAT, la condición se encuentra al final del bucle. El bucle REPEAT se ejecuta una primera vez y se repite la ejecución hasta que la condición indicada al final se cumpla.

Sintaxis

```
REPEAT
...
UNTIL <Condición>
```

Descripción

REPEAT:	Inicia el bucle de programa.
UNTIL:	Marca el final del bucle y salta atrás hasta el inicio del bucle.
<Condición>:	Condición que debe cumplirse para que deje de ejecutarse el bucle REPEAT.

Ejemplo

```
Código de programa          Comentarios
...
REPEAT                      ; Llamada del bucle REPEAT.
...
UNTIL ...                   ; Comprobación del cumplimiento de la condición.
...
```

3.1.7.6 Ejemplo de programa con estructuras de control imbricadas

```
Código de programa                                  Comentarios
LOOP
   IF NOT $P_SEARCH                                 ; IF no hay búsqueda de secuencia
      G1 G90 X0 Z10 F1000
      WHILE $AA_IM[X] <= 100                        ; WHILE (consigna eje X <= 100)
         G1 G91 X10 F500                            ; Figura de taladrado
         Z-5 F100
         Z5
      ENDWHILE
   ELSE                                             ; ELSE búsqueda de secuencia
     MSG("Durante la búsqueda de secuencia no
se taladra")
   ENDIF                                            ; ENDIF
   $A_OUT[1] = 1                                    ; Siguiente chapa a taladrar
   G4 F2
ENDLOOP
```

Código de programa	Comentarios
M30	

3.1.8 Coordinación de programa en todos los canales (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)

En principio, un canal del CN puede ejecutar el programa que se ha iniciado en él independientemente de otros canales de su grupo de modos de operación (GMO). No obstante, si varios programas participan simultáneamente en la fabricación de una pieza en varios canales del GMO, es necesario coordinar las ejecuciones de programas en los diferentes canales con las órdenes de coordinación siguientes.

Requisitos

Todos los canales participantes en la coordinación de programas deben pertenecer al **mismo** grupo de modos de operación (GMO).

MD10010 $MC_ASSIGN_CHAN_TO_MODE_GROUP[<Canal>] = <Número GMO>

Nombre de canal en lugar de número de canal

En lugar de los números de canal, también pueden utilizarse los nombres de canal introducidos en MD20000 $MC_CHAN_NAME[<Índice de canal>] como parámetros de los procedimientos predefinidos de la coordinación de programas. Primero debe habilitarse la utilización de los nombres de canal en los programas CN:

MD10280 $MN_PROG_FUNCTION_MASK, Bit 1 = TRUE

Nota

Distancia mínima entre comandos

Entre los comandos `INIT`, `START`, `WAITE`, `WAITM`, `SETM`, `CLEARM` y el comando `WAITMC` debe mantenerse una distancia mínima de dos secuencias de desplazamiento. WAITMC es una secuencia ejecutable, pero se desplaza a la secuencia anterior para fines de optimización y, en tal caso, se borra como secuencia. `SETM`, por ejemplo, es una secuencia no ejecutable y se desplaza a la secuencia siguiente, de modo que, con una distancia de una secuencia, ambos comandos estarían en la secuencia intermedia. Dado que solo es posible una secuencia, en caso de distancia de una secuencia para `WAITMC`, no se lleva a cabo la optimización.

De esta forma el programa se frena y el mecanizado se detiene brevemente.

Sintaxis

```
INIT(<ChanNr>, <Prog>, <AckMode>)
START(<ChanNr>, <ChanNr>, ...)
WAITM(<MarkNr>, <ChanNr>, <ChanNr>, ...)
WAITE(<ChanNr>, <ChanNr>, ...)
WAITMC(<MarkNr>, <ChanNr>, <ChanNr>, ...)
SETM(<MarkNr>, <MarkNr>, ...)
CLEARM(<MarkNr>, <MarkNr>, ...)
```

Descripción

`INIT()`:	Procedimiento predefinido para elegir el programa CN que debe ejecutarse en el canal indicado		
`START()`:	Procedimiento predefinido para el arranque del programa seleccionado en el canal correspondiente		
`WAITM()`:	Procedimiento predefinido para esperar a que se alcance una meta de espera en los canales indicados En el propio canal se activa la meta de espera indicada con `WAITM`. La secuencia anterior finaliza con parada precisa. Tras la sincronización se borra la meta de espera. Simultáneamente se puede aplicar un máximo de 10 metas por canal.		
`WAITE()`:	Procedimiento predefinido para esperar el fin de programa en uno o varios de los otros canales		
`WAITMC()`:[1)]	Procedimiento predefinido para esperar a que se alcance una meta de espera en los canales indicados Al contrario que con `WAITM`, el frenado de los ejes en la parada precisa solo se activa si los otros canales no han alcanzado aún la meta de espera.		
`SETM()`:[1)]	Procedimiento predefinido para ajustar una o varias metas de espera para la coordinación de canales El mecanizado en el canal propio no se verá afectado por esto. `SETM` mantiene su vigencia tras un reset de canal y un arranque de CN.		
`CLEARM()`:[1)]	Procedimiento predefinido para borrar una o varias metas de espera de la coordinación de canales El mecanizado en el canal propio no se verá afectado por esto. `CLEARM()` borra todas las metas de espera en el canal. `CLEARM(0)` solo borra la meta de espera "0". `CLEARM` mantiene su vigencia tras un reset de canal y un arranque de CN.		
`<ChanNr>`:	Número de canal No es necesario indicar el número del propio canal.		
	Tipo:	INT	
`<Prog>`:	Ruta de acceso absoluta o relativa (**opcional**) + nombre de programa		
	Tipo:	STRING	
	Encontrará información sobre la ruta en el Manual de programación Programación CN, capítulo "Direccionamiento de ficheros de la memoria de programas" (Página 570).		
`<AckMode>`:	Modo de acuse (**opcional**)		
	Tipo:	CHAR	
	Valores:	"N"	Sin acuse La ejecución del programa continúa cuando se ha transmitido el comando. El componente que se está ejecutando no recibe información sobre si el comando ha sido o no correctamente ejecutado.
		"S"	Acuse síncrono La ejecución del programa se detiene hasta que el componente receptor haya acusado el comando. La siguiente instrucción se ejecuta tras un acuse en sentido favorable. Si se genera un acuse negativo entonces se emite un aviso de error.

<MarkNr>:	Número de la meta de espera **Nota** En un sistema de varios canales están disponibles como máximo 100 metas de espera (meta de espera 0 ... 99). En un sistema de un canal solo está disponible la meta de espera 0.
1) Para la comunicación específica del usuario o la coordinación de canales también se pueden utilizar metas de espera con SETM/CLEARM sin el comando de espera condicionado WAITMC. Las metas de espera conservan sus valores tras un reset de canal y un arranque de CN.	

Ejemplos

START mediante nombres de canal según MD20000

- Parametrización
```
MD10280 $MN_PROG_FUNCTION_MASK, Bit 1 = TRUE
$MC_CHAN_NAME[ 0 ] = "BEARBEITUNG"  ; Name für Kanal 1
$MC_CHAN_NAME[ 1 ] = "ZUFUEHRUNG"   ; Name für Kanal 2
```
- Programación

Código de programa	Comentarios
START(MECANIZADO)	; Arranque del canal 1
START(ALIMENTACIÓN)	; Arranque del canal 2

START mediante "nombres de canal" locales y variables de usuario

```
Código de programa          Comentarios
DEF INT MÁQUINA = 1         ; Definición variable de usuario para canal 1
DEF INT CARGADOR = 2        ; Definición variable de usuario para canal 2
...
START(MÁQUINA)              ; Arranque del canal 1
START(CARGADOR)             ; Arranque del canal 2
```

START mediante "nombres de canal" locales, variables de usuario y nombres de canal parametrizados

```
Código de programa          Comentarios
DEF INT chanNo1             ; Definición variable de usuario para canal 1
DEF INT chanNo2             ; Definición variable de usuario para canal 2
chanNo1 = CHAN_1            ; Asignación de nombres de canal parametrizados canal 1
chanNo2 = CHAN_2            ; Asignación de nombres de canal parametrizados canal 2
...
START(chanNo1)              ; Arranque del canal 1
START(chanNo2)              ; Arranque del canal 2
```

comando INIT con ruta absoluta

Selección de programa /_N_MPF_DIR/_N_MECANIZ1_MPF en canal 2.

Código de programa

```
INIT(2,"/_N_WKS_DIR/_N_EJE1_WPD/_N_MECANIZ1_MPF")
```

Comando INIT con nombre de programa

Selección del programa con el nombre "MYPROG". El control busca el programa por medio de la ruta de búsqueda.

Código de programa

```
INIT(2,"MYPROG")
```

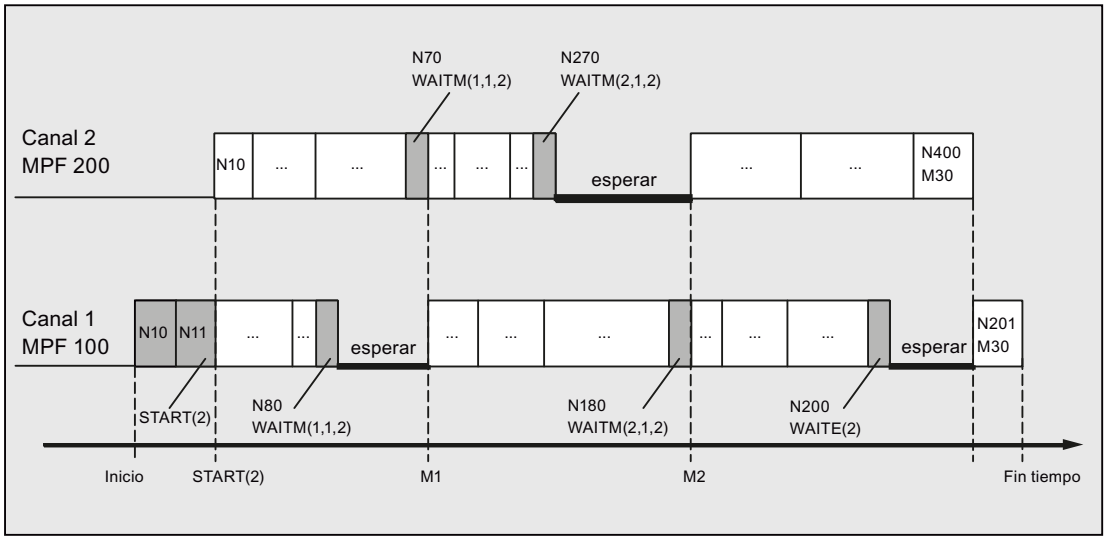
coordinación de programas con WAITM

- Canal 1: el programa /_N_MPF_DIR/_N_MPF100_MPF ya se ha seleccionado e iniciado.

Código de programa	**Comentarios**
	; Programa MPF100
N10 INIT(2,"MPF200","N")	; Selección programa MPF200, canal 2
N11 START(2)	; Arranque del canal 2
...	
N80 WAITM(1,1,2)	; Esperar a meta WAIT 1 en canales 1 y 2
N81 ...	; El canal 1, N81 y el canal 2, N71 se ; inician sincronizados
...	
N180 WAITM(2,1,2)	; Esperar a meta WAIT 2 en canales 1 y 2
N181 ...	; El canal 1, N181 y el canal 2, N271 se ; inician sincronizados
...	
N200 WAITE(2)	; Esperar el fin de programa en canal 2
N201 ...	; N201 no se inicia hasta el fin de programa de ; MPF200 en canal 2
N201 M30	; Fin de programa canal 1

- Canal 2: en el canal 1 se selecciona e inicia el programa MPF200_MPF para el canal 2 a través de las secuencias N10 y N20.

Código de programa	**Comentarios**
;$PATH=/_N_MPF_DIR	; Programa MPF200
...	
N70 WAITM(1,1,2)	Esperar a meta WAIT 1 en canales 1 y 2
N71 ...	; El canal 1, N81 y el canal 2, N71 se ; inician sincronizados
...	
N270 WAITM(2,1,2)	Esperar a meta WAIT 2 en canales 1 y 2
N271 ...	; El canal 1, N181 y el canal 2, N271 se ; inician sincronizados
...	
N400 M30	Fin de programa canal 2



Condiciones

Inicio no sincronizado de la ejecución de las secuencias siguientes después de metas WAIT

Si los canales se coordinan por medio de metas WAIT, puede suceder que la ejecución de las secuencias siguientes no se inicie de forma sincronizada. Esto ocurre cuando, inmediatamente antes de alcanzar la meta WAIT común, en uno de los canales a sincronizar se dispara una acción que provoca en él un borrado de trayecto residual con el consiguiente reposicionamiento (REPOSA) implícito.

Supuesto: asignación de ejes actual en los canales 1 y 2

- Canal 1: ejes X1 y U
- Canal 2: eje X2

Tabla 3-2 Proceso secuencial temporal en los canales 1 y 2

Canal 1	Canal 2	Descripción
...	...	Cualquier ejecución en canales 1 y 2
N100 WAITM(20,1,2)		Canal 1: alcanza la meta WAIT y espera la sincronización con el canal 2
Inicio de la ejecución de GETD(U): • *Intercambio de eje* • *Borrado de trayecto residual* • *REPOSA* *Fin*	N200 GETD(U)	Canal 2: solicita el eje U del canal 1 Canal 1: ejecución de GET(U) en segundo plano
	N210 WAITM(20,1,2)	Canal 2: llega a la marca WAIT. ⇒ De este modo concluye la sincronización de los canales 1 y 2
	N220 G0 X2=100	Canal 2: inicio de la ejecución de N220
N110 G0 X1=100		Canal 1: inicio de la ejecución de N110 con desfase temporal

3.1.9 Técnica de macros (DEFINE ... AS)

ATENCIÓN
La técnica de macros aumenta la complejidad de la programación
En caso de usarse macros, el lenguaje de programación del control puede sufrir modificaciones importantes. El uso de macros requiere la máxima precaución.

Se denomina macro a una recopilación de instrucciones individuales en una nueva instrucción global con una denominación propia. Al llamar la macro en la ejecución del programa, las instrucciones programadas bajo el nombre de la macro se ejecutan sucesivamente.

De acuerdo con el campo de aplicación, es decir, el ámbito en el que la definición de macro es efectiva, existen las siguientes categorías de macros:

- Macros locales
  Las macros locales son macros que están definidas al inicio de un programa CN que no es el programa principal en el momento de la ejecución. Se crean al llamar el programa CN y se borran con un reset al final del programa o bien con el siguiente arranque del control. Solo puede tenerse acceso a las macros locales dentro del programa CN donde están definidas.
- Macros globales de programa
  Las macros globales de programa son macros que están definidas al inicio de un programa CN que se utiliza como programa principal. Se crean al llamar el programa CN y se borran con un reset al final del programa o bien con el siguiente arranque del control. Puede tenerse acceso a las macros globales de programa en el programa principal y en todos los subprogramas.

  Nota

  Disponibilidad de las macros globales de programa

  Las macros globales de programa definidas en el programa principal solo están también disponibles en los subprogramas si está ajustado el siguiente dato de máquina:

  MD11120 $MN_LUD_EXTENDED_SCOPE = 1

  Con MD11120 = 0 las macros globales de programa definidas en el programa principal solo están disponibles en el programa principal.

- Macros globales
  Las macros globales son macros globales del canal o de CN que están definidas en un fichero de definición (fichero de macros) y que se conservan incluso después de un reset al final del programa o tras el siguiente arranque de control. Las macros globales se pueden llamar y ejecutar en cualquier programa principal o subprograma.

  Nota

  Para poder utilizar las macros de un fichero de macros **externo** en el programa CN, el fichero de macros debe cargarse en el CN.

Las macros deben estar definidas antes de su utilización. Se tienen que observar las siguientes reglas:

- En la macro pueden definirse tantos identificadores, funciones G, M, H y nombres de subprograma L como se desee.
- La definición de macros se puede efectuar al inicio del programa o en un fichero de definición propio (fichero de macros).
- Las macros locales y globales de programa se definen al inicio del programa.
- Las macros globales deben definirse en un fichero de macros, p. ej., _N_DEF_DIR/ _N_UMAC_DEF.
- Las macros de comando G solo pueden definirse como macros globales.
- Las funciones H y L deben programarse con 2 dígitos.
- Se pueden programar comandos M y G de 3 dígitos.

Nota

Las palabras y nombres reservados no deben sobrescribirse con macros. Esto también es válido para todos los destinos de salto en un comando GOTO y para las palabras reservadas en bucles de programa, como FOR, WHILE, LOOP o REPEAT.

Sintaxis

Definición de macro:
```
DEFINE <Nombre_macro> AS <Operación_1> <Operación_2> ...
```

Llamada en el programa CN:
```
<Nombre_macro>
```

Descripción

`DEFINE ... AS`:	Combinación de palabras reservadas para definir una macro
`<Nombre_macro>`:	Nombre de la macro Como nombres de macro sólo se admiten identificadores. Con el nombre de la macro ésta se llama desde el programa CN.
`<Operación_1>`:	Primera instrucción de programación en la macro
`<Operación_2>`:	Segunda instrucción de programación en la macro

Ejemplos

Ejemplo 1: definición de macro al inicio del programa

Código de programa	**Comentarios**
`DEFINE LINIE AS G1 G94 F300`	`; Definición de macro`
`...`	

Código de programa	Comentarios
N70 LÍNEA X10 Y20	; Llamada de macro
...	

Ejemplo 2: definiciones de macro en un fichero de macros

Código de programa	Comentarios
DEFINE M6 AS L6	; En el cambio de herramienta se llama a un subprograma que realiza la transferencia de datos necesaria. En el subprograma se emite la función M de cambio de herramienta propiamente dicha (p. ej. M106).
DEFINE G81 AS DRILL(81)	; Emulación del comando G DIN.
DEFINE G33 AS M333 G333	; En el roscado se solicita la sincronización con el PLC. El comando G original G33 ha sido renombrado por DM como G333; la programación permanece idéntica para el usuario.

Ejemplo 3: fichero de macros externo

Después de leer el fichero de macros externo en el control es preciso cargarlo en el CN. Tan sólo después pueden utilizarse las macros en el programa CN.

Código de programa	Comentarios
%_N_UMAC_DEF	
;$PATH=/_N_DEF_DIR	; Macros personalizadas
DEFINE PI AS 3.14	
DEFINE TC1 AS M3 S1000	
DEFINE M13 AS M3 M7	; Cabezal derecho, refrigerante CON
DEFINE M14 AS M4 M7	; Cabezal izquierdo, refrigerante CON
DEFINE M15 AS M5 M9	; Parada del cabezal, refrigerante DES
DEFINE M6 AS L6	; Llamada del programa de cambio de herramienta
DEFINE G80 AS MCALL	; Cancelación ciclo de taladrado
M30	

3.2 Uso de subprogramas

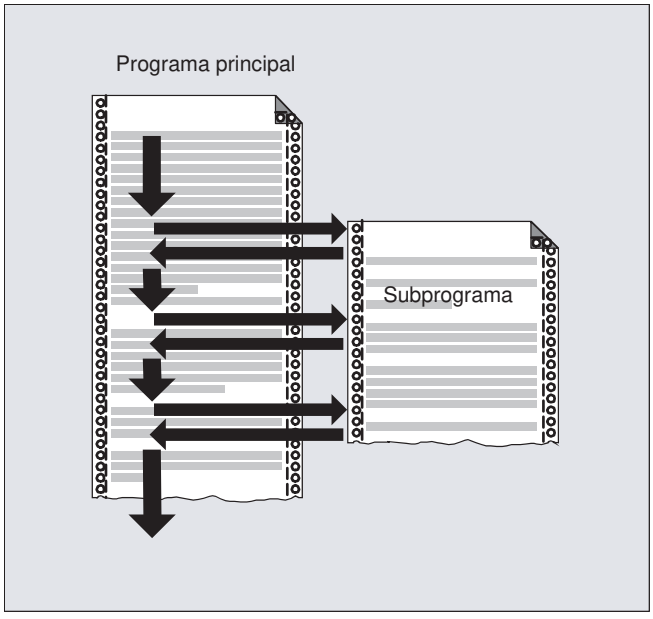
3.2.1 Generalidades

3.2.1.1 Subprograma

El nombre "Subprograma" procede del tiempo en que los programas de pieza estaban divididos de una forma fija en programas principales y subprogramas. Los programas principales eran los programas de pieza que se seleccionaban en el control para la ejecución y luego se iniciaban. Los subprogramas eran programas de pieza que se llamaban desde el programa principal.

Esta división fija ya no existe con el lenguaje CN actual de SINUMERIK. En principio, cualquier programa de pieza puede seleccionarse e iniciarse como programa principal o llamarse como subprograma desde otro programa de pieza.

Con ello, en adelante se denomina subprograma aquel programa de pieza que se llama desde otro programa de pieza.



Aplicación

Como en todos los lenguajes de programación de alto nivel, en el lenguaje CN también se utilizan subprogramas para encapsular secciones de programa que se utilizan varias veces en programas autónomos cerrados en sí mismos.

Los subprogramas ofrecen las siguientes ventajas:

- Mejoran la claridad y legibilidad de los programas.
- Aumentan la calidad reutilizando secciones de programa ya comprobadas.

- Ofrecen la posibilidad de adquirir librerías de proceso específicas.
- Ahorran espacio en memoria.

3.2.1.2 Nombres de subprogramas

Reglas de denominación

El nombre de subprograma puede elegirse libremente de acuerdo con las siguientes reglas:

- Caracteres permitidos:
  - Letras: A ... Z, a ... z
  - Números: 0 ... 9
  - Carácter de subrayado: _
- Los dos primeros caracteres deberían ser dos letras o un carácter de subrayado seguido de una letra.

  Nota

  Si se cumple esta condición, un programa CN puede llamarse como subprograma desde otro programa con solo indicar su nombre. Si, por el contrario, el nombre de programa empieza con números, la llamada de subprograma solo es posible mediante la instrucción CALL.

- Longitud máxima: 24 caracteres

Nota

Mayúsculas/minúsculas

En el lenguaje CN de SINUMERIK **no** se diferencia entre mayúsculas y minúsculas.

Nota

Nombres de programa no permitidos

Para evitar problemas con aplicaciones Windows, **no** deben utilizarse los siguientes nombres de programa:

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

Ampliaciones internas del control

El nombre de subprograma asignado durante la creación del programa se amplía en el interior del control con un prefijo y una extensión:

- Prefijo: `_N_`
- Extensión: `_SPF`

Utilización del nombre de programa

Al utilizar el nombre de programa, p. ej., en la llamada de un subprograma, son posibles todas las combinaciones de prefijos, nombres de programas y extensiones.

Ejemplo:

El subprograma con el nombre `SUB_PROG` puede llamarse con los siguientes identificadores:

1. `SUB_PROG`
2. `_N_SUB_PROG`
3. `SUB_PROG_SPF`
4. `_N_SUB_PROG_SPF`

Programas y subprogramas con el mismo nombre

Si existen un programa principal (.MPF) y un subprograma (.SPF) con el mismo nombre, al utilizar el nombre en el programa CN deberá emplearse la correspondiente extensión de fichero para garantizar la identificación unívoca. De lo contrario, se utilizará el programa que se encuentre con dicho nombre en primer lugar en la ruta de búsqueda.

3.2.1.3 Imbricación de subprogramas

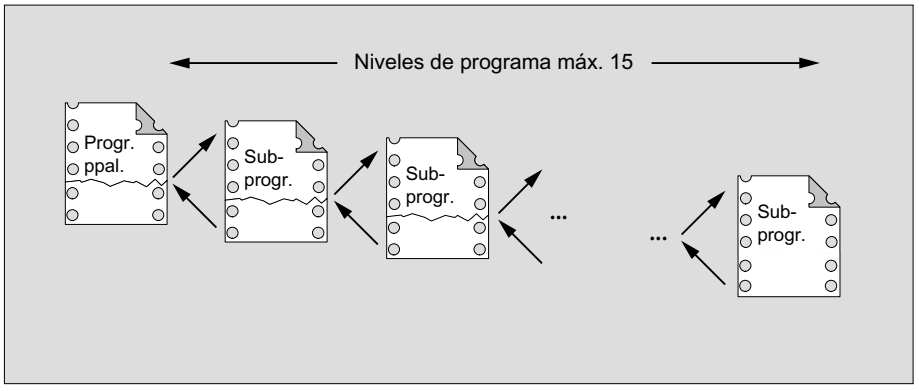
Un programa principal puede llamar a subprogramas que, a su vez, pueden llamar a otros subprogramas. Con ello, las ejecuciones de los programas están imbricadas entre sí. En ese caso, cada programa se ejecuta en un nivel de programa propio.

Profundidad de imbricado

El lenguaje CN ofrece hoy en día 16 niveles de programas. El programa principal siempre se ejecuta en el nivel de programa 0, que es el más alto. Un subprograma se ejecuta siempre en el nivel de programa inmediatamente inferior que sigue a la llamada. Así pues, el nivel de programa 1 es el primer nivel de subprograma.

División de los niveles de programa:

- Nivel de programa 0: nivel de programa principal
- Nivel de programa 1 - 15: nivel de subprograma 1 - 15



Rutinas de interrupción (ASUP)

Si en el contexto de una rutina de interrupción se llama a un subprograma, éste no se ejecuta en el nivel de programa (n) actual activo en el canal, sino también en el nivel de programa inmediatamente inferior (n+1). Para que esto también sea posible en el nivel de programa más bajo, están disponibles 2 niveles de programa adicionales (16 y 17) en combinación con las rutinas de interrupción.

Si se necesitan más de 2 niveles de programa, esto debe tenerse en cuenta explícitamente en la estructuración del programa de pieza ejecutado en el canal. Es decir, el número máximo de niveles de programa que se pueden utilizar es aquél que deje niveles de programa suficientes para la ejecución de rutinas de interrupción.

Si la ejecución de rutinas de interrupción necesita, p. ej. 4 niveles de programa, el programa de pieza debe estructurarse de tal manera que ocupe como máximo el nivel de programa 13. Si se produce entonces una interrupción, ésta tiene a su disposición los 4 niveles de programa necesarios (del 14 al 17).

Ciclos de Siemens

Los ciclos de Siemens necesitan 3 niveles de programa. En consecuencia, un ciclo de Siemens debe llamarse a más tardar en:

- Ejecución del programa de pieza: nivel de programa 12
- Rutina de interrupción: nivel de programa 14

3.2.1.4 Ruta de búsqueda

Al llamar un subprograma sin indicar la ruta, el control busca una cadena de búsqueda predefinida (ver "Ruta de búsqueda en la llamada de subprogramas (Página 575)") en las memorias de programa existentes.

3.2.1.5 Parámetros formales y actuales

Se habla de parámetros formales y actuales en relación con la definición y la llamada de subprogramas con transferencia de parámetros.

Parámetro formal

En la definición de un subprograma, los parámetros que deben transferirse al subprograma, los llamados parámetros formales, deben definirse con tipo y nombre de parámetro.

Por consiguiente, los parámetros formales definen la interfaz del subprograma.

Ejemplo:

Código de programa	**Comentarios**
`PROC CONTORNO (REAL X, REAL Y)`	`; Parámetro formal: X e Y, ambos del tipo REAL`
`N20 X1=X Y1=Y`	`; Desplazamiento del eje X1 a la posición X y del eje Y1 a la posición Y`
`...`	
`N100 RET`	

Parámetro actual

Al llamar a un subprograma, deben transferirse al subprograma valores o variables absolutos, los llamados parámetros actuales.

Así, al efectuar la llamada los parámetros actuales rellenan la interfaz del subprograma con valores actuales.

Ejemplo:

Código de programa	**Comentarios**
N10 DEF REAL ANCHO	; Definición de variables
N20 ANCHO=20.0	; Asignación de variables
N30 CONTORNO(5.5, ANCHO)	; Llamada de subprograma con parámetros actuales: 5.5 y ANCHO
...	
N100 M30	

3.2.1.6 Transferencia de parámetros

Definición de un subprograma con transferencia de parámetros

Un subprograma con transferencia de parámetros se define con la palabra reservada `PROC` y una lista completa de todos los parámetros que espera el subprograma.

Transferencia incompleta de parámetros

Al llamar el subprograma no siempre es necesario transferir explícitamente todos los parámetros definidos en la interfaz del subprograma. Si se omite un parámetro, para este parámetro se transferirá el valor estándar "0".

No obstante, para la identificación unívoca del orden de los parámetros deben indicarse siempre las comas como caracteres separadores de los parámetros. El último parámetro es una excepción. Si éste se omite al efectuar la llamada, también puede omitirse la última coma.

Ejemplo:

Subprograma:

Código de programa	**Comentarios**
PROC SUB_PROG (REAL X, REAL Y, REAL Z)	; Parámetro formal: X, Y y Z
...	
N100 RET	

Programa principal:

Código de programa	**Comentarios**
PROC MAIN_PROG	
...	

Código de programa	Comentarios
N30 SUB_PROG(1.0,2.0,3.0)	; Llamada de subprograma con transferencia de parámetros completa:
	X=1.0, Y=2.0, Z=3.0
…	
N100 M30	

Ejemplos de la llamada de subprograma en N30 con una transferencia de parámetros incompleta:

```
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
```

ATENCIÓN

Transferencia de parámetros Call-By-Reference

Los parámetros transferidos mediante Call-By-Reference no deben omitirse al llamar al subprograma.

ATENCIÓN

Tipo de datos AXIS

Los parámetros del tipo de datos AXIS no deben omitirse al llamar al subprograma.

Comprobación de los parámetros de transferencia

Mediante la variable de sistema $P_SUBPAR [n] con n = 1, 2, ... en el subprograma puede comprobarse si un parámetro se ha transferido explícitamente o si se ha omitido. El índice n se refiere al orden de los parámetros formales. El índice n = 1 se refiere al 1.er parámetro formal; el índice n = 2, al 2.º parámetro formal, etc.

El siguiente segmento de programa muestra, a modo de ejemplo para el 1.er parámetro formal, la manera en que puede realizarse una comprobación:

Programación	Comentarios
PROC SUB_PROG (REAL X, REAL Y, REAL Z)	; Parámetro formal: X, Y y Z
N20 IF $P_SUBPAR[1]==TRUE	; Comprobación del 1.er parámetro formal X.
...	; Estas acciones se ejecutan cuando el parámetro formal X se ha transferido explícitamente.
N40 ELSE	
...	; Estas acciones se ejecutan cuando el parámetro formal X no se ha transferido.
N60 ENDIF	

Programación	Comentarios
...	; Acciones generales
N100 RET	

3.2.2 Definición de un subprograma

3.2.2.1 Subprograma sin transferencia de parámetros

Al definir subprogramas sin transferencia de parámetros, se puede omitir la línea de definición al principio del programa.

Sintaxis

```
[PROC <Nombre de programa>]
...
```

Descripción

PROC:	Instrucción de definición al principio de un programa
<Nombre de programa>:	Nombre del programa

Ejemplo

Ejemplo 1: Subprograma con instrucción PROC

Código de programa	Comentarios
PROC SUB_PROG	; Línea de definición
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; Salto atrás al subprograma

Ejemplo 2: Subprograma sin instrucción PROC

Código de programa	Comentarios
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; Salto atrás al subprograma

Consulte también

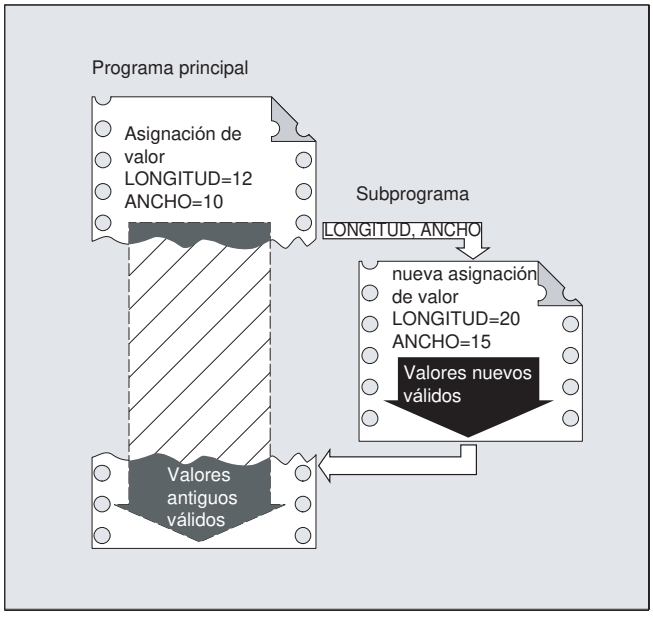
Llamada de subprograma sin transferencia de parámetros (Página 537)

3.2.2.2 Subprograma con transferencia de parámetros Call-By-Value (PROC)

Un subprograma con transferencia de parámetros Call-By-Value se define con la palabra reservada `PROC`, seguida del nombre del programa y una lista completa de todos los parámetros con tipo y nombre. La instrucción de definición debe estar en la primera línea de programa.

Call-by-Value

En una transferencia de parámetros Call-by-Value, el programa desde el que se realiza la llamada transfiere solo el valor de una variable al subprograma. De esta manera, el subprograma no obtiene acceso directo a la variable. En consecuencia, si se modifica el valor de parámetro, solo se modifica el valor visible en el subprograma. El valor de las variables definidas en el programa desde el que se realiza la llamada no varía. Por lo tanto, la transferencia de parámetros Call-By-Value no tiene repercusiones sobre el programa desde el que se realiza la llamada.



Sintaxis

```
PROC <Nombre de programa> (<Tipo de parámetro> <Nombre de
parámetro>=<Valor_inic>, ...)
```

Nota

Se pueden transferir hasta 127 parámetros.

Descripción

`PROC`:	Instrucción de definición al principio de un programa
`<Nombre de programa>`:	Nombre del programa
`<Tipo de parámetro>`:	Tipo de datos del parámetro (p. ej. REAL, INT, BOOL)

<Nombre del parámetro>:	Nombre del parámetro
<Valor_inic>:	Valor opcional para inicializar el parámetro (**opcional**) Si no se indica ningún parámetro al llamar el subprograma, se asigna al parámetro el valor de inicialización.

Ejemplos

Ejemplo 1

Definición de un subprograma SUB_PROG con tres parámetros de tipo REAL con valores por defecto:

Código de programa

```
PROC SUB_PROG(REAL LENGTH=10.0, REAL WIDTH=20.0, REAL HIGHT=30.0)
```

Ejemplo 2

Diferentes variantes de llamada

Código de programa

```
PROC MAIN_PROG
  REAL PAR_1 = 100
  REAL PAR_2 = 200
  REAL PAR_3 = 300
  ; variantes de llamada
  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
```

Consulte también

Llamada de subprograma con transferencia de parámetros (EXTERN) (Página 539)

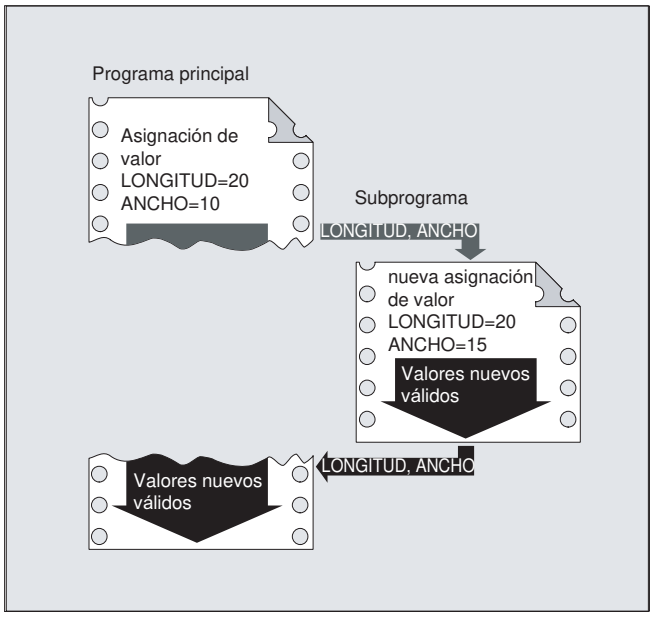
3.2.2.3 Subprograma con transferencia de parámetros Call-By-Reference (PROC, VAR)

Un subprograma con transferencia de parámetros Call-By-Reference se define con la palabra reservada `PROC`, seguida del nombre del programa y una lista completa de todos los parámetros con la palabra reservada `VAR`, tipo y nombre. La instrucción de definición debe estar en la primera línea de programa. Como parámetros también pueden transferirse referencias a matrices.

Call-by-Reference

En una transferencia de parámetros Call-by-Reference, el programa desde el que se realiza la llamada no transfiere al subprograma el valor de una variable, sino una referencia (puntero) a la variable. De esta manera, el subprograma obtiene acceso directo a la variable. En consecuencia, si se modifica el valor de parámetro, no solo se modifica el valor visible en el

subprograma, sino el valor de las variables definidas en el programa desde el que se realiza la llamada. La transferencia de parámetros Call-By-Reference también tiene entonces repercusiones sobre el programa desde el que se realiza la llamada también después de finalizar el subprograma.



Nota

La transferencia de parámetros Call-By-Reference solo será necesaria si la variable transferida está definida localmente en un programa desde el que se realiza la llamada (LUD). No es necesario transferir variables globales de canal y globales de CN, ya que a estas variables también se puede acceder directamente desde el subprograma.

Sintaxis

```
PROC <Nombre de programa> (VAR <Tipo de parámetro> <Nombre de
parámetro>, ...)
PROC <Nombre de programa> (VAR <Tipo de matriz> <Nombre de matriz>
[<m>,<n>,<o>], ...)
```

Nota

Se pueden transferir hasta 127 parámetros.

Descripción

PROC:	Instrucción de definición al principio de un programa
VAR:	Palabra reservada para la transferencia de parámetros por referencia
<Nombre de programa>:	Nombre del programa
<Tipo de parámetro>:	Tipo de datos del parámetro (p. ej. REAL, INT, BOOL)

<Nombre del parámetro>:	Nombre del parámetro	
<Tipo de matriz>:	Tipo de datos de los elementos de matriz (p. ej. REAL, INT, BOOL)	
<Nombre de matriz>:	Nombre de la matriz	
[<m>,<n>,<o>]:	Tamaño de matriz Actualmente se permiten matrices de 3 dimensiones como máximo:	
	<m>:	Tamaño de la matriz para la 1.ª dimensión
	<n>:	Tamaño de la matriz para la 2.ª dimensión
	<o>:	Tamaño de la matriz para la 3.ª dimensión

Nota

- El nombre de programa indicado después de la palabra reservada `PROC` debe coincidir con el nombre de programa asignado a la interfaz de usuario.
- Con matrices de longitud indefinida como parámetros formales pueden ejecutarse matrices de subprogramas de longitud variable. Para ello no se indica la longitud de la 1.ª dimensión como parámetro formal en la definición, p. ej., de una matriz bidimensional. Sin embargo, es preciso escribir la coma.
  Ejemplo: `PROC <Nombre de programa> (VAR REAL MATRIZ[,5])`

Ejemplo

Definición de un subprograma con dos parámetros como referencia al tipo REAL:

Código de programa

```
; Parámetro 1: Referencia al tipo: REAL, nombre: LONGITUD
; parámetro 2: Referencia al tipo: REAL, nombre: ANCHURA
PROC SUB_PROG(VAR REAL LONGITUD, VAR REAL ANCHURA)
```

Consulte también

Llamada de subprograma con transferencia de parámetros (EXTERN) (Página 539)

3.2.2.4 Almacenamiento de las funciones G modales (SAVE)

El atributo SAVE hace que se guarden los comandos G modales activos antes de llamar al subprograma y se reactiven de nuevo tras finalizar el subprograma.

ATENCIÓN

Interrupción del modo de contorneado

Si, estando activo el modo de contorneado, se llama a un subprograma con el atributo SAVE, se interrumpe el modo de contorneado al final del subprograma (salto atrás).

Sintaxis

```
PROC <Nombre del subprograma> SAVE
```

Descripción

SAVE:	Almacenamiento de los comandos G modales antes de llamar al subprograma y restablecimiento tras finalizar el subprograma

Ejemplo

En el subprograma CONTORNO está activo el comando G modal G91 (acotado incremental). En el programa principal está activo el comando G modal G90 (acotado absoluto). Mediante la definición de subprograma con SAVE actúa de nuevo G90 tras finalizar el subprograma en el programa principal.

Definición de subprograma:

Código de programa	Comentarios
PROC CONTORNO (VALOR REAL1) SAVE	; Definición de subprograma con parámetro SAVE
N10 G91 ...	; Comando G modal G91: Acotado incremental
N100 M17	; Fin de subprograma

Programa principal:

Código de programa	Comentarios
N10 G0 X... Y... G90	; Comando G modal G90: Acotado absoluto
N20 ...	
...	
N50 CONTORNO (12.4)	; Llamada a un subprograma
N60 X... Y...	; Comando G modal G90 reactivado mediante SAVE

Condiciones

Frames

El comportamiento de frames en relación con subprogramas con el atributo SAVE depende del tipo de frame y puede ajustarse mediante los datos de máquina.

3.2.2.5 Suprimir la ejecución de secuencia individual (SBLOF, SBLON)

Con la ejecución de secuencia individual activa, el usuario también tiene la posibilidad de ejecutar un programa CN total o parcialmente sin efectuar ninguna parada. La ejecución de secuencia individual se suprime con el comando SBLOF y la reactivación se efectúa con el comando SBLON.

Supresión de la ejecución de secuencia individual para todo el programa CN

Si se ha programado la desactivación de la ejecución de secuencia individual (SBLOF) en la primera línea (PROC...) de un **programa principal**, esto se aplica hasta que el programa CN finalice o se interrumpa. A continuación, en caso de secuencia individual, el programa CN se ejecutará sin detenerse.

Si se ha programado la desactivación de la ejecución de secuencia individual (SBLOF) en la primera línea (PROC...) de un **subprograma**, esto se aplica hasta que el programa CN finalice

o se interrumpa. Con el comando de salto atrás programado se decide si se parará al final de un subprograma o no:

- Salto atrás con M17: Parada al final del subprograma
- Salto atrás con RET: **Sin** parada al final del subprograma

Supresión de la ejecución de secuencia individual dentro del programa CN

Si se ha programado la desactivación de la ejecución de secuencia individual (SBLOF) en una secuencia dentro de un programa CN, se desactiva la ejecución de secuencia individual a partir de esa secuencia hasta la siguiente activación programada de la ejecución de secuencia individual (SBLON) o el fin del nivel de subprograma activo.

Sintaxis

Supresión de la ejecución de secuencia individual para todo el programa CN:

```
PROC ... SBLOF
...
```

Supresión de la ejecución de secuencia individual dentro del programa CN:

```
...
SBLOF
...
SBLON
...
```

Descripción

PROC:	Primera instrucción de un programa	
SBLOF:	Procedimiento predefinido para desactivar la ejecución de secuencia individual	
	Único elemento de la secuencia:	Sí, es posible en la secuencia PROC
	Efecto:	modal
SBLON:	Procedimiento predefinido para activar la ejecución de secuencia individual	
	Único elemento de la secuencia:	Sí
	Efecto:	modal

Particularidades

- **Visualización de secuencia en caso de ejecución de secuencia individual suprimida**
  La visualización de secuencia actual puede suprimirse en subprogramas con DISPLOF. Si se programa DISPLOF junto con SBLOF, al efectuar paradas de secuencia individual dentro del subprograma, se mostrará la llamada del subprograma.
- **Supresión de la ejecución de secuencia individual en subprogramas asíncronos (ASUP)**
  Para ejecutar un ASUP con la ejecución de secuencia individual activa en un paso, debe programarse una instrucción PROC con SBLOF en el ASUP. Lo mismo se aplica para la función "ASUP editable de sistema" (MD11610 $MN_ASUP_EDITABLE).
  Si la parada de secuencia individual se suprime en el ASUP de sistema o usuario programando SBLOF en la línea de PROC o mediante los ajustes del dato de máquina MD10702 $MN_IGNORE_SINGLEBLOCK_MASK (bit 0 = 1 o bit 1 = 1), esta puede activarse de nuevo programando SBLON en el ASUP.
  Si la parada de secuencia individual se suprime en el ASUP de usuario mediante el ajuste del dato de máquina MD20117 $MC_IGNORE_SINGLEBLOCK_ASUP, **no** puede activarse de nuevo programando SBLON en el ASUP.
- **Particularidades en diferentes tipos de ejecución de secuencia individual**
  - "SB2: Secuencia de cálculo" y MD10702 $MN_IGNORE_SINGLEBLOCK_MASK, bit 12 = 1:
    → No se detiene en la secuencia SBLON.
  - "SB3: Secuencia a secuencia fino":
    → Se suprime el comando SBLOF.
- **Supresión de la ejecución de secuencia individual en programas imbricados**
  Si en un subprograma se ha programado SBLOF en la instrucción PROC, se parará en el salto atrás del subprograma con M17. De este modo se evita que ya se ejecute la siguiente secuencia en el programa desde el cual se efectúa la llamada. Si en un subprograma se activa una supresión de la ejecución de secuencia individual con SBLOF, sin SBLOF en la instrucción PROC, se parará solo después de la siguiente secuencia de función de máquina del programa desde el que se efectúa la llamada. Si esto no se desea, es necesario programar de nuevo SBLON en el subprograma antes del salto atrás (M17). En caso de un salto atrás con RET a un programa de nivel superior, no se efectuará ninguna parada.

Ejemplos

Ejemplo 1: Supresión de la ejecución de secuencia individual dentro del programa CN

Situación inicial: la ejecución de secuencia individual está activa.

Código de programa	Comentarios
N10 G1 X100 F1000	
N20 **SBLOF**	; Desactivación de la ejecución de secuencia individual
N30 Y20	
N40 M100	
N50 R10=90	
N60 **SBLON**	; Reactivación de la ejecución de secuencia individual
N70 M110	
N80 ...	

El área entre N20 y N60 se ejecuta en el modo secuencia a secuencia como un solo paso.

Ejemplo 2: el ciclo tiene que actuar para el usuario como un comando

Situación inicial: la ejecución de secuencia individual está activa.

Programa principal:

Código de programa
...
N100 G1 X10 G90 F200
N120 X-4 Y6
N130 **CYCLE1**
N140 G1 X0
N150 M30

Ciclo CYCLE1:

Código de programa	**Comentarios**
N100 **PROC** CYCLE1 DISPLOF **SBLOF**	; Supresión de la ejecución de secuencia individual para todo el programa.
N110 R10=3*SIN(R20)+5	
N120 IF (R11 <= 0)	
N130 SETAL(61000)	
N140 ENDIF	
N150 G1 G91 Z=R10 F=R11	
N160 M17	

El ciclo CYCLE1 se procesa con la ejecución de secuencia individual activa, es decir, para la ejecución del ciclo CYCLE1 debe pulsarse una vez la tecla de arranque.

Ejemplo 3: un ASUP iniciado por el PLC para activar el decalaje de origen y correcciones de herramienta modificados no debe ser visible

Código de programa	**Comentarios**
N100 **PROC** NV **SBLOF DISPLOF**	; Supresión de la ejecución de secuencia individual y visualización de secuencia.
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
N200 END: D=$P_TOOL T=$P_TOOLNO
N210 RET
```

Ejemplo 4: parada selectiva en subprograma

Situación inicial:

- La ejecución de secuencia individual está activa.
- MD10702 $MN_IGNORE_SINGLEBLOCK_MASK, bit 12 = 1

Programa principal:

```
N10 G0 X0                ; Parada de secuencia individual
N20 X10                  ; Parada de secuencia individual
N30 CYCLE                ; Secuencia de desplazamiento generada por el ciclo.
N50 G90 X20              ; Parada de secuencia individual
M30
```

Ciclo CYCLE:

```
PROC CYCLE SBLOF         ; Supresión de la ejecución de secuencia individual
N100 R0=1
N110 SBLON               ; Sin parada de secuencia individual debido a MD10702
                         bit 12 = 1
N120 X1                  ; Parada de secuencia individual
N140 SBLOF
N150 R0=2
RET
```

Ejemplo 5: supresión de la ejecución de secuencia individual con imbricación del programa

Situación inicial:

- La ejecución de secuencia individual de tipo 2 está activa.
- En la secuencia SBLON no debe efectuarse ninguna parada (MD10702 $MN_IGNORE_SINGLEBLOCK_MASK, bit 12 = 1)

```
N10 X0 F1000                          ; Parada de secuencia individual
N20 UP1(0)
      PROC UP1(INT _NR) SBLOF         ; Desactivación de la ejecución de se-
                                      cuencia individual para UP1
      N100 X10
      N110 UP2(0)
            PROC UP2(INT _NR)
```

```
Código de programa                          Comentarios
            N200 X20
            N210 SBLON                      ; Activación de la ejecución de secuen-
                                              cia individual
            N220 X22                        ; Parada de secuencia individual
            N230 UP3(0)
                  PROC UP3(INT _NR)
                  N300 SBLOF                ; Desactivación de la ejecución de se-
                                              cuencia individual
                  N305 X30
                  N310 SBLON                ; Activación de la ejecución de secuen-
                                              cia individual
                  N320 X32                  ; Parada de secuencia individual
                  N330 SBLOF                ; Desactivación de la ejecución de se-
                                              cuencia individual
                  N340 X34
                  N350 M17                  ; Parada de secuencia individual (M17)
            N240 X24                        ; Parada de secuencia individual (N210)
            N250 M17                        ; Parada de secuencia individual (M17)
      N120 X12
      N130 M17                              ; Parada de secuencia individual (M17)
N30 X0                                      ; Parada de secuencia individual
N40 M30                                     ; Parada de secuencia individual
```

3.2.2.6 Suprimir la visualización de secuencia actual (DISPLOF, DISPLON, ACTBLOCNO)

La secuencia de programa actual se muestra de manera estándar en la visualización de secuencia. En ciclos o subprogramas puede suprimirse la visualización de la secuencia actual como el comando `DISPLOF`. En vez de la secuencia actual se visualizará en ese caso la llamada del ciclo o del subprograma. Con el comando `DISPLON` puede volverse a eliminar la supresión de la visualización de secuencia.

`DISPLOF` o `DISPLON` se programa en la línea de programa con la instrucción `PROC` y actúa para el subprograma entero e implícitamente para todos los subprogramas a los que llama este subprograma y que no contienen el comando `DISPLON` o `DISPLOF`. Este comportamiento también se da para los ASUP.

Sintaxis

```
PROC … DISPLOF
PROC … DISPLOF ACTBLOCNO
PROC … DISPLON
```

Descripción

DISPLOF:	Comando para suprimir la visualización de secuencia actual.	
	Colocación:	Al final de la línea del programa con la instrucción PROC
	Eficacia:	hasta el salto atrás desde el subprograma o el final del programa.
	Nota: Si desde el subprograma se llaman otros subprogramas con el comando DISPLOF, también se suprimirá en ellos la visualización de secuencia actual siempre que ahí no esté explícitamente programado DISPLON.	
DISPLON:	Comando para anular la supresión de la visualización de secuencia actual	
	Colocación:	Al final de la línea del programa con la instrucción PROC
	Eficacia:	hasta el salto atrás desde el subprograma o el final del programa.
	Nota: Si desde el subprograma se llaman otros subprogramas con el comando DISPLON, también se visualizará en ellos la secuencia actual de programa siempre que ahí no esté explícitamente programado DISPLOF.	
ACTBLOCNO:	DISPLOF junto con el atributo ACTBLOCNO hace que, en caso de alarma, se emita el número de la secuencia actual en la que se ha producido la alarma. Esto también se aplica cuando en un nivel de programa inferior sólo está programado DISPLOF. Con DISPLOF sin ACTBLOCNO, por el contrario, se muestra el número de secuencia de la llamada a ciclo o subprograma del último nivel de programa no identificado con DISPLOF.	

Ejemplos

Ejemplo 1: suprimir la visualización de secuencia actual en el ciclo

Código de programa	**Comentarios**
PROC CYCLE (AXIS TOMOV, REAL POSITION) SAVE DISPLOF	; Suprimir la visualización de secuencia actual. En su lugar, se debe mostrar la llamada de ciclo, p. ej.: CYCLE(X,100.0)
DEF REAL DIFF	; Contenido del ciclo
G01 ...	
...	
RET	; Salto atrás a subprograma. En la visualización de secuencia se muestra la secuencia que sigue a la llamada de ciclo.

Ejemplo 2: visualización de secuencia con emisión de alarma

Subprograma SUBPROG1 (con ACTBLOCNO):

Código de programa	**Comentarios**
PROC SUBPROG1 DISPLOF ACTBLOC- NO	
N8000 R10 = R33 + R44	
...	

Código de programa	**Comentarios**
N9040 R10 = 66 X100	; Disparar alarma 12080
...	
N10000 M17	

Subprograma SUBPROG2 (sin ACTBLOCNO):

Código de programa	**Comentarios**
PROC SUBPROG2 DISPLOF	
N5000 R10 = R33 + R44	
...	
N6040 R10 = 66 X100	; Disparar alarma 12080
...	
N7000 M17	

Programa principal:

Código de programa	**Comentarios**
N1000 G0 X0 Y0 Z0	
N1010 ...	
...	
N2050 SUBPROG1	; Emisión de alarma = "12080 canal K1 secuencia N9040 error de sintaxis en texto R10="
N2060 ...	
N2350 SUBPROG2	; Emisión de alarma = "12080 canal K1 secuencia N2350 error de sintaxis en texto R10="
...	
N3000 M30	

Ejemplo 3: anular la supresión de la visualización de secuencia actual

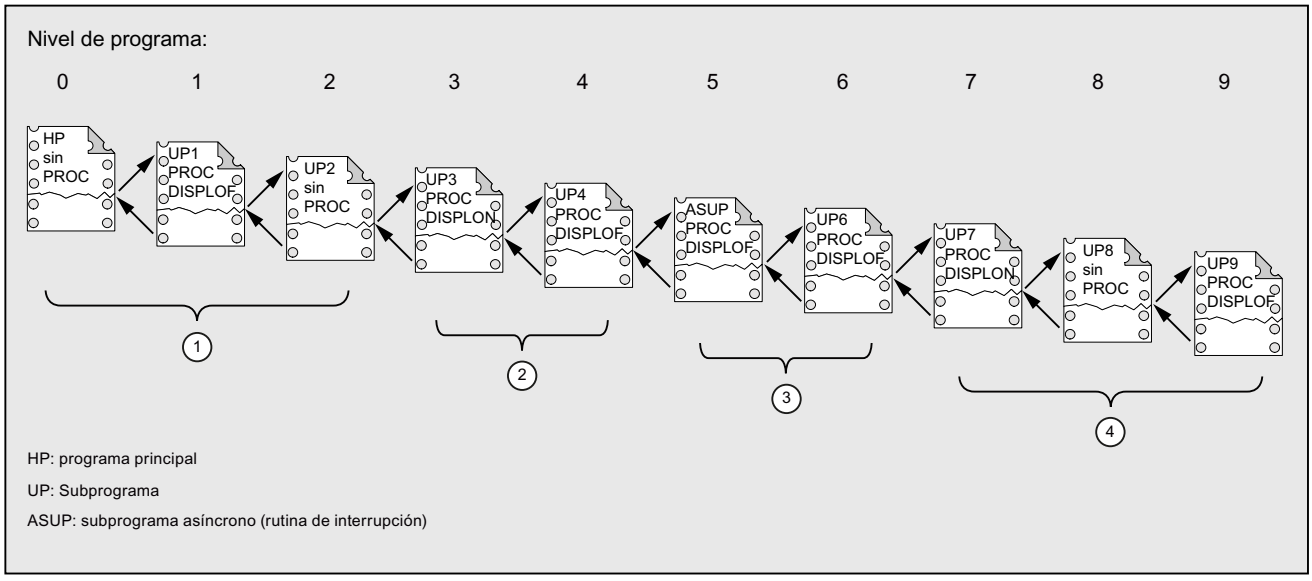
Subprograma SUB1 con supresión:

Código de programa	**Comentarios**
PROC SUB1 DISPLOF	; Suprimir la visualización de secuencia actual en el subprograma SUB1. En lugar de ello, la secuencia debe visualizarse con la llamada SUB1.
...	
N300 SUB2	; Llamar a subprograma SUB2.
...	
N500 M17	

Subprograma SUB2 sin supresión:

Código de programa	Comentarios
`PROC SUB2 DISPLON`	`; Anular la supresión de la visualización de secuencia actual en el subprograma SUB2.`
`...`	
`N200 M17`	`; Salto atrás al subprograma SUB1. En SUB1 se vuelve a suprimir la visualización de secuencia actual.`

Ejemplo 4: comportamiento de la visualización con diferentes combinaciones DISPLON/ DISPLOF



① En la visualización de secuencia actual, se muestran las líneas de programa de pieza desde el nivel de programa 0.

② En la visualización de secuencia actual, se muestran las líneas de programa de pieza desde el nivel de programa 3.

③ En la visualización de secuencia actual, se muestran las líneas de programa de pieza desde el nivel de programa 3.

④ En la visualización de secuencia actual, se muestran las líneas de programa de pieza desde el nivel de programa 7/8.

3.2.2.7 Identificación de subprogramas con preparación (PREPRO)

Con la palabra reservada `PREPRO` se pueden identificar todos los ficheros en el arranque al final de la línea de instrucciones `PROC`.

Nota

Este tipo de preparación de programa depende del ajuste de dato de máquina MD10700 $MN_PREPROCESSING_LEVEL. Tener en cuenta las indicaciones del fabricante de la máquina.

Sintaxis

```
PROC … PREPRO
```

Descripción

PREPRO:	Palabra reservada para identificar todos los ficheros preparados en el arranque de los programas CN que se encuentran en directorios de ciclos

Lectura de subprogramas con preparación y llamada de subprogramas

Tanto en el arranque de subprogramas preparados con parámetros como en la llamada de subprogramas, los directorios de ciclos se tratan con el mismo orden:

1. _N_CUS_DIR ciclos de usuario
2. _N_CMA_DIR ciclos de fabricante
3. _N_CST_DIR ciclos estándar

En el caso de programas CN con el mismo nombre y distinta versión, se activa la instrucción PROC encontrada en primer lugar y se pasa por alto la otra instrucción PROC sin producirse un aviso de alarma.

3.2.2.8 Salto atrás al subprograma M17

Al final de un subprograma se encuentra el comando de salto atrás M17 (o bien el comando de fin del programa de pieza M30). Provoca el salto atrás al programa desde el que se realiza la llamada en la secuencia del programa de pieza después de la llamada del subprograma.

Nota

M17 y M30 se tratan de una forma equivalente en el lenguaje CN.

Sintaxis

```
PROC <Nombre de programa>
...
M17/M30
```

Condiciones

Efecto del salto atrás al subprograma en el modo de contorneado

Si M17 (o M30) se encuentra solo en la secuencia del programa de pieza, con él se interrumpe un modo de contorneado activo en el canal.

Para impedir que el modo de contorneado se interrumpa, M17 (o M30) debe escribirse también en la última secuencia de desplazamiento. Adicionalmente debe estar ajustado a "0" el siguiente dato de máquina:

DM20800 $MC_SPF_END_TO_VDI = 0 (ninguna salida M30/M17 a la interfaz CN/PLC)

Ejemplo

1. Subprograma con `M17` en la secuencia propia

Código de programa	Comentarios
`N10 G64 F2000 G91 X10 Y10`	
`N20 X10 Z10`	
`N30 M17`	`; Salto atrás con interrupción del modo de contorneado.`

2. Subprograma con `M17` en la última secuencia de desplazamiento

Código de programa	Comentarios
`N10 G64 F2000 G91 X10 Y10`	
`N20 X10 Z10 M17`	`; Salto atrás sin interrupción del modo de contorneado.`

3.2.2.9 Salto atrás al subprograma RET

En lugar del comando de salto atrás `M17`, en el subprograma se puede utilizar también el comando `RET`. `RET` se debe programar en una secuencia de programa de pieza propia. Al igual que `M17`, `RET` provoca el salto atrás al programa desde el que se realiza la llamada en la secuencia del programa de pieza después de la llamada del subprograma.

Nota

Mediante la programación de parámetros se puede modificar el comportamiento del salto atrás de `RET` (ver "Salto atrás parametrizable al subprograma (RET ...) (Página 527)").

Aplicación

La instrucción `RET` debe utilizarse cuando un modo de contorneado G64 (`G641` ... `G645`) no deba interrumpirse por el salto atrás.

Requisitos

El comando `RET` únicamente puede utilizarse en subprogramas que no se hayan definido con el atributo `SAVE`.

Sintaxis

```
PROC <Nombre de programa>
...
RET
```

Ejemplo

Programa principal:

Código de programa	Comentarios
PROC MAIN_PROGRAM	; Inicio del programa
...	
N50 SUB_PROG	; Llamada a un subprograma: SUB_PROG
N60 ...	
...	
N100 M30	; Fin del programa

Subprograma:

Código de programa	Comentarios
PROC SUB_PROG	
...	
N100 RET	; El salto atrás se produce en la secuencia N60 en el programa principal.

3.2.2.10 Salto atrás parametrizable al subprograma (RET ...)

Por lo general, con el comando `RET` se regresa desde un subprograma al programa desde el que se ha llamado. La ejecución se continúa con la línea de programa que sigue a la llamada al subprograma. Si la ejecución del programa debe continuar en otro punto, existen las siguientes posibilidades:

- Continuación de la ejecución del programa tras llamar a los ciclos de desbaste en modo de dialecto ISO (después de la descripción de contorno).
- Salto atrás al programa principal desde cualquier nivel de subprograma (también después de ASUP) en el tratamiento de errores.
- Salto atrás de varios niveles de programa para aplicaciones especiales en ciclos de compilación y en el modo de dialecto ISO.

A tal efecto es necesario programar el comando `RET` con más parámetros.

Sentido de búsqueda

Al indicar el parámetro `<Secuencia meta>`, el salto atrás no se produce hasta la secuencia posterior a la secuencia de llamada. Seguidamente se busca la meta hacia el **final** del programa al que se regresa. Si la búsqueda no tiene éxito, a continuación se busca hacia el inicio del programa.

Sintaxis

```
RET("<Secuencia meta>")
RET("<Secuencia meta>", <Secuencia tras secuencia meta>)
RET("<Secuencia meta>"),
<Secuencia tras secuencia meta>,<Cantidad niveles salto atrás>)
RET("<Secuencia meta>",<Cantidad niveles salto atrás>)
```

```
RET("<Secuencia meta>"), <Secuencia tras secuencia
meta>,<Cantidad niveles salto atrás>,
<Salto atrás hasta inicio programa>)
RET( , ,<Cantidad niveles salto atrás>,<Salto atrás hasta inicio pro
grama>)
```

Descripción

RET:	Fin de subprograma		
<Secuencia meta>:	En el parámetro se indica como destino del salto la secuencia en la que debe continuar la ejecución del programa. Si no está programado el parámetro <Cantidad niveles salto atrás>, el destino del salto se encuentra en el programa desde el que se ha llamado al subprograma actual. Posibles indicaciones:		
	<Número de secuencia>	Número de la secuencia meta. La búsqueda del número de secuencia se realiza en el programa al que se regresa y en primer lugar hacia el final del programa.	
	<Marca de salto>	Marca de salto que debe estar presente en el programa al que se regresa. La búsqueda de la marca de salto se realiza en el programa al que se regresa y en primer lugar hacia el final del programa.	
	<Cadena de caracteres>	Cadena de caracteres que debe estar presente en el programa al que se regresa (p. ej., nombre de programa o de variable). La búsqueda de la cadena de caracteres se realiza en el programa al que se regresa y en primer lugar hacia el final del programa. Para la programación de la cadena de caracteres en la secuencia meta, son válidas las siguientes reglas: • **Espacio al final** (a diferencia de la marca de salto, que se identifica por un ":" al final). • **Antes de la cadena de caracteres** solo deben fijarse un número de secuencia y/o una marca de salto, pero **ningún comando de programa**.	
<Secuencia tras secuencia meta>:	En este parámetro se indica si la ejecución del programa debe continuar en la secuencia indicada en el parámetro <Secuencia meta> o en la secuencia siguiente a esta.		
	Tipo:	INT	
	Valor:	0	El salto atrás se produce hasta la secuencia indicada en el parámetro <Secuencia meta>.
		> 0	El salto atrás se produce hasta la secuencia posterior a la indicada en el parámetro <Secuencia meta>.

<Número de niveles de salto atrás>:	En este parámetro se indica el número de niveles de programa que deben saltarse hacia atrás para buscar allí la secuencia meta y reanudar la ejecución del programa.		
	Tipo:	INT	
	Valor:	1	El programa continúa en el "nivel de programa actual -1" (como RET sin parámetros).
		2	El programa continúa en el "nivel de programa actual -2", es decir, se salta un nivel hacia atrás.
		3	El programa continúa en el "nivel de programa actual -3", es decir, se saltan dos niveles hacia atrás.
		...	
	Rango de valores:	1 ... 15	
<Salto atrás hasta inicio de programa>:	En este parámetro se indica si el programa debe reanudarse desde el principio en caso de salto atrás al programa principal estando activo allí el **modo de dialecto ISO**.		
	Tipo:	BOOL	
	Valor:	1	Si el salto atrás tiene lugar en el programa principal y allí hay activo un **modo de dialecto ISO**, se deriva al inicio del programa.

Nota

En el caso de salto atrás al subprograma con una cadena de caracteres como indicación para la búsqueda de secuencia meta, se busca siempre primero una marca de salto en el programa desde el que se llama.

Si debe definirse unívocamente un destino de salto con una cadena de caracteres, ésta no debe coincidir con el nombre de una marca de salto ya que, de lo contrario, el salto atrás al subprograma se ejecutará siempre en la marca de salto y no en la cadena de caracteres (ver ejemplo 2).

Condiciones

Al saltar atrás varios niveles de programa se evalúan las instrucciones SAVE de los distintos niveles de programa.

Si en un salto atrás de varios niveles de programa hay activo un subprograma modal y se ha programado en uno de los subprogramas saltados el comando de cancelación MCALL para el subprograma modal, éste seguirá activo.

ATENCIÓN
Error de programación
Al saltar atrás varios niveles de programa, la responsabilidad de que se continúe con los ajustes modales correctos recae exclusivamente en el usuario. Esto puede lograrse, p. ej., programando una secuencia principal adecuada.

Ejemplos

Ejemplo 1: reposicionamiento en el programa principal tras un procesamiento ASUP

```
Programación                          Comentarios
N10010 CALL "UP1"                     ; Nivel de programa 0 (programa principal)
  N11000 PROC UP1                     ; Nivel de programa 1
  N11010 CALL "UP2"
  N12000 PROC UP2                     ; Nivel de programa 2
  ...
  N19000 PROC ASUP                    ; Nivel de programa 3 (procesamiento ASUP)
  ...
  N19100 RET("N10900", ,$P_STACK)     ; Salto atrás del subprograma al programa prin-
                                      cipal
                                      ; $P_STACK: nivel de programa actual
N10900                                ; Secuencia meta en el programa principal
N10910 MCALL                          ; Desactivar llamada de subprograma modal
N10920 G0 G60 G40 M5                  ; Inicializar otros ajustes modales
```

Ejemplo 2: cadena de caracteres (<String>) como indicación para la búsqueda de secuencia meta

Programa principal:

```
Código de programa                    Comentarios
PROC MAIN_PROGRAM
N1000 DEF INT iVar1=1, iVar2=4
N1010 ...
N1200 subProg1                        ; Llamada del subprograma "subProg1"
N1210 M2 S1000 X10 F1000
N1220 ......
N1400 subProg2                        ; Llamada del subprograma "subProg2"
N1410 M3 S500 Y20
N1420 ..
N1500 lab1: iVar1=R10*44
N1510 F500 X5
N1520 ...
N1550 subprog1: G1 X30                ; "subProg1" está definido aquí como marca de
                                      salto.
N1560 ...
N1600 subProg3                        ; Llamada del subprograma "subProg3"
N1610 ...
N1900 M30
```

Subprograma subProg1:

Código de programa	**Comentarios**
PROC subProg1	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg2")	; Salto atrás al programa principal, secuencia N1400

Subprograma subProg2:

Código de programa	**Comentarios**
PROC subProg2	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("iVar1")	; Salto atrás al programa principal, secuencia N1500

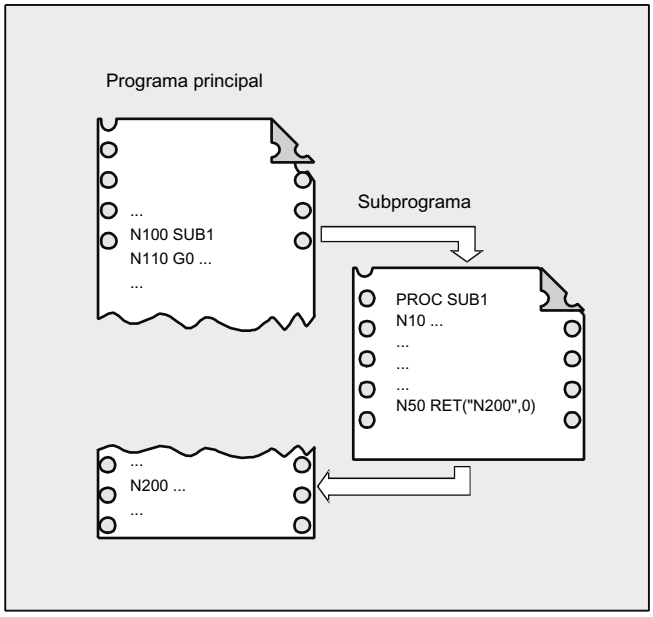
Subprograma subProg3:

Código de programa	**Comentarios**
PROC subProg3	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg1")	; Salto atrás al programa principal, secuencia N1550

Información adicional

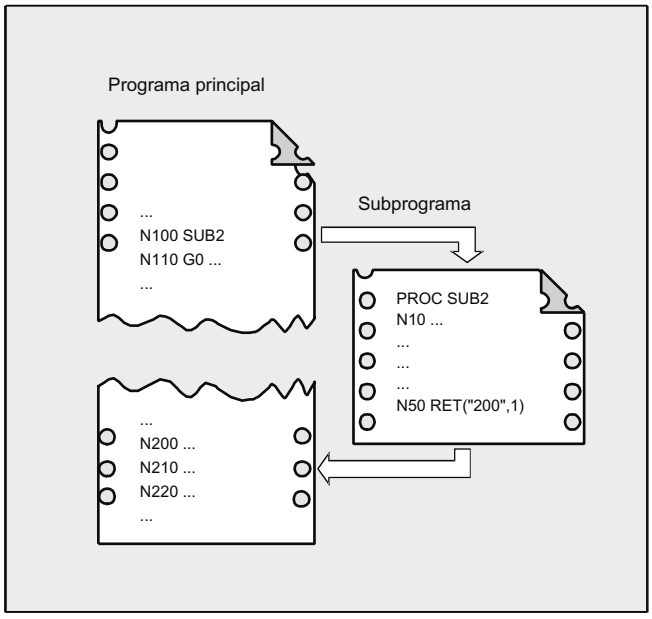
Los siguientes gráficos muestran los distintos efectos de los parámetros de salto atrás

1. <Secuencia meta> = "N200", <Secuencia tras secuencia meta> = 0



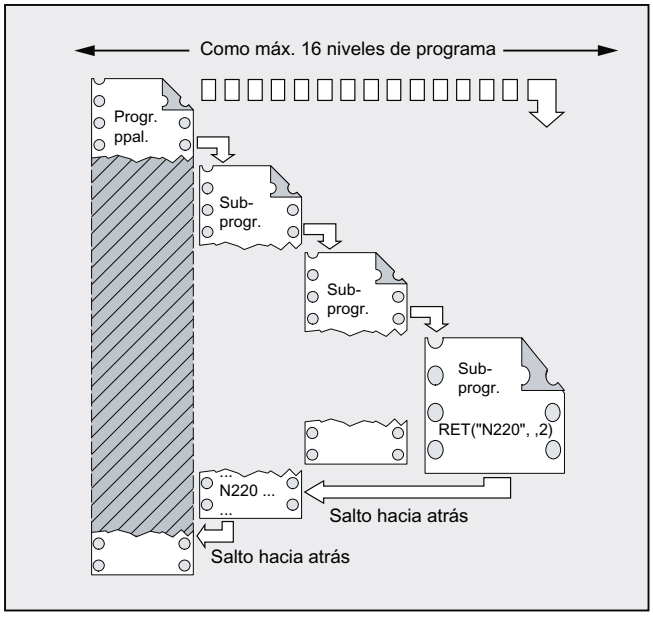
Después del comando `RET`, continúa la ejecución del programa con la secuencia `N200` en el programa principal.

2. <Secuencia meta> = "N200", <Secuencia tras secuencia meta> = 1



Después del comando `RET`, continúa la ejecución del programa con la secuencia (`N210`) que sigue a la secuencia `N200` en el programa principal.

3. <Secuencia meta> = "N220", <Cantidad niveles salto atrás> = 2



Después del comando `RET`, se salta atrás dos niveles de programa hacia atrás y continúa la ejecución del programa con la secuencia `N220`.

3.2.2.11 Salto atrás parametrizable al subprograma (RETB ...)

Por lo general, con el comando `RETB` se regresa desde un subprograma al programa desde el que se ha llamado. La ejecución se continúa con la línea de programa que sigue a la llamada al subprograma. Si la ejecución del programa debe continuar en otro punto, existen las siguientes posibilidades:

- Continuación de la ejecución del programa tras llamar a los ciclos de desbaste en modo de dialecto ISO (después de la descripción de contorno).
- Salto atrás al programa principal desde cualquier nivel de subprograma (también después de ASUP) en el tratamiento de errores.
- Salto atrás de varios niveles de programa para aplicaciones especiales en ciclos de compilación y en el modo de dialecto ISO.

A tal efecto es necesario programar el comando `RETB` con más parámetros.

Sentido de búsqueda

Al indicar el parámetro `<Secuencia meta>`, el salto atrás no se produce hasta la secuencia posterior a la secuencia de llamada. Seguidamente se busca la meta hacia el **inicio** del programa al que se regresa. Si la búsqueda no tiene éxito, a continuación se busca hacia el final del programa.

Sintaxis

```
RETB("<Secuencia meta>")
RETB("<Secuencia meta>",<Secuencia tras secuencia meta>)
```

```
RETB("<Secuencia
meta>",<Secuencia tras secuencia meta>,<Cantidad niveles salto atrás
>)
RETB("<Secuencia meta>", ,<Cantidad niveles salto atrás>)
RETB("<Secuencia
meta>",<Secuencia tras secuencia meta>,<Cantidad niveles salto atrás
>,
<Salto atrás hasta inicio programa>)
RETB( , ,<Cantidad niveles salto atrás>,<Salto atrás hasta inicio pr
ograma>)
```

Descripción

RETB:	Fin de subprograma	
<Secuencia meta>:	En el parámetro se indica como destino del salto la secuencia en la que debe continuar la ejecución del programa. Si no está programado el parámetro <Cantidad niveles salto atrás>, el destino del salto se encuentra en el programa desde el que se ha llamado al subprograma actual. Posibles indicaciones:	
	<Número de secuencia>	Número de la secuencia meta. La búsqueda del número de secuencia se realiza en el programa al que se regresa y en primer lugar hacia el inicio del programa.
	<Marca de salto>	Marca de salto que debe estar presente en el programa al que se regresa. La búsqueda de la marca de salto se realiza en el programa al que se regresa y en primer lugar hacia el inicio del programa.
	<Cadena de caracteres>	Cadena de caracteres que debe estar presente en el programa al que se regresa (p. ej., nombre de programa o de variable). La búsqueda de la cadena de caracteres se realiza en el programa al que se regresa y en primer lugar hacia el inicio del programa. Para la programación de la cadena de caracteres en la secuencia meta, son válidas las siguientes reglas: • **Espacio al final** (a diferencia de la marca de salto, que se identifica por un ":" al final). • **Antes de la cadena de caracteres** solo deben fijarse un número de secuencia y/o una marca de salto, pero **ningún comando de programa**.

`<Secuencia tras secuencia meta>`:	En este parámetro se indica si la ejecución del programa debe continuar en la secuencia indicada en el parámetro <Secuencia meta> o en la secuencia siguiente a esta.		
	Tipo:	INT	
	Valor:	0	El salto atrás se produce hasta la secuencia indicada en el parámetro <Secuencia meta>.
		> 0	El salto atrás se produce hasta la secuencia posterior a la indicada en el parámetro <Secuencia meta>.
`<Número de niveles de salto atrás>`:	En este parámetro se indica el número de niveles de programa que deben saltarse hacia atrás para buscar allí la secuencia meta y reanudar la ejecución del programa.		
	Tipo:	INT	
	Valor:	1	El programa continúa en el "nivel de programa actual -1" (como `RET` sin parámetros).
		2	El programa continúa en el "nivel de programa actual -2", es decir, se salta un nivel hacia atrás.
		3	El programa continúa en el "nivel de programa actual -3", es decir, se saltan dos niveles hacia atrás.
		...	
	Rango de valores:	1 ... 15	
`<Salto atrás hasta inicio de programa>`:	En este parámetro se indica si el programa debe reanudarse desde el principio en caso de salto atrás al programa principal estando activo allí el **modo de dialecto ISO**.		
	Tipo:	BOOL	
	Valor:	1	Si el salto atrás tiene lugar en el programa principal y allí hay activo un **modo de dialecto ISO**, se deriva al inicio del programa.

Nota

En el caso de salto atrás al subprograma con una cadena de caracteres como indicación para la búsqueda de secuencia meta, se busca siempre primero una marca de salto en el programa desde el que se llama.

Si debe definirse unívocamente un destino de salto con una cadena de caracteres, ésta no debe coincidir con el nombre de una marca de salto ya que, de lo contrario, el salto atrás al subprograma se ejecutará siempre en la marca de salto y no en la cadena de caracteres (ver ejemplo 2).

Condiciones

Al saltar atrás varios niveles de programa se evalúan las instrucciones `SAVE` de los distintos niveles de programa.

Si en un salto atrás de varios niveles de programa hay activo un subprograma modal y se ha programado en uno de los subprogramas saltados el comando de cancelación `MCALL` para el subprograma modal, éste seguirá activo.

ATENCIÓN
Error de programación Al saltar atrás varios niveles de programa, la responsabilidad de que se continúe con los ajustes modales correctos recae exclusivamente en el usuario. Esto puede lograrse, p. ej., programando una secuencia principal adecuada.

Ejemplo

Código de programa	Comentarios
EJEMPLO.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	

Código de programa	Comentarios
PROC END_CYC(…)	; Llamada en el programa principal línea N3040
N10000 …	
N15000 if status == 1	
N15010 RETB("START_CYC")	; Salto atrás al programa desde el que se llama EJEMPLO.MPF
	; Búsqueda de la cadena de caracteres "START_CYC"
	; Sentido de búsqueda: hacia atrás, hacia el inicio del programa
	; Continuación de la ejecución del programa con línea N3000
N15020 endif	
N15030 if status == 0	

Código de programa	Comentarios
N15040 RET	; Salto atrás al programa desde el que se llama EJEMPLO.MPF
	; Continuación de la ejecución del programa con línea N3050
N15050 endif	
N16000 RET("START_CYC")	; Salto atrás al programa desde el que se llama EJEMPLO.MPF
	; Búsqueda de la cadena de caracteres "START_CYC"
	; Sentido de búsqueda: hacia delante, hacia el final del programa
	; Continuación de la ejecución del programa con línea N4500
N17060 RETB	; Salto atrás al programa desde el que se llama EJEMPLO.MPF
	; Continuación de la ejecución del programa con línea N3050
	; RETB sin parámetros es idéntico a RET

3.2.3 Llamada de un subprograma

3.2.3.1 Llamada de subprograma sin transferencia de parámetros

La llamada de un subprograma se realiza bien con la dirección L y el número de subprograma, bien indicando el nombre de programa.

También se puede llamar a un programa principal como si se tratase de un subprograma. El fin de programa `M2` o `M30`, definido en el programa principal, se evalúa en este caso como `M17` (fin de programa con salto atrás al programa desde el que se realiza la llamada).

Nota

También se puede arrancar un subprograma como si fuese un programa principal.

Estrategia de búsqueda del control numérico:

¿Hay *_MPF?

¿Hay *_SPF?

De ello resulta: en el caso de que el nombre del subprograma llamado sea idéntico al nombre del programa principal desde donde se realiza la llamada, se abre de nuevo el programa desde el que se realiza la llamada. Este efecto, generalmente no deseado, se deberá evitar mediante la asignación unívoca de nombres tanto a programas como a subprogramas.

Nota

Los subprogramas que no requieren transferencia de parámetros también pueden llamarse desde un fichero de inicialización.

Sintaxis

```
L<Número>/<Nombre de programa>
```

Nota

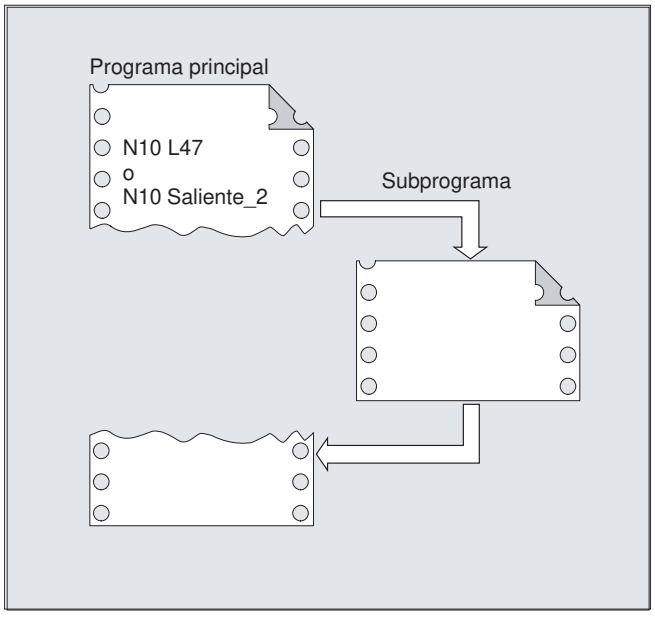
La llamada de un subprograma se tiene que programar siempre en una secuencia CN propia.

Descripción

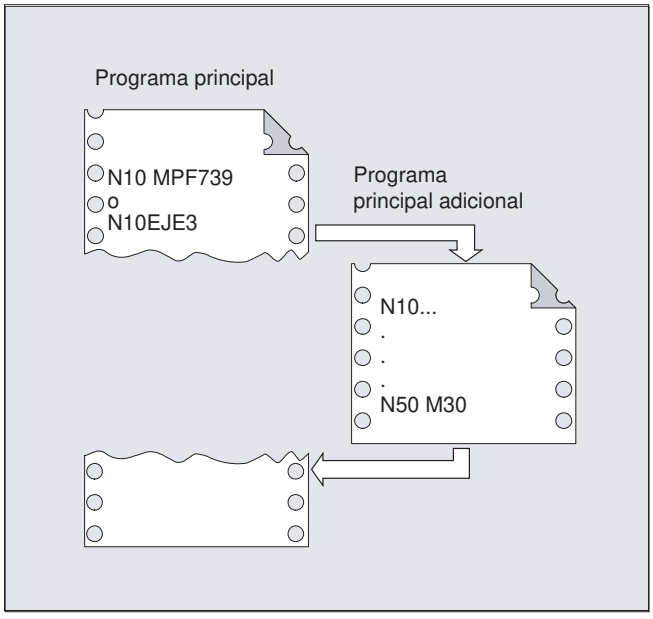
L:	Dirección para la llamada del subprograma	
<Número>:	Número del subprograma	
	Tipo:	INT
	Valor:	Como máximo 7 números decimales **Atención:** Los ceros a la izquierda son relevantes al asignar nombres (⇒ L123, L0123 y L00123 son tres subprogramas diferentes).
<Nombre de programa>:	Nombre del subprograma (o del programa principal)	

Ejemplos

Ejemplo 1: llamada de un subprograma sin transferencia de parámetros



Ejemplo 2: llamada de un programa principal como subprograma



Consulte también

Subprograma sin transferencia de parámetros (Página 511)

3.2.3.2 Llamada de subprograma con transferencia de parámetros (EXTERN)

En la llamada de subprograma con transferencia de parámetros se pueden transferir directamente variables o valores (no en parámetros `VAR`).

Los subprogramas con transferencia de parámetros se deben declarar con `EXTERN` antes de la llamada en el programa principal (p. ej., al inicio del programa). Para ello se indican el nombre del subprograma y los tipos de variables en el orden en que se transfieren.

ATENCIÓN
Peligro de confusión
Tanto los tipos de variables como el orden de transferencia deben coincidir con las definiciones establecidas con `PROC` en el subprograma. Los nombres de los parámetros pueden ser distintos en el programa principal y en el subprograma.

Sintaxis

```
EXTERN <Nombre del programa>(<Tipo Par1>,<Tipo Par2>,<Tipo Par3>)
...
<Nombre del programa>(<Valor Par1>,<Valor Par2>,<Valor Par3>)
```

Nota

La llamada de subprograma se tiene que programar en una secuencia CN propia.

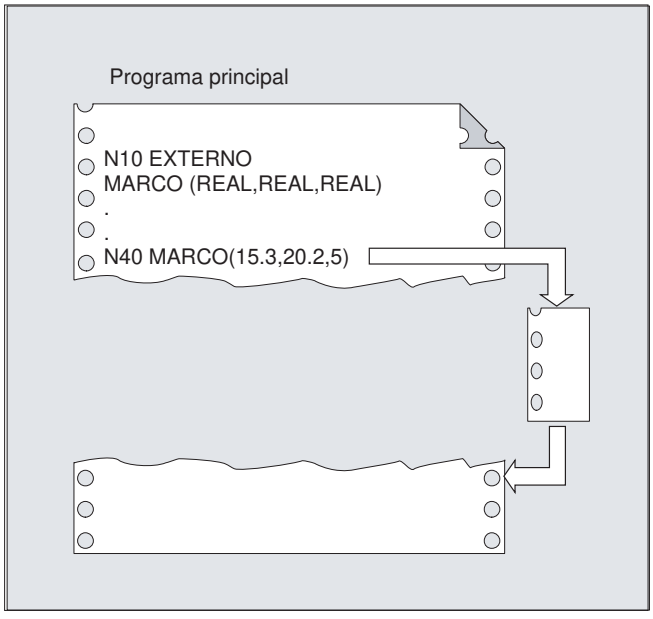
Descripción

`<Nombre de programa>`:	Nombre del subprograma
`EXTERN`:	Palabra reservada para declarar un subprograma con transferencia de parámetros **Nota:** `EXTERN` sólo debe indicarse cuando el subprograma está en el directorio de piezas o en el directorio global de subprogramas. Los ciclos no se declaran con `EXTERN` .
`<Tipo Par1>,<Tipo Par2>,<Tipo Par3>`:	Tipos de variable de los parámetros que deben transferirse en el orden de transmisión
`<Valor Par1>,<Valor Par2>,<Valor Par3>`:	Valores de variable para los parámetros que deben transferirse

Ejemplos

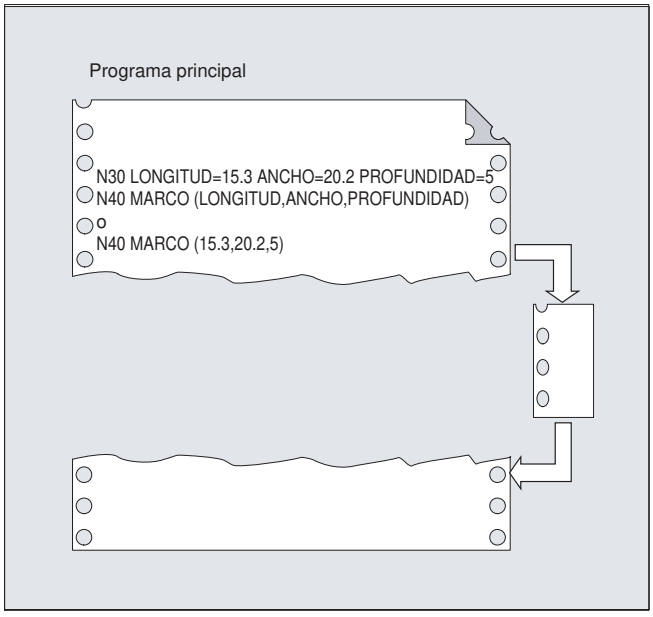
Ejemplo 1: llamada de subprograma con declaración anterior

Código de programa	**Comentarios**
`N10 MARCO EXTERNO(REAL,REAL,REAL)`	`; Indicación del subprograma.`
`...`	
`N40 MARCO(15.3,20.2,5)`	`; Llamada del subprograma con transferencia de parámetros.`



Ejemplo 2: llamada de subprograma sin declaración

Código de programa	Comentarios
N10 DEF REAL LARGO, ANCHO, PROFUNDIDAD	
N20 …	
N30 LONGITUD=15.3 ANCHO=20.2 PROFUNDIDAD=5	
N40 MARCO(LONGITUD,ANCHO,PROFUNDIDAD)	; o: N40 MARCO(15.3,20.2,5)



Consulte también

Subprograma con transferencia de parámetros Call-By-Value (PROC) (Página 512)

Subprograma con transferencia de parámetros Call-By-Reference (PROC, VAR) (Página 513)

3.2.3.3 Cantidad de repeticiones de programa (P)

Si se debe ejecutar un subprograma varias veces seguidas, en la secuencia de llamada del subprograma se puede programar el número de repeticiones deseado con la dirección `P`.

⚠ PRECAUCIÓN

Llamada de subprograma con repetición de programa y transferencia de parámetros

Los parámetros se transfieren sólo en el momento de la llamada al programa o en la primera pasada. Para el resto de pasadas no se alteran los parámetros. Si se desea modificar los parámetros en las repeticiones de programa es necesario realizar las definiciones correspondientes en el subprograma.

Sintaxis

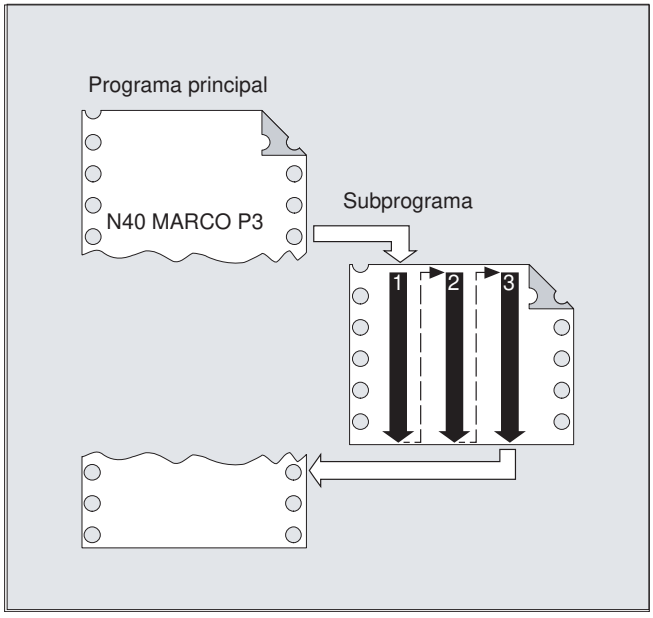
```
<Nombre de programa> P<Valor>
```

Descripción

<Nombre de programa>:	Llamada a un subprograma	
P:	Dirección para la programación de repeticiones de programa	
<Valor>:	Cantidad de repeticiones de programa	
	Tipo:	INT
	Rango de valores:	1 … 9999 (sin signo)

Ejemplo

Código de programa	**Comentarios**
...	
N40 MARCO P3	; El subprograma MARCO debe ejecutarse tres veces seguidas.
...	



3.2.3.4 Llamada de subprograma modal (MCALL)

La llamada de subprograma modal `MCALL <Nombre de programa>` hace que el subprograma indicado no se llame inmediatamente. En lugar de ello, a partir de ese momento la llamada se produce automáticamente en el programa de pieza después de cada secuencia de desplazamiento con desplazamientos de trayectoria. También pasando por todos los niveles de programa.

Nota

En una ejecución de programa actúa solo la última llamada de subprograma modal `MCALL <Nombre de programa>`. La llamada de subprograma modal actual sustituye la que estaba activa hasta el momento.

Si se transfieren parámetros al subprograma, dicha transferencia se produce únicamente en la llamada:

```
MCALL(<Nombre de programa>(Par1, Par2, ...))
```

Nota

La zona de definición de datos del principio del subprograma solo se recorre una vez con la ejecución de la secuencia con la llamada MCALL programada. En las siguientes llamadas de subprograma posteriores a las secuencias de desplazamiento ya no se recorre la zona de definición de datos. Por consiguiente, un valor asignado en la definición de una variable local de usuario (LUD) ya no está disponible tras la primera llamada, sino el último valor escrito en el ciclo. La variable tras la primera llamada no se vuelve a crear, sino que mantiene el último valor de la llamada anterior. Para evitar este comportamiento, se recomienda separar la definición LUD de la asignación del valor LUD, y programar la asignación de valor en una secuencia propia tras la zona de definición de datos.

ATENCIÓN

Llamadas de subprogramas modales sin desplazamiento de trayectoria

El subprograma modal se llama también en las siguientes situaciones, aunque no esté programado ningún desplazamiento de trayectoria:

- Programación de las direcciones S o F si está activo G0 o G1.
- Si G0 o G1 están solos en la secuencia o se han programado junto con otros comandos G.

Sintaxis

```
MCALL <Nombre de programa>
...
MCALL
```

Descripción

`MCALL <Nombre de programa>:`	Activar la función "Llamada de subprograma modal"
`<Nombre de programa>:`	Nombre del subprograma
`MCALL:`	Con MCALL pero sin indicar un nombre de programa, la función "Llamada de subprograma modal" se desactiva.

Condiciones

ASUP

Si se interrumpe la ejecución de un programa de pieza mediante un ASUP (ver capítulo "Rutina de interrupción (ASUP) (Página 555)"), no se ejecuta ninguna llamada de subprograma modal en dicho ASUP.

Si se inicia un ASUP en el estado del canal "Reset", este se comporta como un programa de pieza normal en lo que respecta a las llamadas de subprograma modales.

Ciclo de cambio de herramienta

Si en el ciclo de cambio de herramienta se cancela la función "Llamada de subprograma modal", debe prestarse atención a que el ciclo de cambio de herramienta, dado el caso, también sea llamado implícitamente tras la búsqueda de secuencia mediante el ASUP de búsqueda de secuencia, o bien manualmente mediante sobrememorización. En esta situación no debe cancelarse la función "Llamada de subprograma modal", pues el resultado de la búsqueda de secuencia se falsearía. Por tanto, se recomienda programar la cancelación de la función "Llamada de subprograma modal" en el ciclo de cambio de herramienta de la siguiente manera:

```
Código de programa          Comentarios
...
  IF $AC_ASUP == 0          ; La llamada no se realiza mediante el ASUP de búsqueda de
                              secuencia ni mediante sobrememorización.
    MCALL                   ; Desactivar la función "Llamada de subprograma modal".
  ENDIF
...
```

Ejemplos

Ejemplo 1

```
Código de programa          Comentarios
N10 G0 X0 Y0
N20 MCALL L70               ; Activar la llamada de subprograma modal para L70.
N30 X10 Y10                 ; Se realiza el desplazamiento a X10 Y10, a continuación se
                              llama a L70.
N40 X20 Y20                 ; Se realiza el desplazamiento a X20 Y20, a continuación se
                              llama a L70.
```

Código de programa	Comentarios
...	
N100 MCALL	; Desactivar la función "Llamada de subprograma modal".
N110 X0 Y0	; Se realiza el desplazamiento a X0 Y0, **no** se llama a L70.

Ejemplo 2

Código de programa
N10 G0 X0 Y0
N20 MCALL L70
N30 L80

En este ejemplo, las secuencias CN siguientes con ejes de contorneado programados se encuentran en el subprograma L80. L70 se llama a través de L80.

3.2.3.5 Llamada de subprograma indirecta (CALL)

En dependencia de las condiciones dadas se pueden llamar en un mismo punto diferentes subprogramas. A tal efecto, se memoriza el nombre del subprograma en una variable del tipo STRING. La llamada de subprograma se efectúa con `CALL` y el nombre de la variable.

Nota

La llamada indirecta de subprograma sólo resulta posible para subprogramas sin transferencia de parámetros. Para una llamada directa a un subprograma, el nombre debe consignarse en una constante STRING.

Sintaxis

```
CALL <nombre programa>
```

Descripción

`CALL`:	Comando para la llamada de subprograma indirecta	
`<Nombre de programa>`:	Nombre del subprograma (variable o constante)	
	Tipo:	STRING

Ejemplo

Llamada directa con constante STRING:

Código de programa	Comentarios
…	
CALL "/_N_WKS_DIR/_N_SUBPROG_WPD/_N_PIEZA1_SPF"	; Llamar directamente a subprograma PIEZA1 con CALL.
…	

Llamada indirecta mediante variable:

Código de programa	Comentarios
…	
`DEF STRING[100] PROGNAME`	`; Definir variable.`
`PROGNAME="/_N_WKS_DIR/_N_SUBPROG_WPD/_N_PIEZA1_SPF"`	`; Asignar subprograma PIEZA1 a la variable PROGNAME.`
`CALL PROGNAME`	`; Llamar indirectamente subprograma PIEZA1 mediante CALL y la variable PROGNAME.`
…	

3.2.3.6 Llamada de programa indirecta con indicación de la sección de programa que se va a ejecutar (CALL BLOCK ... TO ...)

Con `CALL` y la combinación de palabras reservadas `BLOCK ... TO` se llama indirectamente a un subprograma y se ejecuta la sección del programa identificada con la marca inicial y la marca final.

Sintaxis

```
CALL <>Nombre de programa> BLOCK <Marca inicial> TO <Marca final>
CALL BLOCK <Marca inicial> TO <Marca final>
```

Descripción

`CALL`:	Comando para la llamada de subprograma indirecta	
`<Nombre de programa>`:	Nombre del subprograma (variable o constante) que contiene la sección del programa que debe ejecutarse (**indicación opcional**).	
	Tipo:	STRING
	Nota: Si no hay programado ningún `<Nombre de programa>`, se busca y ejecuta la sección del programa identificada con `<Marca inicial>` y `<Marca final>` en el programa actual.	
`BLOCK ... TO ...` :	Combinación de palabras reservadas para la ejecución indirecta de secciones del programa	
`<Marca inicial>`:	Variable que remite al inicio de la parte del programa que debe ejecutarse.	
	Tipo:	STRING
`<Marca final>`:	Variable que remite al final de la parte del programa que debe ejecutarse.	
	Tipo:	STRING

Ejemplo

Programa principal:

Código de programa	Comentarios
...	
DEF STRING[20] STARTLABEL, ENDLABEL	; Definición de variables para la marca inicial y la final.
STARTLABEL="LABEL_1"	
ENDLABEL="LABEL_2"	
...	
CALL "CONTUR_1" BLOCK STARTLABEL TO ENDLABEL	; Llamada de subprograma indirecta e identificación de la sección de programa que va a ejecutarse.
...	

Subprograma:

Código de programa	Comentarios
PROC CONTUR_1 ...	
LABEL_1	; Marca inicial: inicio de la ejecución de la sección de programa
N1000 G1 ...	
...	
LABEL_2	; Marca final: fin de la ejecución de la sección de programa
...	

3.2.3.7 Llamada indirecta de un programa programado en lenguaje ISO (ISOCALL)

Con la llamada de programa indirecta ISOCALL puede llamarse a un programa programado en lenguaje ISO. Con ello se activa el modo ISO ajustado en los datos de máquina. Al final del programa el modo de procesamiento original vuelve a ser efectivo. Si no hay ningún modo ISO ajustado en los datos de máquina, la llamada del subprograma se efectúa en modo Siemens.

Encontrará **más información** sobre el modo ISO en el Manual de funciones Dialectos ISO

Sintaxis

```
ISOCALL <nombre programa>
```

Descripción

ISOCALL:	Palabra reservada para la llamada de subprograma indirecta con la que se activa el modo ISO ajustado en los datos de máquina.
<Nombre de programa>:	Nombre del programa programado en un lenguaje ISO (variable o constante del tipo STRING)

Ejemplo: llamada de un contorno con programación de ciclos desde el modo ISO

Código de programa	**Comentarios**
0122_SPF	; Descripción de contorno en el modo ISO
N1010 G1 X10 Z20	
N1020 X30 R5	
N1030 Z50 C10	
N1040 X50	
N1050 M99	
N0010 DEF STRING[5] PROGNAME = "0122"	; Programa de pieza Siemens (ciclo)
...	
N2000 R11 = $AA_IW[X]	
N2010 ISOCALL PROGNAME	
N2020 R10 = R10+1	; Ejecutar programa 0122.spf en el modo ISO
...	
N2400 M30	

3.2.3.8 Llamada a subprograma con ruta y parámetros (PCALL)

Con PCALL pueden llamarse subprogramas con indicación de ruta absoluta y transferencia de parámetros.

Sintaxis

```
PCALL <Ruta/nombre de programa>(<Parámetro 1>,…,<Parámetro n>)
```

Descripción

PCALL:	Palabra reservada para llamada de subprograma con ruta de acceso absoluta.
<Ruta/nombre de programa>:	Indicación de ruta absoluta incluido nombre de subprograma. Reglas sobre la indicación de ruta, ver "Direccionamiento de ficheros de la memoria de programas (Página 570)". Si no se ha indica ninguna ruta absoluta, PCALL se comporta como una llamada de subprograma estándar con identificador de programa. El nombre de programa se indica sin prefijo y sin identificación de fichero. Si el nombre del programa debe programarse con prefijo e identificación de fichero, debe declararse explícitamente con prefijo e identificación de fichero mediante el comando EXTERN.
<Parámetro 1>, ...:	Parámetro actual según la instrucción PROC del subprograma.

Ejemplo

Código de programa
PCALL/_N_WKS_DIR/_N_WELLE_WPD/WELLE(Parámetro 1, parámetro 2, …)

3.2.3.9 Ampliación de la ruta de búsqueda en la llamada de subprograma (CALLPATH)

El comando CALLPATH permite ampliar la ruta de búsqueda en las llamadas de subprograma. Así pueden llamarse también subprogramas desde un directorio de pieza no seleccionado sin indicar el nombre de ruta absoluto y completo del subprograma.

El modo de operación EES "EES sin GDIR" ofrece otra posibilidad de aplicación si se utiliza un directorio en una memoria de programas externa para guardar subprogramas globales. En este caso, CALLPATH permite ampliar la ruta de búsqueda añadiendo dicho directorio del subprograma.

La ampliación de la ruta de búsqueda tiene lugar antes de la entrada de ciclos de usuario (_N_CUS_DIR).

Los siguientes sucesos permiten anular la selección de la ampliación de la ruta de búsqueda:

- CALLPATH con espacio
- CALLPATH sin parámetros
- Fin del programa de pieza
- Reset

Sintaxis

```
CALLPATH("<Nombre de ruta>")
```

Descripción

CALLPATH:	Palabra reservada para la ampliación de la ruta de búsqueda programable. Se programa en una línea propia del programa de pieza.
<Nombre de ruta>:	Constante o variable del tipo STRING. Contiene la indicación de ruta absoluta del directorio con el que debe ampliarse la ruta de búsqueda. Reglas sobre la indicación de ruta, ver "Direccionamiento de ficheros de la memoria de programas (Página 570)".

Ejemplo

La ruta de búsqueda debe ampliarse con un directorio de piezas determinado:

```
Código de programa
...
CALLPATH ("/_N_WKS_DIR/_N_MYWPD_WPD")
...
```

Con ello se ajusta la siguiente ruta de búsqueda (el punto 5 es nuevo):

1. Directorio actual/*nombre*
2. Directorio actual/*nombre_SPF*
3. Directorio actual/*nombre_MPF*
4. //NC:/_N_SPF_DIR/*nombre_SPF*
5. **/_N_WKS_DIR/_N_MYWPD_WPD/nombre_SPF**
6. /N_CUS_DIR/*nombre_SPF*
7. /_N_CMA_DIR / *nombre_SPF*
8. /_N_CST_DIR / *nombre_SPF*

Condiciones

- `CALLPATH` verifica si el nombre de la ruta programado realmente existe. En caso de fallo, la ejecución del programa de pieza se interrumpe con la alarma de secuencia a corregir 14009.
- `CALLPATH` puede programarse también en ficheros INI. Es efectivo mientras dura la ejecución del fichero INI (fichero WPD-INI o programa de inicialización para datos CN activos, p. ej., frames en el 1.er canal _N_CH1_UFR_INI). Después de esto se restablece la ruta de búsqueda.

3.2.3.10 Ejecutar subprograma externo (EXTCALL)

Con el comando `EXTCALL` es posible recargar y ejecutar un programa de pieza desde una memoria externa.

Como memoria externa se dispone de:

- Unidad local
- Unidad de red
- Unidad USB

Nota

Como interfaz para la ejecución de un programa externo ubicado en una unidad USB, solo deben utilizarse las interfaces USB del panel de operador o TCU.

ATENCIÓN

Daños en la herramienta/pieza causados por la unidad Flash USB

Para ejecutar un subprograma externo se recomienda no utilizar una unidad Flash USB. Si se interrumpe la comunicación con la unidad Flash USB durante la ejecución del programa de pieza por problemas de contacto, caída, interrupción por golpes o desconexión accidental, se para inmediatamente el mecanizado. Esto podría dañar la herramienta o la pieza.

Ajuste predeterminado de la ruta externa del programa

La ruta al directorio externo del programa puede predefinirse con el dato de operador:

SD42700 $SC_EXT_PROG_PATH

La ruta completa del programa de pieza que se desea llamar se genera junto con la ruta o identificador del programa indicados en la llamada `EXTCALL`.

Nota

Si la ruta del programa solo debe indicarse a través de la llamada `EXTCALL`, SD42700 debe estar vacío.

Nota

Parámetros

Al llamar a un programa externo, no se le puede transferir ningún parámetro.

Sintaxis

```
EXTCALL("<Ruta/><Nombre de programa>")
```

Descripción

EXTCALL:	Comando para llamar a un subprograma externo	
"<Ruta/><Nombre de programa>":	Constante/variable del tipo STRING	
	<Ruta/>:	Ruta de acceso absoluta o relativa (**opcional**)
	<Nombre de programa>:	El nombre del programa se escribe sin prefijo "_N_". Se puede agregar al nombre del programa la extensión de fichero ("MPF", "SPF") con el carácter "_" o "." (**opcional**). Ejemplo: "EJE" "EJE_SPF" "EJE.SPF"

Definición de ruta: nombres abreviados

En la definición de la ruta se pueden utilizar los siguientes nombres abreviados:

- Unidad local: "**LOCAL_DRIVE:**"
- Tarjeta CF: "**CF_CARD:**"
- Unidad USB (panel de operador): "**USB:**"

Pueden utilizarse alternativamente las abreviaturas "CF_CARD:" y "LOCAL_DRIVE:".

Ejemplo

Ejecución desde unidad local

El programa principal "MAIN.MPF" se encuentra en la memoria CN y está seleccionado para la ejecución.

Subprograma "SP_1"

El subprograma externo "SP_1.SPF" o bien "SP_1.MPF" se encuentra en la unidad local, en el directorio "/user/sinumerik/data/prog/WKS.DIR/WST1.WPD".

La ruta al directorio del programa externo debe especificarse con:

SD42700 $SC_EXT_PROG_PATH = LOCAL_DRIVE:WKS.DIR/WST1.WPD

Nota

Ruta para la llamada del subprograma externo:

- Sin utilizar el ajuste predeterminado: "LOCAL_DRIVE:WKS.DIR/WST1.WPD/SP_1"
- Utilizando el ajuste predeterminado: "SP_1"

Subprograma "SP_2"

El subprograma externo "SP_2.SPF" o "SP_2.MPF" se encuentra en el directorio WKS.DIR/WST1.WPD de la unidad USB. El ajuste predeterminado de la ruta al directorio del programa

externo se utiliza para la ruta del subprograma "SP_1" y tampoco se cambia en el programa principal. Por tanto, al llamar el subprograma "SP_2" debe indicarse la ruta completa.

Programa principal "MAIN"

Código de programa
N010 PROC MAIN
N020 ...
N030 EXTCALL("SP_1")
N030 EXTCALL("USB:WKS.DIR/WST1.WPD/SP_2")
N050 ...
N060 M30

Información adicional

Llamada EXTCALL con definición de ruta absoluta

Si el subprograma existe en la ruta indicada, se ejecuta con la llamada EXTCALL. Si el subprograma no existe en la ruta indicada, la ejecución del programa se interrumpe con la llamada EXTCALL.

Llamada EXTCALL con definición de ruta relativa/sin definición de ruta

En una llamada EXTCALL con definición de ruta relativa o sin definición de ruta se buscan las memorias de programas existentes según el siguiente modelo:

1. Si hay una ruta predefinida en SD42700 $SC_EXT_PROG_PATH, se busca primero partiendo de esta ruta la indicación en la llamada EXTCALL (nombre de programa, en su caso, con su ruta relativa). La ruta absoluta resulta entonces de la concatenación de caracteres de:
   - Ruta predeterminada en SD42700 $SC_EXT_PROG_PATH
   - Carácter de separación "/"
   - Ruta y nombre de subprograma en el comando EXTCALL
2. Si el subprograma no se encuentra en 1, se busca en los directorios de la memoria de usuario.

La búsqueda finaliza cuando el subprograma se haya encontrado por primera vez. Si no se encuentra el subprograma, se cancela la ejecución del programa con la llamada EXTCALL.

Memoria de recarga ajustable (búfer FIFO)

Para ejecutar un subprograma externo se necesita una memoria de recarga. El tamaño de la memoria de recarga está predefinido en 30 kbytes y solo el fabricante de la máquina puede modificarlo (mediante MD18360 MM_EXT_PROG_BUFFER_SIZE).

Nota

Subprogramas con instrucciones de salto

En subprogramas externos que contienen instrucciones de salto (`GOTOF`, `GOTOB`, `CASE`, `FOR`, `LOOP`, `WHILE`, `REPEAT`, `IF`, `ELSE`, `ENDIF`, etc.), los destinos de salto deben encontrarse dentro de la memoria de recarga.

Nota

Programas ShopMill/ShopTurn

Los programas ShopMill y ShopTurn deben estar completamente guardados en la memoria de recarga debido a las descripciones de los contornos que hay insertadas en el final del fichero.

Para subprogramas externos que se ejecutan en paralelo se necesita cada vez una memoria de recarga propia.

Reset/Final de programa/POWER ON

Con Reset y POWER ON se cancelan las llamadas a subprograma externas y se borran las correspondientes memorias de recarga.

Un programa seleccionado para "Ejecución de externo" sigue seleccionado para "Ejecución de externo" después de Reset, fin de programa o POWER ON. Este comportamiento no difiere del de programas seleccionados internamente, siempre y cuando la memoria externa de programas siga estando disponible.

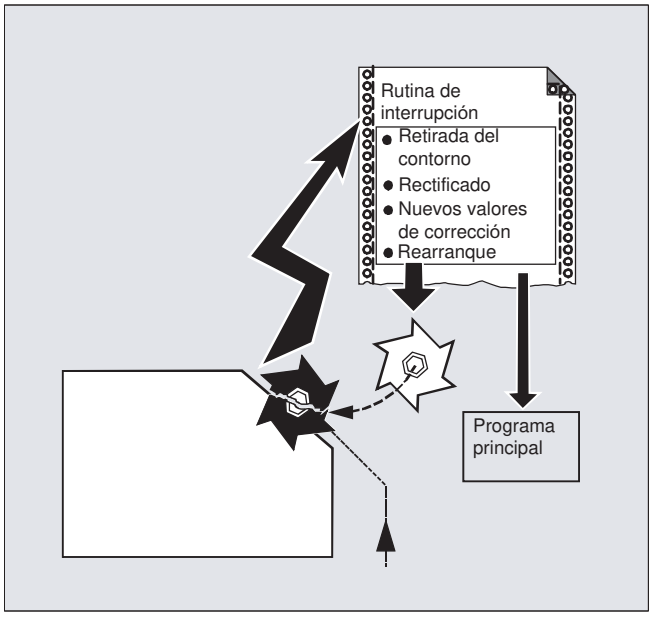
3.3 Rutina de interrupción (ASUP)

3.3.1 Función de una rutina de interrupción

Nota

Los términos "Subprograma asíncrono (ASUP)" y "Rutina de interrupción", que aparecen alternándose en la siguiente descripción, designan la misma funcionalidad.

La función de una rutina de interrupción se ilustra mediante un ejemplo típico:



Durante el mecanizado se detecta rotura de herramienta. De este modo, se dispara una señal que detiene el proceso de mecanizado y simultáneamente inicia un subprograma denominado rutina de interrupción. Dicha rutina contiene todas las instrucciones necesarias que se deben ejecutar en este caso.

Una vez ejecutado el subprograma (y con ello restablecida la disponibilidad operacional), el control numérico retorna al programa principal y continúa el mecanizado (dependiendo del comando REPOS) en el punto donde tuvo lugar la interrupción (ver "Reposicionamiento en el contorno (Página 882)").

⚠ PRECAUCIÓN

Peligro de colisión

Si no hay ningún comando REPOS programado en el subprograma, se colocará en el punto final de la secuencia que sigue a la secuencia interrumpida.

3.3.2 Creación de una rutina de interrupción

Creación de una rutina de interrupción como subprograma

La rutina de interrupción se define como un subprograma en la cabecera de definición.

Ejemplo:

```
Código de programa  Comentarios
PROC RETIR_Z        ; Nombre del programa "RETIR_Z"
N10 ...             ; A continuación siguen las secuencias CN.
...
N50 M17             ; Para concluir, final del programa y regreso al programa princi-
                    pal.
```

Almacenamiento de comandos G modales (SAVE)

La rutina de interrupción puede identificarse durante la definición con SAVE.

El atributo SAVE hace que se guarden los comandos G modales activos antes de la llamada de la rutina de interrupción y se reactiven de nuevo antes de finalizar dicha rutina (ver "Subprogramas con mecanismo SAVE (SAVE) (Página 515)").

De esta forma es posible continuar con la ejecución en el punto de interrupción tras finalizar la rutina de interrupción.

Ejemplo:

```
Código de programa
PROC RETIR_Z SAVE
N10 ...
...
N50 M17
```

Asignación de otras rutinas de interrupción (SETINT)

Dentro de una rutina de interrupción se pueden programar instrucciones SETINT (ver "Asignación y arranque de una rutina de interrupción (SETINT)" (Página 556)) y activar de este modo más rutinas de interrupción. Su ejecución se activa sólo con las correspondientes señales de entrada.

3.3.3 Asignar e iniciar una rutina de interrupción (SETINT, PRIO, BLSYNC)

El control dispone de varias entradas rápidas (entradas 1 a 8) que disparan su respectiva interrupción (1 a 8). A cada interrupción se le puede asignar una prioridad y una rutina de interrupción mediante el comando `SETINT`. Si la interrupción se dispara al activar la entrada rápida, se interrumpe la ejecución actual en el canal y se inicia la rutina de interrupción.

Prioridades en la interrupción

Si se asignan interrupciones a varias entradas de un programa de pieza, es necesario establecer diferentes prioridades para las interrupciones.

A una interrupción se le puede asignar un valor de prioridad de 1 a 128. El valor 1 es el de mayor prioridad; el valor 128, el de menor prioridad.

Sintaxis

```
SETINT(<n>) <NOMBRE>
SETINT(<n>) PRIO=<Valor> <NOMBRE>
SETINT(<n>) PRIO=<Valor> <NAME> BLSYNC
SETINT(<n>) PRIO=<Valor> <NOMBRE> LIFTFAST
```

Descripción

`SETINT(<n>):`	A la señal de interrupción <n> se le asigna el programa CN (ASUP) <Nombre>. La rutina de interrupción asignada se inicia en cuanto se detecta la señal de interrupción <n> == 1. **Nota:** Si a la señal de interrupción <n> se le asigna otra rutina de interrupción, la asignación anterior pierde su efecto.	
`<n>`:	Número de la señal de interrupción	
	Tipo:	INT
	Rango de valores:	1 ... 32
`PRIO=`:	Prioridad de la interrupción (opcional)	
`<Valor>`:	(**opcional**) Valor de prioridad	
	Tipo:	INT
	Rango de valores:	1 ... 128 (1 ⇒ prioridad más alta)
`<NOMBRE>`:	Nombre del programa CN (ASUP)	
`BLSYNC`:	(**opcional**) `BLSYNC` provoca que, tras dispararse la interrupción, primero se espere a que se haya ejecutado la secuencia actual. Solo entonces se inicia la rutina de interrupción.	
`LIFTFAST`:	(**opcional**) `LIFTFAST` provoca que, tras dispararse la interrupción, primero se realice una retirada rápida (ver capítulo "Retirada rápida del contorno (SETINT LIFTFAST, ALF) (Página 560)"). Solo entonces se inicia la rutina de interrupción.	

Condiciones

Reglas de interrupción

1. Por cada interrupción que no puede ejecutarse inmediatamente, o que está siendo ejecutada actualmente, se guarda otra solicitud de interrupción. El resto de solicitudes de interrupción para dicha interrupción se pierden.
2. Si se está ejecutando una interrupción y se dispara otra interrupción de mayor prioridad, esta última suspende la interrupción de menor prioridad. La interrupción de menor prioridad no se reanuda hasta que haya concluido la interrupción de mayor prioridad. Si mientras se ejecuta la interrupción de mayor prioridad llegan otras solicitudes para la interrupción de menor prioridad, se guarda una solicitud. Las demás se pierden.
3. Si se está ejecutando una interrupción y se dispara otra interrupción de mayor prioridad, esta última suspende la interrupción de menor prioridad. Se ejecutará la interrupción de mayor prioridad. Si, de nuevo se dispara una interrupción de mayor prioridad, la interrupción actual se interrumpe y se ejecuta la interrupción de mayor prioridad. Puede haber un máximo de seis niveles de interrupción activos. Es decir, un nivel de interrupción ejecutándose y cinco niveles en espera. Por cada nivel de interrupción activo se guarda como máximo una solicitud de interrupción adicional. Cualquier otra solicitud de interrupción se pierde. También se pierden solicitudes de interrupción si estas se realizan para otros niveles de interrupción (nivel de interrupción ≥ 7).

Ejemplos

Ejemplo 1: asignar rutinas de interrupción y definir prioridad

Código de programa	**Comentarios**
N20 SETINT(3) PRIO=1 ABHEB_Z	; SI entrada 3 == 1
	; ENTONCES arrancar rutina de interrupción "ABHEB_Z"
N30 SETINT(2) PRIO=2 ABHEB_X	; SI entrada 2 == 1
	; ENTONCES arrancar rutina de interrupción "ABHEB_X"

Las rutinas de interrupción se ejecutan una tras otra siguiendo el orden de los valores de prioridad cuando las entradas están presentes al mismo tiempo: en primer lugar "ABHEB_Z", a continuación "ABHEB_X".

Ejemplo 2: reasignar la rutina de interrupción

Código de programa	**Comentarios**
N20 SETINT(3) PRIO=2 ABHEB_Z	; SI entrada 3 == 1
	; ENTONCES arrancar rutina de interrupción **"ABHEB_Z"**
...	
N80 SETINT(3) PRIO=1 ABHEB_X	; SI entrada 3 == 1
	; ENTONCES arrancar rutina de interrupción **"ABHEB_X"**

3.3.4 Desactivación/reactivación de la asignación de una rutina de interrupción (DISABLE, ENABLE)

Una instrucción `SETINT` puede desactivarse con `DISABLE` y volver a activarse con `ENABLE` sin que se pierda la asignación entrada → rutina de interrupción.

Sintaxis

```
DISABLE(<n>)
ENABLE(<n>)
```

Descripción

`DISABLE(<n>):`	Comando: **desactivar** la asignación de rutinas de interrupción de la entrada <n>	
`ENABLE(<n>):`	Comando: **reactivar** la asignación de rutinas de interrupción de la entrada <n>	
`<n>:`	Parámetros: Número de la señal de interrupción	
	Tipo:	INT
	Rango de valores:	1 ... 32

Ejemplo

Código de programa	**Comentarios**
`N20 SETINT(3) PRIO=1 ABHEB_Z`	`; Cuando se conecta la entrada 3, debe`
	`; arrancar la rutina de interrupción "ABHEB_Z".`
`...`	
`N90 DISABLE(3)`	`; Se desactiva la instrucción SETINT de N20.`
`...`	
`N130 ENABLE(3)`	`; Se reactiva la instrucción SETINT de N20.`
`...`	

3.3.5 Borrado de la asignación de una rutina de interrupción (CLRINT)

Una asignación de una señal de interrupción definida con `SETINT` para un programa CN (ASUP) puede borrarse con `CLRINT`.

Sintaxis

```
CLRINT(<n>)
```

Descripción

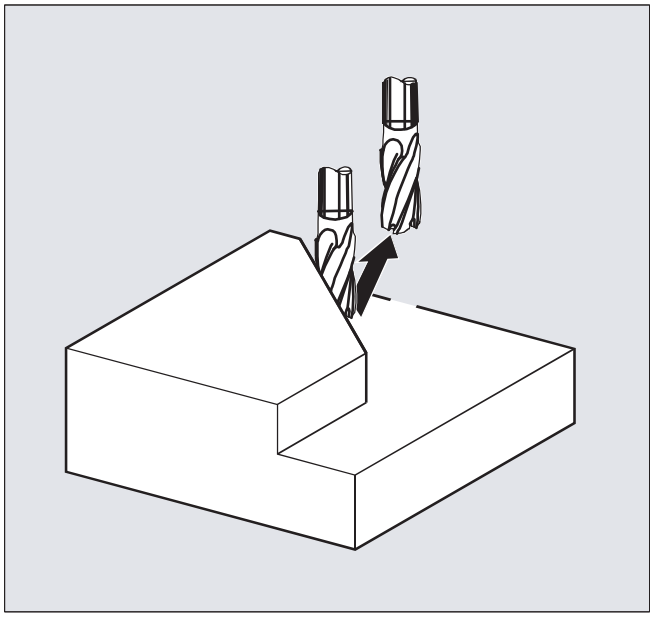
CLRINT(<n>):	Comando: Borrado de la asignación de la señal de interrupción <n> para el programa CN (ASUP) <n> definido con SETINT	
<n>:	Parámetros: Número de la señal de interrupción	
	Tipo:	INT
	Rango de valores:	1 ... 32

Ejemplo

Código de programa	Comentarios
N20 SETINT(3) PRIO=2 ABHEB_Z	
...	
N50 CLRINT(3)	; Se ha borrado la asignación entre la entrada "3" y la rutina de interrupción "RETIR_Z".

3.3.6 Retirada rápida del contorno (SETINT LIFTFAST, ALF)

En una instrucción SETINT con LIFTFAST la herramienta se retira mediante la retirada rápida del contorno de la pieza al conectar la entrada.



El proceso subsiguiente depende de si la instrucción SETINT contiene una rutina de interrupción además de LIFTFAST:

Con rutina de interrupción: **después** de la retirada rápida, se ejecuta la rutina de interrupción.
Sin rutina de interrupción: la ejecución se para con alarma después de la retirada rápida.

Sintaxis

```
SETINT(<n>) PRIO=1 LIFTFAST
SETINT(<n>) PRIO=1 <NOMBRE> LIFTFAST
```

Descripción

SETINT(<n>):	Comando: asignar la entrada <n> a una rutina de interrupción. La rutina de interrupción asignada arranca cuando se conecta la entrada <n>.	
<n>:	Parámetros: número de entrada	
	Tipo:	INT
	Rango de valores:	1 ... 8
PRIO= :	Definición de la prioridad	
<Valor>:	Valor de prioridad	
	Rango de valores:	1 ... 128
	La prioridad 1 corresponde a la máxima prioridad.	
<NOMBRE>:	Nombre del subprograma (rutina de interrupción) que debe ejecutarse.	
LIFTFAST:	Comando: retirada rápida del contorno	
ALF=… :	Comando: Dirección de desplazamiento programable (contenida en la secuencia de desplazamiento) En relación con las opciones de programación con ALF ver el tema "Sentido de desplazamiento en la retirada rápida del contorno (Página 562)".	

Condiciones

Comportamiento con frame activo con simetría especular

Antes de determinarse la dirección de retirada, se comprueba si está activo algún frame con simetría especular. En ese caso se intercambian a izquierda y derecha con relación a la dirección de la tangente, en la dirección de retirada. Los componentes en el sentido de la herramienta no se simetrizan. Este comportamiento se activa con el ajuste de DM:

DM21202 $MC_LIFTFAST_WITH_MIRROR = TRUE

Ejemplo

Una herramienta rota debe sustituirse automáticamente por otra herramienta idéntica. El mecanizado debe continuar con la nueva herramienta.

Programa principal:

Programa principal	**Comentarios**
N10 SETINT(1) PRIO=1 W_WECHS LIFTFAST	; Si se conecta la entrada 1, la herramienta se retira inmediatamente del contorno mediante retirada rápida (n.° de código 7 para la corrección de radio de herramienta G41). A continuación se ejecuta la rutina de interrupción "W_WECHS".

Programa principal	**Comentarios**
N20 G0 Z100 G17 T1 ALF=7 D1	
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	

Subprograma:

Subprograma	**Comentarios**
PROC CAMBIO_H SAVE	; Subprograma con almacenamiento del estado operativo actual
N10 G0 Z100 M5	; Desplazamiento a posición de cambio de herramienta, parada del cabezal
N20 T11 M6 D1 G41	; Cambio de herramienta
N30 REPOSL RMBBL M3	; Rearranque del contorno y salto atrás hasta el programa principal (se programa en una secuencia)

3.3.7 Sentido de desplazamiento en la retirada rápida del contorno

Movimiento de retroceso

El plano del movimiento de retroceso se define mediante los siguientes comandos G:

- LFTXT
  El plano del movimiento de retroceso se define a partir de la tangente de trayectoria y de la dirección de herramienta (ajuste estándar).
- LFWP
  El plano del movimiento de retroceso es el plano de trabajo activo que se selecciona con los comandos G G17, G18 o G19. La dirección del movimiento de retirada es independiente de la tangente a la trayectoria. Esto permite programar movimientos de retirada rápida que discurren en paralelo a un eje.
- LFPOS
  Retirada del eje declarado con POLFMASK/POLFMLIN a la posición de eje absoluta programada con POLF.
  ALF no influye en la dirección de retirada para varios ejes, como tampoco para varios ejes en relación lineal.

Sentido de desplazamiento programable (ALF=…)

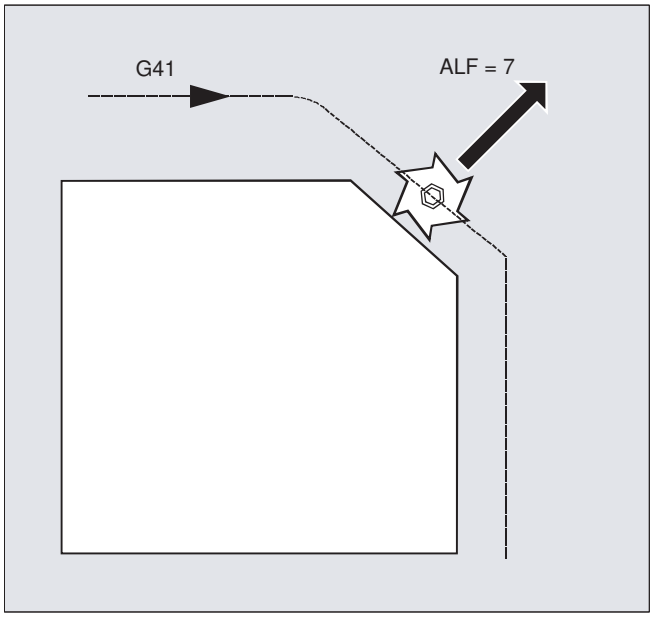
La dirección se programa en el plano de retirada con `ALF` en pasos discretos de 45 grados.

Los posibles sentidos de desplazamiento para estos casos se definen en el control numérico mediante códigos numéricos especiales; que a su vez sirven para poder llamarlos.

Ejemplo:

Código de programa
`N10 SETINT(2) PRIO=1 RETIR_Z LIFTFAST ALF=7`

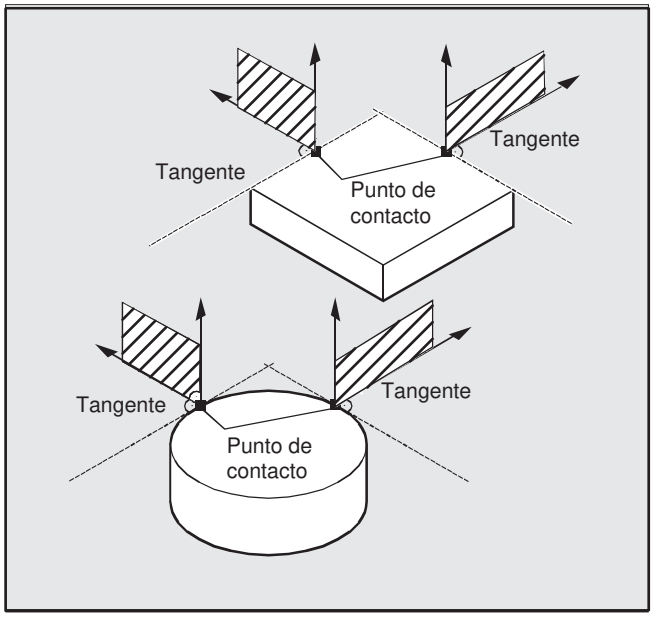
La herramienta se retira con `G41` activo (dirección de mecanizado a la izquierda del contorno) en dirección perpendicular al contorno.



Plano de referencia para la descripción de los sentidos de desplazamiento con LFTXT

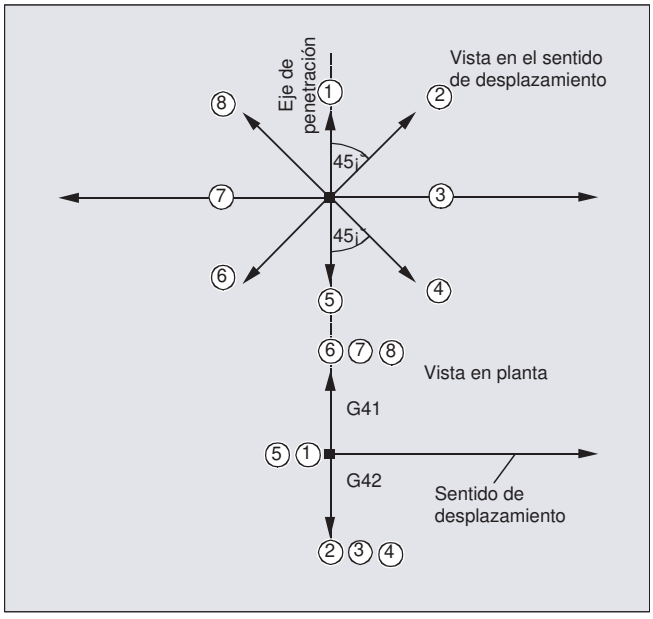
En el punto de contacto entre la herramienta y el contorno programado se define un plano de trabajo que sirve como referencia para definir los códigos numéricos de los movimientos de retirada.

El plano de referencia queda definido por el eje longitudinal de la herramienta (eje de penetración) y un vector perpendicular a dicho eje y a su vez perpendicular a la tangente al contorno por el punto de contacto.



Números de código con sentidos de desplazamiento con LFTXT

A partir del plano de referencia, encontrará en la siguiente figura los números de código con sentidos de desplazamiento.



Para `ALF=1` la retirada se define en la dirección de la herramienta.

Con `ALF=0` está desactivada la función "Retirada rápida".

⚠ PRECAUCIÓN

Peligro de colisión

Si la corrección de radio de herramienta está activada:

- los códigos 2, 3, 4 con `G41` y
- los códigos 6, 7, 8 con `G42`

no deben utilizarse ya que, en estos casos, la herramienta se desplazaría hacia el contorno y colisionaría con la pieza.

Números de código con sentidos de desplazamiento con LFWP

Con `LFWP`, la dirección en el plano de mecanizado se asigna de la siguiente manera:

- `G17`: Plano X/Y
  `ALF=1`: Retirada en dirección X
  `ALF=3`: Retirada en dirección Y
- `G18`: Plano Z/X
  `ALF=1`: Retirada en dirección Z
  `ALF=3`: Retirada en dirección X
- `G19`: Plano Y/Z
  `ALF=1`: Retirada en dirección Y
  `ALF=3`: Retirada en dirección Z

Consulte también

Retirada rápida durante el tallado de roscas (LFON, LFOF, DILF, ALF, LFTXT, LFWP, LFPOS, POLF, POLFMASK, POLFMLIN) (Página 236)

3.3.8 Desarrollo del movimiento en rutinas de interrupción

Rutina de interrupción sin LIFTFAST

Los desplazamientos de eje se frenan en la trayectoria hasta alcanzar la parada. A continuación arranca la rutina de interrupción.

La posición de parada se almacena como posición de interrupción y se alcanza al final de la rutina de interrupción en `REPOS` con `RMIBL`.

Rutina de interrupción con LIFTFAST

Los desplazamientos de eje se frenan en la trayectoria. Al mismo tiempo, el desplazamiento `LIFTFAST` se ejecuta como desplazamiento superpuesto. Si el desplazamiento en la trayectoria y el desplazamiento `LIFTFAST` se han parado, se inicia la rutina de interrupción.

Como posición de interrupción se almacena la posición en el contorno en la que se ha iniciado el movimiento `LIFTFAST` y, con ello, abandonado la trayectoria.

La rutina de interrupción se comporta con `LIFTFAST` y `ALF=0` de forma idéntica que la rutina de interrupción sin `LIFTFAST`.

Nota

El valor de retirada de los ejes geométricos en la retirada rápida del contorno puede ajustarse mediante un dato de máquina.

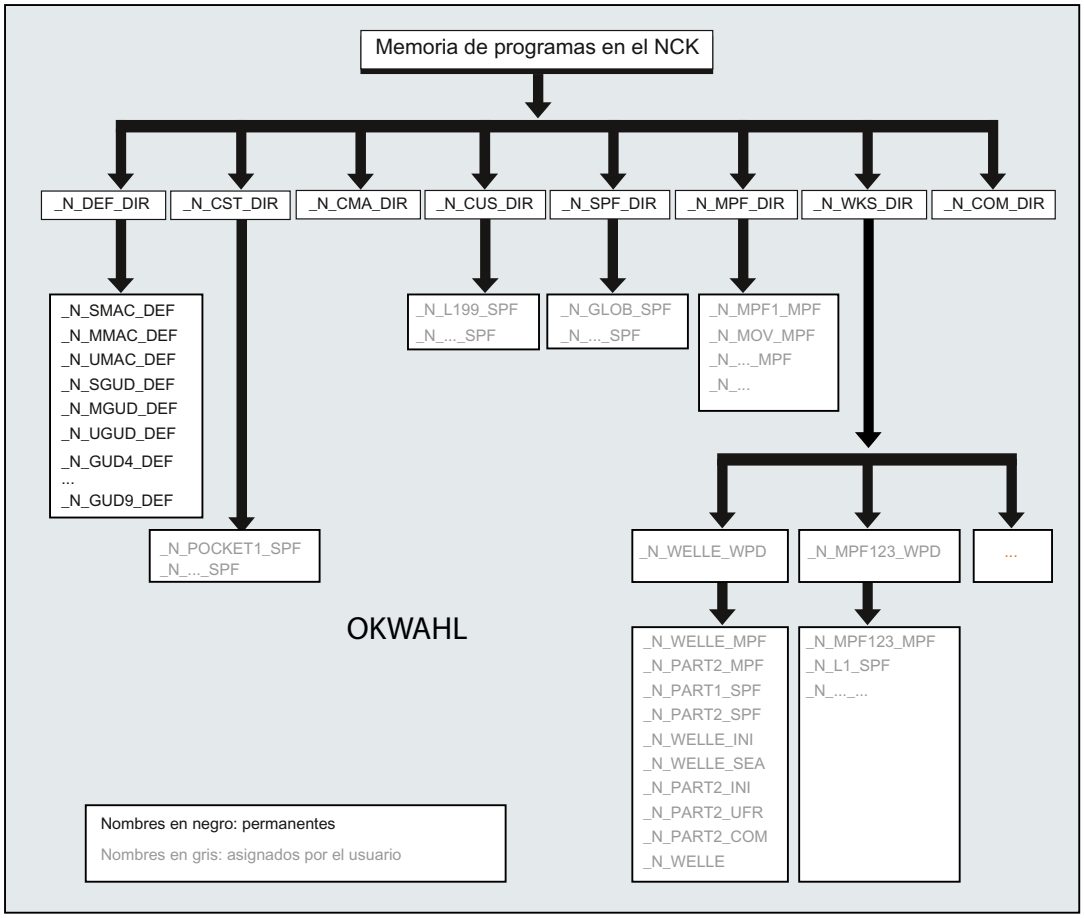
3.4 Administración de programas y ficheros

3.4.1 Memoria de programas

3.4.1.1 Memoria de programas en el NCK

En la memoria de programas se guardan ficheros y programas (p. ej. programas principales y subprogramas, definiciones de macro) de modo persistente (→ sistema pasivo de ficheros).

Además, ciertos tipos de archivo pueden guardarse aquí para transferirlos en caso necesario (p. ej. al mecanizar una pieza determinada) a la memoria principal (p. ej., para inicialización).



Directorios estándar

Como estándar existen los siguientes directorios:

Directorio	Contenido
_N_DEF_DIR	Bloques de datos y módulos de macro
_N_CST_DIR	Ciclos estándar

Directorio	Contenido
_N_CMA_DIR	Ciclos del fabricante
_N_CUS_DIR	Ciclos de usuario
_N_WKS_DIR	Piezas
_N_SPF_DIR	Subprogramas globales
_N_MPF_DIR	Programas principales
_N_COM_DIR	Comentarios

Tipos de fichero

En la memoria de programas se pueden almacenar los siguientes tipos de fichero:

Tipo de fichero	Descripción
<Nombre>_MPF	Programa principal
<Nombre>_SPF	Subprograma
<Nombre>_TEA	Datos de máquina
<Nombre>_SEA	Datos del operador
<Nombre>_TOA	Correcciones de herramientas
<Nombre>_UFR	Decalajes de origen/frames
<Nombre>_INI	Fichero de inicialización
<Nombre>_GUD	Datos globales del usuario
<Nombre>_RPA	Parámetros R
<Nombre>_COM	Comentarios
<Nombre>_DEF	Definición para datos de usuario globales y macros

Directorio raíz de piezas (_N_WKS_DIR)

El directorio raíz de piezas está creado por defecto con el nombre _N_WKS_DIR en la memoria de programas de pieza. En el directorio raíz de piezas se encuentran los correspondientes directorios de piezas de todas las piezas programadas.

Directorios raíz de piezas (..._WPD)

Un directorio de pieza contiene todos los archivos necesarios para mecanizar una pieza. Estos pueden ser programas principales, subprogramas, ficheros de inicialización y ficheros de comentario.

Los programas de inicialización se ejecutan una vez después de seleccionar el programa con el primer inicio de programa de pieza (según el dato de máquina MD11280 $MN_WPD_INI_MODE).

Ejemplo:

El directorio de pieza _N_EJE_WPD creado para la pieza EJE contiene los siguientes ficheros:

Archivo	Descripción
_N_EJE_MPF	Programa principal
_N_PART2_MPF	Programa principal

Archivo	Descripción
_N_PART1_SPF	Subprograma
_N_PART2_SPF	Subprograma
_N_EJE_INI	Programa general de inicialización de datos para la pieza
_N_EJE_SEA	Programa de inicialización para los datos de operador
_N_PART2_INI	Programa general de inicialización para los datos del programa Part 2
_N_PART2_UFR	Programa de inicialización para los datos del frame asociados al programa Part 2
_N_EJE_COM	Fichero de comentarios

En el directorio de pieza pueden almacenarse también datos no requeridos inmediatamente para el mecanizado mediante el CN. Puede tratarse de archivos ASCII, archivos binarios como p. ej. imágenes en formato JPG, o descripciones en formato PDF. Para que el CN los interprete como archivos binarios, sus extensiones deben ser conocidas por el CN (ajuste durante la puesta en marcha mediante MD17000 $MN_ EXTENSIONS_OF_BIN_FILES; en la configuración básica están preasignadas las siguientes extensiones: JPG, GIF, PNG, BMP, PDF, ICO, HTM).

Selección del programa de pieza para el mecanizado

Es posible asociar un directorio para la ejecución en un cierto canal. Si en dicho directorio se encuentra un programa principal **con el mismo nombre** o un único programa principal (_MPF), éste queda automáticamente seleccionado para su ejecución.

3.4.1.2 Memoria de programas externa

Además del sistema de ficheros pasivo del CN, en una máquina también puede haber disponibles memorias externas de programas (p. ej., en la unidad local o en una unidad de red).

Las funciones "Ejecución de externo" o "EES (Execution from External Storage)" permiten ejecutar programas de pieza **directamente** desde memorias externas de programas.

Información adicional: Manual de funciones Funciones básicas

Memoria global de programas de pieza (GDIR)

Al definir las unidades de disco puede declararse una de ellas como memoria global de programas de pieza (GDIR).

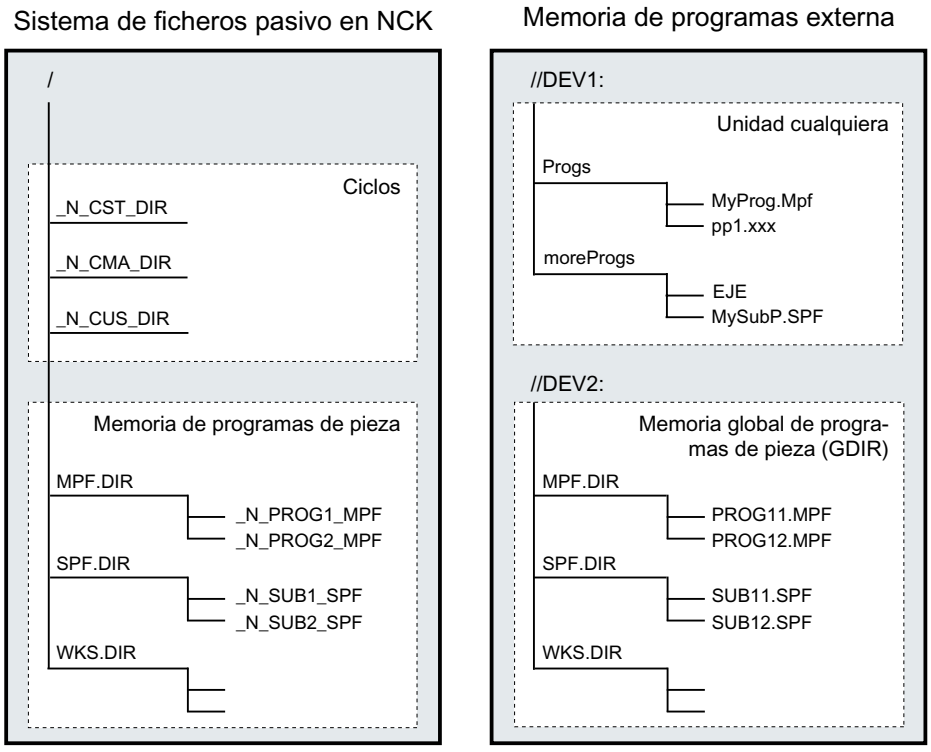
Tras ello, el sistema creará automáticamente los directorios MPF.DIR, SPF.DIR y WKS.DIR en el disco. Estos tres directorios forman el GDIR.

El GDIR es relevante solo para la función EES. Según la configuración de la unidad, el GDIR sustituye o amplía la memoria de programa de pieza CN. Sin embargo, la configuración de un GDIR no es obligatoria para el modo de operación EES.

Los directorios y ficheros del GDIR pueden direccionarse en el programa de pieza del mismo modo que en el sistema de ficheros pasivo. Esto hace posible el desplazamiento de un programa CN con indicaciones de ruta desde el sistema de ficheros pasivo al GDIR. El directorio SPF.DIR del GDIR está contenido en la ruta de búsqueda para subprogramas.

Organización del programa

La siguiente figura muestra la organización de los programas en las memorias externas de programas:



Sistemas de ficheros case-insensitive

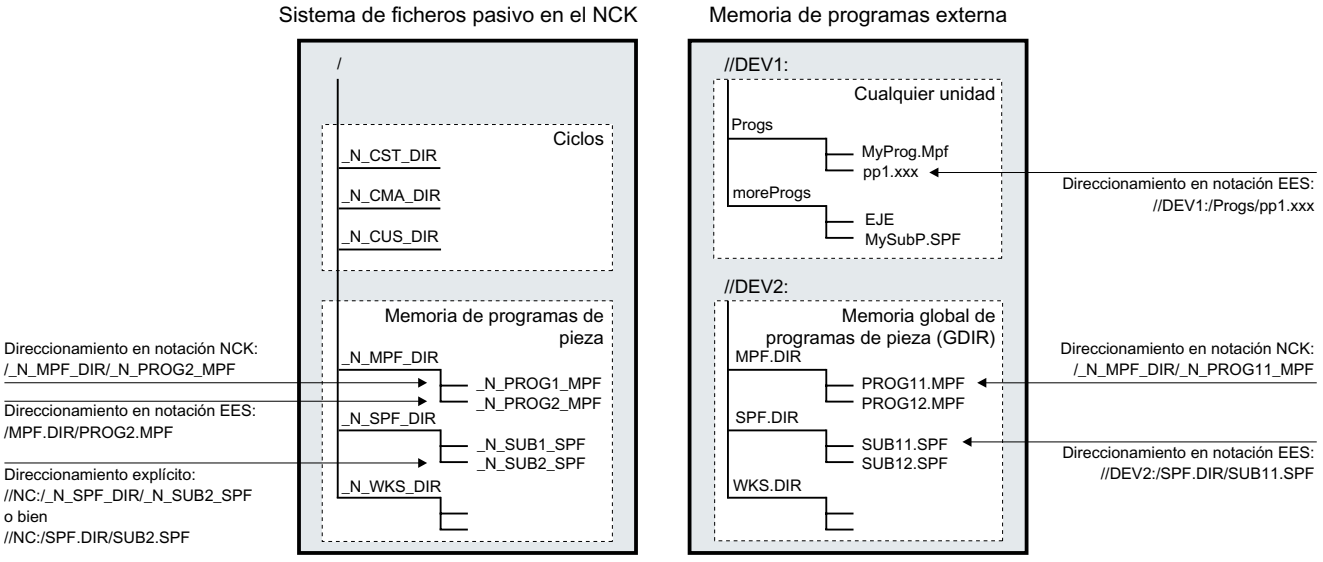
Nota

Para evitar problemas debidos al uso de mayúsculas y minúsculas al direccionar ficheros (ver "Direccionamiento de ficheros de la memoria de programas (Página 570)"), se recomienda utilizar sistemas de ficheros **case-insensitive** como memoria de programa externa.

3.4.1.3 Direccionamiento de ficheros de la memoria de programas

Un fichero de la memoria de programas que se activa por medio de un comando para manejar archivos (p. ej., WRITE, DELETE, READ, ISFILE, FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) se referencia a través de una ruta absoluta más el nombre del fichero o solo a través del nombre del fichero. En este último caso, se utiliza la ruta del programa seleccionado como ruta de fichero.

Direccionamiento en notación CN/EES



Direccionamiento de ficheros del sistema de ficheros pasivo

Los ficheros del sistema de ficheros pasivo se suelen direccionar en **notación CN** (los nombres de directorio y de fichero empiezan con la identificación de dominio "_N_"; el carácter de separación para la identificación de directorio/fichero es "_") sin indicar el nombre de la unidad. Sin embargo, también se admite el direccionamiento en **notación EES** (sin identificación de dominio "_N_"; el carácter de separación para la identificación de directorio/fichero es ".").

Ejemplo:

- Notación CN: "/_N_SPF_DIR/_N_SUB1_SPF"
- Notación EES: "/SPF.DIR/SUB1.SPF"

Nota

Los direccionamientos de ficheros del sistema de ficheros pasivo en notación EES se convierten internamente a notación CN de acuerdo con las reglas siguientes:

- A los nombres de directorio y fichero se les agrega la identificación de dominio "_N_".
- Si el cuarto carácter del nombre del directorio o fichero empezando por el final es un punto ("."), se convierte en un carácter de subrayado ("_").

También puede direccionarse selectivamente el sistema de ficheros pasivo mediante el nombre de unidad predefinido "//NC:".

Ejemplo:

- Notación CN: "//NC:/_N_SPF_DIR/_N_SUB1_SPF"
- Notación EES: "//NC:/SPF.DIR/SUB1.SPF"

Direccionamiento de ficheros de una memoria externa de programas

El direccionamiento de ficheros de una memoria de programas externa que no esté declarada como GDIR debe realizarse en notación EES. Al principio de la ruta de direccionamiento debe indicarse el nombre de la unidad (p. ej., "//DEV1:"). Se admiten todos los nombres simbólicos de equipos configurados en /user/sinumerik/hmi/cfg/logdrive.ini

Ejemplo:

- Notación EES: "//DEV1:/MyProgDir/pp1.xxx"
- Notación de CN: no admisible

Direccionamiento de ficheros de la memoria global de programas de pieza (GDIR)

En el direccionamiento de ficheros del GDIR se permite indicar la ruta, además de en notación EES, también en notación CN.

Ejemplo:

- Notación EES: "//DEV2:/MPF.DIR/PROG11.MPF"
- Notación CN: "/_N_MPF_DIR/_N_PROG11_MPF"

Nota

Los direccionamientos de ficheros del GDIR en notación CN se convierten internamente a notación EES de acuerdo con las reglas siguientes:

- Se elimina la identificación de dominio "_N_" de los nombres de directorio y fichero.
- Si el cuarto carácter del nombre del directorio o fichero empezando por el final es un carácter de subrayado ("_"), se convierte en un punto (".").

Reglas para la ruta de acceso

La ruta de acceso completa está compuesta por el nombre de unidad, la ruta del directorio y el nombre de fichero.

Nombre de la unidad

Para indicar el nombre de la unidad se aplican las siguientes reglas:

- Se admiten todos los nombres simbólicos de equipos configurados en /user/sinumerik/hmi/cfg/logdrive.ini
- Al principio se utiliza el carácter "//", seguido de al menos una letra o una cifra.
- Los caracteres siguientes pueden ser letras, cifras, "_" o espacios combinados libremente.
- El nombre concluye con una letra o una cifra seguidas de ":".
- No se permiten otros caracteres especiales.

Nota

Para el sistema de ficheros pasivo está predefinido el nombre de unidad "//NC:".

Ejemplos:

- Memoria de programas externa:
  - //Drive1:
  - //Drive_1:
  - //Drive 1:
  - //A B:
  - //1 B C 2:

Ruta de directorio

Para indicar el nombre del directorio se aplican las siguientes reglas:

- Al principio y al final de la ruta del directorio y como carácter de separación entre los distintos componentes de la ruta se utiliza "/".

  Nota

  La doble barra inclinada ("//") **no** está permitida dentro de la ruta del directorio.

- Nombres de directorio:
  - Los nombres de directorio deben empezar con una letra o una cifra. Los nombres de directorio empiezan con la identificación de dominio "_N_" únicamente en el direccionamiento en notación CN.
  - Los siguientes caracteres pueden ser letras, cifras o "_" combinados libremente.

    Nota

    En las memorias de programa externas se admiten espacios en los nombres de directorio. Esto no se aplica cuando la memoria de programas externa está configurada como memoria global de programas de pieza (GDIR).

  - No se permiten otros caracteres especiales.
- Extensiones de directorio:
  - Las extensiones de directorio deben contener exactamente tres letras o cifras.
  - Se separan del nombre del directorio con "_" (notación CN) o "." (notación EES).

  Nota

  En el sistema de ficheros pasivo solo existen las extensiones de fichero _DIR y _WPD.

Ejemplos:

- Sistema de ficheros pasivo o GDIR:
  - Notación CN: _N_WKS_DIR/_N_MYNCPROGS_WPD/...
  - Notación EES: WKS.DIR/MYPROGS.WPD/...
- Memoria de programas externa:
  - /abc
  - /ab_c.def
  - /ab c1.def
  - /a b c .d11
  - /abc.def/ghi.klm

Nombre de fichero

Para los nombres de fichero se aplican las siguientes reglas:

- Los nombres de fichero empiezan con la identificación de dominio "_N_" únicamente en el direccionamiento en notación CN.
- Los dos caracteres siguientes deberían ser dos letras o un carácter de subrayado seguido de una letra.

  Nota

  Si se cumple esta condición, un programa CN puede llamarse como subprograma desde otro programa con solo indicar su nombre. Si, por el contrario, el nombre de programa empieza con números, la llamada de subprograma solo es posible mediante la instrucción CALL.

- Los siguientes caracteres pueden ser letras, cifras o "_" combinados libremente.
- Extensión de fichero:
  - La extensión de fichero debe contener exactamente tres letras o cifras.

    Nota

    Encontrará las extensiones de fichero autorizadas en el sistema de ficheros pasivo en "Memoria de programas en el NCK (Página 567)".

  - Se separa del nombre del fichero con "_" (notación NC) o "." (notación EES).

Ejemplos:

- Sistema de ficheros pasivo o GDIR:
  - Notación CN: _N_SUB1_SPF
  - Notación EES: SUB1.SPF
- Memoria de programas externa:
  - Pieza1
  - _Pieza1
  - Pieza_1.spf
  - Pieza1.mpf

Nombre de subprograma DIN

Para los nombres de subprograma DIN se aplican las siguientes reglas:

- El primer carácter es la letra "L".
- Los siguientes caracteres son cifras (una como mínimo).
- Extensión de fichero:
  - La extensión de fichero debe contener exactamente tres letras.
  - Se separa del nombre del fichero con "_" (notación NC) o "." (notación EES).

Ejemplos:

- L123
- L1_SPF (notación CN) o L1.SPF (notación EES)

Longitud máxima de la ruta

Para especificar el nombre de unidad y la ruta del directorio se dispone de un máximo de 128 bytes. La longitud máxima del nombre de fichero es de 31 bytes. Por lo tanto, se dispone un máximo de 159 bytes para la ruta completa.

3.4.1.4 Ruta de búsqueda en la llamada de subprogramas

En la llamada de subprogramas sin indicación de ruta, la ruta absoluta se determina ejecutando una ruta de búsqueda fija.

En este caso, se busca en las memorias de programa en el siguiente orden:

		Directorio	Descripción
1		Directorio actual/*nombre*	El directorio actual es el directorio desde el que se ha abierto el programa. Este puede ser: • un directorio de piezas o el directorio estándar _N_MPF_DIR en la memoria de programas de pieza del CN o la memoria global de programas de pieza o bien • un directorio cualquiera de una memoria de programas externa
2		Directorio actual/*nombre_SPF*	
3		Directorio actual/*nombre_MPF*	
4	a	//NC:/_N_SPF_DIR / *nombre_SPF*	Directorio de subprogramas en la memoria de programas de pieza del CN
	b	//DEV2:/_N_SPF_DIR / nombre_SPF [1]	Directorio de subprogramas en la memoria global de programas de pieza **Nota:** Este paso de la búsqueda se elimina si no se ha configurado ninguna memoria global de programas de pieza o el programa se ha abierto en la memoria de programas de pieza del CN.
5		Ampliación de la ruta de búsqueda programada con `CALLPATH` (ver "Ampliación de la ruta de búsqueda en la llamada de subprograma (CALLPATH) (Página 549)"). **Nota:** Este paso de la búsqueda se elimina si no se ha programado ningún `CALLPATH`.	
6		/_N_CUS_DIR / *nombre_SPF*	Directorio de ciclos de usuario
7		/_N_CMA_DIR / *nombre_SPF*	Directorio de ciclos de fabricante
8		/_N_CST_DIR / *nombre_SPF*	Directorio de ciclos estándar

[1] //DEV2:" ejemplariza la unidad en la que está configurada la memoria global de programas de pieza.

Para la búsqueda se aplican las siguientes reglas:

- La ruta de búsqueda se ejecuta para cada una de las llamadas de subprograma, es decir, el lugar en el que se encuentra el programa superior no es relevante.
- En función del directorio se tienen en cuenta tipos de fichero diferentes.
- Por principio, dentro de un directorio no se busca en directorios subordinados, es decir, anidados.

3.4.1.5 Consulta de ruta y nombre de fichero

Para consultar la ruta y el nombre de fichero de un programa CN se dispone de las siguientes variables de sistema legibles en el programa de pieza:

Variable de sistema	Tipo	Descripción
$P_STACK	INT	Proporciona el nivel de programa en el que se ejecuta el programa CN actual.
$P_PATH[<n>]	STRING	Proporciona la ruta del programa CN que se ejecuta en el nivel de programa seleccionado mediante el índice de matriz <n>. Ejemplos: $P_PATH[0] proporciona la ruta para el programa principal, p. ej. "/_N_WKS_DIR/_N_EJE_WPD/". $P_PATH[$P_STACK - 1] proporciona la ruta del programa desde el que se llama. Si la ruta hace referencia a un programa CN guardado en el sistema de ficheros pasivo del CN o en la memoria global de programas de pieza (GDIR), la ruta se suministra en notación CN. Si la ruta hace referencia a un programa CN ejecutado por una memoria externa de programas diferente a la memoria global de programas de pieza, $P_PATH proporciona la ruta en notación EES.
$P_PROG[<n>]	STRING	Proporciona el nombre del programa CN que se ejecuta en el nivel de programa seleccionado mediante el índice de matriz <n>. Si el programa CN se ha guardado en el sistema de ficheros pasivo del CN o en la memoria global de programas de pieza, el nombre del programa se suministra en notación CN. Si el programa CN se ejecuta desde una unidad externa diferente a la memoria global de programas de pieza, $P_PROG proporciona el nombre en notación EES.
$P_PROGPATH	STRING	Proporciona la ruta del programa CN que se está ejecutando en ese momento. La llamada de $P_PROGPATH es idéntica a $P_PATH[$P_STACK].

Variable de sistema	Tipo	Descripción	
$P_IS_EES_PATH[<n>]	BOOL	Consulta si la ruta suministrada por $P_PATH[<n>] o el nombre de programa suministrado por $P_PROG[<n>] está en notación CN o en notación EES.	
		= FALSE	$P_PATH[<n>] y $P_PROG[<n>] devuelven notación CN. Es decir, cada identificador tiene el prefijo "_N_". El símbolo de separación para la identificación de fichero es "_". Ejemplos: • Ruta en notación CN: "/_N_WKS_DIR/_N_MYWPD_WPD/" • Nombre de programa en notación CN:"_N_MYPROG_MPF" Una ruta en notación CN puede hacer referencia tanto al sistema de ficheros pasivo en el CN como a la memoria global de programas de pieza.
		= TRUE	$P_PATH[<n>] y $P_PROG[<n>] devuelven notación EES. Es decir, los identificadores no tienen el prefijo "_N_". El símbolo de separación para la identificación de fichero es ".". Ejemplos: • Ruta en notación EES: "//DEV1:/WKS.DIR/MYWPD.WPD/" • Nombre de programa en notación EES: "MYPROG.MPF"

<n>: El índice <n> define el nivel de programa desde el que debe leerse la información de ruta (rango: 0 ... 17)

Nota

En el funcionamiento EES, las variables de sistema $P_PROG, $P_PATH y $P_PROGPATH suministran nombres de ruta en notación EES fuera de la memoria global de programas de pieza (GDIR). Por lo tanto, los programas de usuario que evalúan y siguen procesando estos nombres de ruta deben ampliarse para el modo EES de forma que también puedan procesar nombres de ruta en notación EES.

3.4.2 Memoria de trabajo (CHANDATA, COMPLETE, INITIAL)

Función

La memoria de trabajo contiene los datos actuales del sistema y el usuario con los que se utiliza el control (sistema de ficheros activo), p. ej.:

- Datos de máquina activos
- Datos de corrección de herramienta

- Decalajes de origen
- ...

Programas de inicialización

Los programas de inicialización realizan la asignación previa (inicial) de datos en la memoria de trabajo. Para ello se pueden utilizar los siguientes tipos de fichero:

Tipo de fichero	Descripción
name_TEA	Datos de máquina
name_SEA	Datos del operador
name_TOA	Correcciones de herramientas
name_UFR	Decalajes de origen/frames
name_INI	Fichero de inicialización
name_GUD	Datos globales del usuario
name_RPA	Parámetros R

Áreas de datos

Los datos se pueden asociar a distintas áreas para las cuales sean válidos. Por ejemplo, un control puede disponer de varios canales o, habitualmente, también de varios ejes.

Existen las siguientes áreas de datos:

identificación	Áreas de datos
CN	Datos específicos de CN
CH<n>	Datos específicos de canal (<n> indica el número de canal)
AX<n>	Datos específicos de eje (<n> indica el número del eje de máquina)
TO	Datos de herramienta
COMPLETE	Todos los datos

Generar el programa de inicialización en un PC externo

Utilizando identificadores de área y de tipo de datos, se pueden definir las áreas que deben considerarse como una unidad al almacenar los datos.

_N_AX5_TEA_INI	Datos de máquina para el eje 5
_N_CH2_UFR_INI	Frames del canal no 2
_N_COMPLETE_TEA_INI	Todos los datos de máquina

Tras realizar la puesta en marcha del control numérico quedan definidos una serie de datos que permiten el correcto funcionamiento del control.

Forma de proceder en controles de varios canales (CHANDATA)

CHANDATA(<Número de canal>) para varios canales sólo se permite en el fichero _N_INITIAL_INI. Es el fichero de puesta en marcha con el que se inicializan todos los datos del control.

Código de programa	Comentarios
%_N_INITIAL_INI	
CHANDATA(1)	
	; Asignación de ejes de máquina al canal 1:
$MC_AXCONF_MACHAX_USED[0] = 1.	
$MC_AXCONF_MACHAX_USED[1] = 2.	
$MC_AXCONF_MACHAX_USED[2] = 3.	
CHANDATA(2)	
	; Asignación de ejes de máquina al canal 2:
$MC_AXCONF_MACHAX_USED[0] = 4.	
$MC_AXCONF_MACHAX_USED[1] = 5.	
CHANDATA(1)	
	; Datos de máquina por eje:
	; Ventana de parada exacta basta:
$MA_STOP_LIMIT_COARSE[AX1]=0.2	; Eje 1
$MA_STOP_LIMIT_COARSE[AX2]=0.2	; Eje 2
	; Ventana de parada exacta fina:
$MA_STOP_LIMIT_FINE[AX1]=0.01	; Eje 1
$MA_STOP_LIMIT_FINE[AX1]=0.01	; Eje 2

ATENCIÓN

Instrucción CHANDATA

En el programa de pieza, la instrucción CHANDATA puede definirse solo para el canal en el que se ejecuta el programa CN. Es decir, la instrucción puede utilizarse para evitar que los programas CN se ejecuten en un canal no previsto.

En caso de error, se interrumpe la ejecución del programa.

Nota

Los ficheros INI de listas de trabajos no reciben ninguna instrucción CHANDATA.

Guardar programas de inicialización (COMPLETE, INITIAL)

Los archivos de la memoria de trabajo pueden guardarse en un PC externo y leerse de nuevo desde allí.

- Los datos se guardan con COMPLETE.
- Con INITIAL se crea, a través de todas las áreas, un fichero INI (_N_INITIAL_INI).

Leer programas de inicialización

ATENCIÓN
Pérdida de datos Si se lee el fichero con el nombre "INITIAL_INI", todos los datos no alimentados en el fichero se inicializan con datos estándar. Tan sólo se exceptúan los datos de máquina. Por tanto, se alimentan **datos de operador, datos de herramienta, DO, valores GUD...** con datos estándar (normalmente "CERO").

Para leer datos de máquina individuales es adecuado, p. ej., el fichero COMPLETE_TEA_INI. En este fichero, el control sólo espera datos de máquina. De esta forma no se modifican las demás áreas de datos en este caso.

Cargar programas de inicialización

Cuando solamente se utilizan datos por un canal, se pueden utilizar programas del tipo "INI" como programas de pieza y por lo tanto se pueden seleccionar y ejecutar. Así también se pueden inicializar datos de forma controlada por programa.

3.5 Manejo de ficheros

3.5.1 Escribir fichero (WRITE)

El comando `WRITE` permite escribir secuencias/datos del programa CN al final de un fichero existente en el sistema de ficheros pasivo o en una memoria de programas externa (fichero de protocolo). También puede tratarse del programa que actualmente se está ejecutando.

Nota

Si no existe en la memoria de programas, se crea el fichero que se quiere escribir con el comando `WRITE`.

Requisitos

El nivel de protección ajustado actualmente debe ser igual o superior al derecho WRITE del fichero. Si no es el caso, se deniega el acceso con aviso de error (valor de retorno de las variables de error = 13).

Sintaxis

```
DEF INT <Error>
...
WRITE(<Error>,"<Nombre de fichero>"/"<EqExt>","<Secuencia/datos>")
```

Descripción

`WRITE`:	Comando para insertar una secuencia o datos al final del fichero indicado		
`<Error>`:	**Parámetro 1:** Variable para la devolución del valor erróneo		
	Tipo:	INT	
	Valor:	0	Sin errores
		1	Ruta no autorizada
		2	Ruta no encontrada
		3	Fichero no encontrado
		4	Tipo de fichero incorrecto
		10	Fichero lleno
		11	El fichero está siendo utilizado
		12	No hay recursos libres
		13	Acceso no autorizado
		14	EXTOPEN inexistente o fallido para el equipo de salida
		15	Error al escribir en equipo externo
		16	Ruta externa programada no válida

<Nombre de fichero>:	**Parámetro 2:** Nombre del fichero en el que se deben insertar la secuencia indicada o los datos indicados	
	Tipo:	STRING
	La ruta absoluta puede indicarse antes del nombre del fichero propiamente dicho. Sin indicación de ruta, el fichero se busca en el directorio actual (= directorio del programa seleccionado). Reglas sobre la indicación de ruta, ver "Direccionamiento de ficheros de la memoria de programas (Página 570)".	
<EqExt>:	Si deben emitirse datos a un equipo/fichero externo a través de la función "Process DataShare", en lugar del nombre de fichero debe indicarse el identificador simbólico del equipo/fichero externo que se desea abrir.	
	Tipo:	STRING
	Para más información, ver "Process DataShare - Salida a un equipo/fichero externo (EXTOPEN, WRITE, EXTCLOSE): (Página 1031)". **Nota:** El identificador debe ser idéntico al indicado en el comando EXTOPEN.	
<Secuencia/ datos>:	**Parámetro 3:** Secuencia o datos que se deben insertar en el fichero indicado.	
	Tipo:	STRING

Nota

Al escribir en el sistema de ficheros pasivo o en una memoria de programas externa, el comando WRITE añade de manera implícita un carácter "LF" (LINE FEED = salto de línea) al final de la cadena de salida.

Este comportamiento no es válido para la salida a un equipo/fichero externo a través de la función "Process DataShare". Si debe añadirse un carácter "LF", debe indicarse explícitamente en la cadena de salida.

→ Ver al respecto el ejemplo 3: "LF" implícito/explícito.

Condiciones

- **Tamaño máximo de fichero (→ fabricante de la máquina)**
  El tamaño máximo posible de ficheros de protocolo del sistema de ficheros pasivo se ajusta con el dato de máquina:
  MD11420 $MN_LEN_PROTOCOL_FILE
  El tamaño máximo de fichero es válido para todos los ficheros creados en el sistema de ficheros pasivo con el comando WRITE. Si se supera, se emite un aviso de error y la secuencia o los datos no se guardan. Si la memoria tiene suficiente capacidad, se puede crear un nuevo fichero.

Ejemplos

Ejemplo 1: comando WRITE en el sistema de ficheros pasivo sin indicación de ruta absoluta

Código de programa	Comentarios
N10 DEF INT ERROR	; Definición de las variables de error.

Código de programa	Comentarios
N20 WRITE(ERROR,"PROT","PROTOCOLO DEL 7.2.97")	; Escribir el texto "PROTOCOLO DEL 7.2.97" en el fichero _N_PROT_MPF.
N30 IF ERROR	; Evaluación de errores.
N40 MSG ("Error en comando WRITE:" << ERROR)	
N50 M0	
N60 ENDIF	
...	

Ejemplo 2: comando WRITE en el sistema de ficheros pasivo con indicación de ruta absoluta

```
Código de programa
...
WRITE(ERROR,"/_N_WKS_DIR/_N_PROT_WPD/_N_PROT_MPF","PROTOCOLO DEL
7.2.97")
...
```

Ejemplo 3: "LF" implícito/explícito

a) Escribir en el sistema de ficheros pasivo con un "LF" generado de manera implícita

```
Código de programa
...
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
```

Resultado de salida:

MY_STRING

MY_STRING

b) Escribir en un fichero externo sin "LF" generado de manera implícita

```
Código de programa
...
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)
N270 M30
```

Resultado de salida:

MY_STRINGMY_STRING

c) Escribir en un fichero externo con un "LF" programado explícitamente

Para obtener el mismo resultado que en a), debe programarse lo siguiente:

Código de programa

```
...
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
```

Resultado de salida:

MY_STRING

MY_STRING

3.5.2 Borrar fichero (DELETE)

Con el comando `DELETE` se pueden borrar todos los ficheros, independientemente de si han sido creados o no con el comando `WRITE`. También los ficheros creados con un nivel de acceso superior se pueden borrar con `DELETE`.

Sintaxis

```
DEF INT <Error>
DELETE(<Error>,"<Nombre de fichero>")
```

Descripción

DELETE:	Comando para borrar el fichero indicado		
<Error>:	Variable para la devolución del valor erróneo		
	Tipo:	INT	
	Valor:	0	Sin errores
		1	Ruta no autorizada
		2	Ruta no encontrada
		3	Fichero no encontrado
		4	Tipo de fichero incorrecto
		11	El fichero está siendo utilizado
		12	No hay recursos libres
		20	Otros errores
<Nombre de fichero>:	Nombre del fichero que se desea borrar		
	Tipo:	STRING	
	La ruta absoluta puede indicarse antes del nombre del fichero propiamente dicho. Sin indicación de ruta, el fichero se busca en el directorio actual (= directorio del programa seleccionado). Reglas sobre la indicación de ruta, ver "Direccionamiento de ficheros de la memoria de programas (Página 570)".		

Ejemplo

Código de programa	**Comentarios**
N10 DEF INT ERROR	; Definición de las variables de error.
N15 STOPRE	; Parada de decodificación previa.
N20 DELETE(ERROR,"/_N_SPF_DIR/_N_TEST1_SPF")	; Borrar el fichero TEST1 en el directorio del subprograma.
N30 IF ERROR	; Evaluación de errores.
N40 MSG("Error en comando DELETE:" <<ERROR)	
N50 M0	
N60 ENDIF	

3.5.3 Leer líneas en fichero (READ)

El comando READ lee una o varias líneas del fichero indicado y guarda la información leída en una matriz del tipo STRING. Cada línea leída ocupa un elemento de matriz en este campo.

Requisitos

El nivel de protección ajustado actualmente debe ser igual o superior al derecho READ del fichero. Si no es el caso, se deniega el acceso con aviso de error (valor de retorno de las variables de error = 13).

Sintaxis

```
DEF INT <Error>
DEF STRING[<Longitud del string>] <Resultado>[<n>,<m>]
READ(<Error>,"<Nombre de fichero>",<Línea inicial>,<Número de
líneas>,<Resultado>)
```

Descripción

READ:	Comando para leer líneas del fichero indicado y para guardar estas líneas en un elemento de matriz.		
<Error>:	Variable para la devolución del valor erróneo (parámetro Call-By-Reference)		
	Tipo:	INT	
	Valor:	0	Sin errores
		1	Ruta no autorizada
		2	Ruta no encontrada
		3	Fichero no encontrado
		4	Tipo de fichero incorrecto
		11	El fichero está siendo utilizado
		13	Derechos de acceso insuficientes
		21	Línea no presente (parámetro <Línea inicial> o <Número de líneas> mayor que el número de líneas en el fichero indicado).
		22	La longitud de matriz de las variables de resultado (<Resultado>) es demasiado pequeña.
		23	Rango de líneas demasiado grande (se ha seleccionado un parámetro <Número de líneas> tan grande que se lee más allá del final del fichero).
<Nombre de fichero>:	Nombre del fichero que se va a leer (parámetro Call-By-Value)		
	Tipo:	STRING	
	La ruta absoluta puede indicarse antes del nombre del fichero propiamente dicho. Sin indicación de ruta, el fichero se busca en el directorio actual (= directorio del programa seleccionado). Reglas sobre la indicación de ruta, ver "Direccionamiento de ficheros de la memoria de programas (Página 570)".		
<Línea inicial>:	Línea inicial del área del fichero que se va a leer (parámetro Call-By-Value)		
	Tipo:	INT	
	Valor:	0	Se lee la cantidad de líneas indicada con el parámetro <Número de líneas> antes del final del fichero.
		1 … n	Número de la primera línea que se va a leer.
<Número de líneas>:	Número de líneas que se van a leer (parámetro Call-By-Value)		
	Tipo:	INT	

<Resultado>:	Variable de resultado (parámetro Call-By-Reference) Elemento de matriz en el que se guarda el texto leído.	
	Tipo:	STRING (longitud máx.: 255)
	Si en el parámetro <Número de líneas> están indicadas menos líneas que el tamaño [<n>,<m>] de la matriz de las variables de resultado, el resto de elementos de matriz no se modificará. La terminación de una línea con el carácter de control "LF" (Line Feed) o "CR LF" (Carriage Return Line Feed) **no** se guarda en la variable de resultado. Las líneas leídas se cortan cuando la línea es más larga que la longitud de cadena definida. No se señaliza ningún aviso de error.	

Nota

No se pueden leer ficheros binarios. Se emite el error "Tipo de fichero incorrecto" (valor de retorno de las variables de error = 4). Los siguientes tipos de fichero no se pueden leer: _BIN, _EXE, _OBJ, _LIB, _BOT, _TRC, _ACC, _CYC, _NCK.

Ejemplo

Código de programa	**Comentarios**
N10 DEF INT ERROR	; Definición de las variables de error.
N20 DEF STRING[255] RESULT[5]	; Definición de las variables de resultado.
N30 READ(ERROR,"/_N_CST_DIR/_N_TESTFILE_MPF",1,5,RESULT)	; Nombre del fichero con identificación de dominio y de fichero y ruta de acceso.
N40 IF ERROR <>0	; Evaluación de errores.
N50 MSG("ERROR"<<ERROR<<"CON COMANDO READ")	
N60 M0	
N70 ENDIF	
...	

3.5.4 Comprobar la presencia de un fichero (ISFILE)

Con el comando ISFILE se verifica si un fichero está presente en la memoria de programas.

Sintaxis

```
<Resultado>=ISFILE("<nombre de fichero>")
```

Descripción

ISFILE:	Comando para verificar la presencia de un fichero		
<Nombre de fichero>:	Nombre del fichero cuya presencia se desea verificar.		
	Tipo:	STRING	
	La ruta absoluta puede indicarse antes del nombre del fichero propiamente dicho. Sin indicación de ruta, el fichero se busca en el directorio actual (= directorio del programa seleccionado). Reglas sobre la indicación de ruta, ver "Direccionamiento de ficheros de la memoria de programas (Página 570)".		
<Resultado>:	Variable de resultado para el registro del resultado de prueba		
	Tipo:	BOOL	
	Valor:	TRUE	Fichero presente
		FALSE	Fichero no presente

Ejemplos

Ejemplo 1

```
Código de programa                    Comentarios
N10 DEF BOOL RESULT                   ; Definición de las variables de resultado.
N20 RESULT=ISFILE("TESTFILE")
N30 IF(RESULT==FALSE)
N40    MSG("FICHERO NO PRESENTE")
N50    M0
N60 ENDIF
...
```

Ejemplo 2

```
Código de programa                    Comentarios
N10 DEF BOOL RESULT                   ; Definición de las variables de resultado.
N20 RESULT=ISFILE("TESTFILE")
N30 IF(NOT ISFILE("TESTFILE"))
N40    MSG("FICHERO NO PRESENTE")
N50    M0
N60 ENDIF
...
```

3.5.5 Leer información de fichero (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO)

Mediante los comandos FILEDATE, FILETIME, FILESIZE, FILESTAT y FILEINFO se pueden leer determinadas informaciones de fichero como la fecha o la hora del último acceso de escritura, el tamaño de fichero actual, el estado del fichero o la suma de estas informaciones.

Requisitos

El nivel de protección ajustado actualmente debe ser igual o superior al derecho Show del directorio superior. Si no es el caso, se deniega el acceso con aviso de error (valor de retorno de las variables de error = 13).

Sintaxis

```
FILE....(<error>,"<nombre de fichero>",<resultado>)
```

Descripción

FILEDATE:	Indica la **fecha** del último acceso de escritura a un fichero		
FILETIME:	Indica la **hora** del último acceso de escritura a un fichero		
FILESIZE:	Suministra el **tamaño actual** de un fichero		
FILESTAT:	Proporciona el **estado** de un fichero en relación con los siguientes **derechos**: • Lectura (r: read) • Escritura (w: write) • Ejecución (x: execute) • Visualizar (s: show) • Borrar (d: delete) **Nota:** Estos niveles de protección son propiedades especiales del sistema de ficheros pasivo. Por lo tanto, al acceder a la memoria de programas externa, FILESTAT solo proporciona derechos de acceso ajustados por defecto (77777).		
FILEINFO:	Proporciona la **suma de las informaciones** que se pueden leer a través de FILEDATE, FILETIME, FILESIZE y FILESTAT para un fichero		
<Error>:	Variable para la devolución del valor erróneo (parámetro Call-By-Reference)		
	Tipo:	VAR INT	
	Valor:	0	Sin errores
		1	Ruta no autorizada
		2	Ruta no encontrada
		3	Fichero no encontrado
		4	Tipo de fichero incorrecto
		13	Derechos de acceso insuficientes
		22	La longitud de cadena de las variables de resultado (<Resultado>) es demasiado pequeña.
<Nombre de fichero>:	Nombre del fichero del que deben leerse las informaciones de fichero		
	Tipo:	CHAR[160]	
	La ruta absoluta puede indicarse antes del nombre del fichero propiamente dicho. Sin indicación de ruta, el fichero se busca en el directorio actual (= directorio del programa seleccionado). Reglas sobre la indicación de ruta, ver "Direccionamiento de ficheros de la memoria de programas (Página 570)".		

<Resultado>:	Variable de resultado (parámetro Call-By-Reference) Variable en la que se guarda la información de fichero solicitada.			
	Tipo:	VAR CHAR[8]	con	FILEDATE Formato: "dd.mm.aa"
		VAR CHAR[8]	con	FILETIME Formato: "hh:mm.ss"
		VAR INT	con	FILESIZE El tamaño del fichero se indica en bytes.
		VAR CHAR[5]	con	FILESTAT Formato: "rwxsd" (r: **r**ead, w: **w**rite, x: e**x**ecute, s: **s**how, d: **d**elete)
		VAR CHAR[32]	con	FILEINFO Formato: "rwxsd nnnnnnnn dd.mm.yy hh:mm:ss"

Ejemplo

Código de programa	**Comentarios**
N10 DEF INT ERROR	; Definición de las variables de error.
N20 STRING[32] RESULT	; Definición de las variables de resultado.
N30 FILEINFO(ERROR,"/_N_MPF_DIR/_N_TESTFILE_MPF",RESULT)	; Nombre del fichero con identificación de dominio y de fichero y ruta de acceso.
N40 IF ERROR <> 0	; Evaluación de errores
N50 MSG("ERROR"<<ERROR<<"CON COMANDO FILEINFO")	
N60 M0	
N70 ENDIF	
...	

En la variable de resultado RESULT, este ejemplo podría ofrecer, p. ej., el siguiente resultado:

"77777 12345678 26.05.00 13:51:30"

3.6 Zonas protegidas

3.6.1 Definición de las zonas protegidas (CPROTDEF, NPROTDEF)

Cada zona protegida que deba proteger elementos de la máquina contra colisiones se define en el programa de pieza en bloques. Estos incluyen los siguientes elementos:

1. Definición del plano de trabajo
   Antes de la definición propiamente dicha de la zona protegida, debe seleccionarse el plano de trabajo al que debe referirse la descripción de contorno de la zona protegida.
2. Inicio de la definición
   Dependiendo del comando CN, se crea una zona protegida específica del canal, o bien específica de la máquina.
3. Descripción del contorno de la zona protegida
   El contorno de una zona protegida se describe mediante desplazamientos. Estos no se ejecutan y no tienen ninguna conexión con las descripciones geométricas anteriores o posteriores. Definen exclusivamente la zona protegida.
4. Final de la definición

Sintaxis

```
DEF INT <Var>
G17/G18/G19
CPROTDEF/NPROTDEF(<n>,<t>,<LimAp>,<ApMás>,<ApMenos>)
G0/G1/... X/Y/Z...
...
EXECUTE(<Var>)
```

Descripción

`DEF INT <Var>`:	Definición de una variable auxiliar local del tipo de datos INTEGER	
`<Var>`:	Nombre de la variable auxiliar	
`G17/G18/G19`:	Plano de trabajo **Nota:** El plano de trabajo no puede ser modificado antes del final de la definición. No se permite una programación de la aplicada entre el inicio y el final de la definición.	
`CPROTDEF()`:	Procedimiento predefinido para la definición de una zona protegida específica del **canal**	
`NPROTDEF()`:	Procedimiento predefinido para la definición de una zona protegida específica de la **máquina**	
`<n>`:	Número de la zona protegida definida	
	Tipo de dato:	INT

<t>:	Tipo de zona protegida		
	Tipo de dato:	BOOL	
	Valor:	TRUE	Zona protegida relativa a la **herramienta**
		FALSE	Zona protegida relativa a la **pieza**
<LimAp>:	Tipo de limitación en la 3ª dimensión		
	Tipo de dato:	INT	
	Valor:	0	Sin limitaciones
		1	Límite en sentido positivo
		2	Límite en sentido negativo
		3	Límite en sentido positivo y negativo
<ApMás>:	Valor del límite en sentido positivo de la 3.ª dimensión		
	Tipo de dato:	REAL	
<ApMenos>:	Valor del límite en sentido negativo de la 3.ª dimensión		
	Tipo de dato:	REAL	
G0/G1/... X/Y/Z... ...:	El contorno de una zona protegida se indica con máx. 11 desplazamientos en el plano de trabajo seleccionado. El primer desplazamiento corresponde al movimiento en el contorno. El último punto de la descripción de contorno tiene que coincidir siempre con el primer punto de la descripción de contorno. Se considera zona protegida el área situada a la izquierda del contorno: • Zona protegida interior El contorno de una zona protegida interior debe describirse en sentido antihorario. • Zona protegida exterior (¡solo permitida para zonas protegidas relativas a la pieza!) El contorno de una zona protegida exterior debe describirse en sentido horario. Se admiten los siguientes elementos de contorno: • G0, G1 para elementos de contorno rectos • G2 para segmentos circulares en sentido horario ¡Solo permitida para zonas protegidas relativas a la pieza! No está permitida en zonas protegidas relativas a la herramienta ya que estas solo pueden ser convexas. • G3 para segmentos circulares en sentido antihorario **Nota:** No puede definirse una zona protegida mediante una circunferencia. Una circunferencia debe estar dividida en dos arcos. **Nota:** ¡No se admite la secuencia G2 → G3 o G3 → G2! Entre las dos secuencias circulares debe insertarse una breve secuencia G1.		
EXECUTE(<Var>):	Procedimiento predefinido que marca el final de la definición Con EXECUTE se vuelve a conmutar a la ejecución normal del programa.		

Ejemplo

Ver ejemplo en "Activar/desactivar zonas protegidas (CPROT, NPROT) (Página 595)".

Información adicional

Zonas protegidas específicas de la máquina

Una zona protegida específica de la máquina o su contorno se definen mediante ejes geométricos, es decir, referidos al sistema de coordenadas básico (BKS) de un canal. Para que pueda efectuarse una correcta vigilancia de la zona protegida en todos los canales en los que está activada la zona protegida específica de la máquina, el sistema de coordenadas básico (BKS) de todos los canales afectados debe ser idéntico:

- Posición del origen de coordenadas respecto al origen de la máquina
- Orientación de los ejes de coordenadas

Punto de referencia de la descripción del contorno

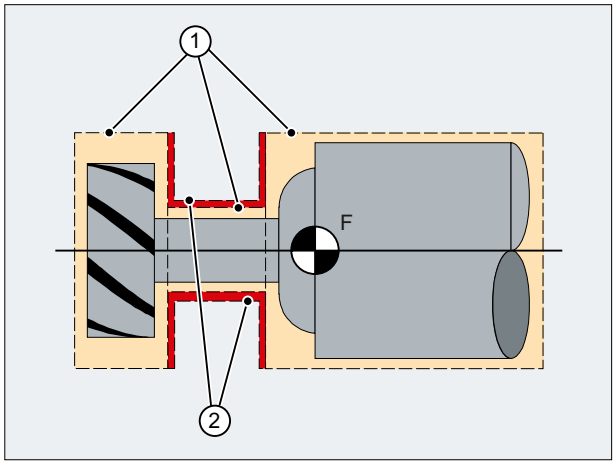
- Zonas protegidas relativas a la herramienta
  Las coordenadas para las zonas protegidas relativas a la **herramienta** deben indicarse de forma absoluta respecto al **punto de referencia del portaherramientas F**.
- Zonas protegidas relativas a la pieza
  Las coordenadas para las zonas protegidas relativas a la **pieza** deben indicarse de forma absoluta respecto al origen del **sistema de coordenadas básico (BKS)**.

Zonas protegidas con simetría de rotación

En zonas protegidas con simetría de rotación (p. ej., mandril del cabezal) se tiene que describir el contorno completo, no solo hasta el centro de giro.

Zonas protegidas relativas a la herramienta

Las zonas protegidas relativas a la herramienta siempre tienen que ser convexas. Si se desea una zona protegida cóncava, se tiene que dividir en varias zonas protegidas convexas.



① Zonas protegidas convexas
② Zonas protegidas cóncavas (**¡no permitidas!**)
F Punto de referencia del portaherramientas

Condiciones

Durante la definición de una zona protegida no pueden estar activas ni utilizarse las siguientes funciones:

- Corrección del radio de herramienta (corrección del radio de la fresa, corrección del lado de filo)
- Transformación
- Búsqueda del punto de referencia (`G74`)
- Desplazamiento a punto fijo (`G75`)
- Tiempo de espera (`G4`)
- Parada de búsqueda de secuencia (`STOPRE`)
- Fin del programa (`M17`, `M30`)
- Funciones M: `M0`, `M1`, `M2`

3.6.2 Activar/desactivar zonas protegidas (CPROT, NPROT)

Las zonas protegidas definidas anteriormente en el programa de pieza se pueden activar en cualquier momento o preactivarse para una activación posterior mediante el programa de usuario PLC. Las zonas protegidas activas pueden volver a desactivarse en cualquier momento.

En la activación o preactivación existe además la posibilidad de desplazar de forma relativa el punto de referencia de la zona protegida.

Nota

Una zona protegida solo se tiene en cuenta después del referenciado de todos los ejes geométricos del canal en el que se ha activado.

Nota

Vigilancia de las zonas protegidas

Si no está activa ninguna zona protegida relativa a la herramienta, la trayectoria de la herramienta se comprueba frente a las zonas protegidas relativas a la pieza.

Si no está activa ninguna zona protegida relativa a la pieza, no tiene lugar ninguna vigilancia de zonas protegidas.

Sintaxis

```
CPROT(<n>,<Estado>,<XMov>,<YMov>,<ZMov>)
NPROT(<n>,<Estado>,<XMov>,<YMov>,<ZMov>)
```

Descripción

CPROT:	Procedimiento predefinido para la activación de una zona protegida específica del **canal**		
NPROT:	Procedimiento predefinido para la activación de una zona protegida específica de la **máquina**		
<n>:	Número de la zona protegida		
	Tipo de dato:	INT	
<Estado>:	Con este parámetro se ajusta el estado de activación específico del canal		
	Tipo de dato:	INT	
	Valor:	0	Desactivar zona protegida
		1	Preactivar zona protegida
		2	Activar zona protegida
		3	Preactivar zona protegida con parada condicionada
<XMov>,<YMov>,<ZMov>:	Valores de decalaje aditivos en dirección X/Y/Z El desplazamiento se puede realizar en 1, 2 o 3 dimensiones. Los valores de decalaje se refieren a: • el origen de máquina en zona protegida relativa a la **pieza** • el punto de referencia del portaherramientas F en zona protegida relativa a la **herramienta**		
	Tipo de dato:	REAL	

Ejemplo

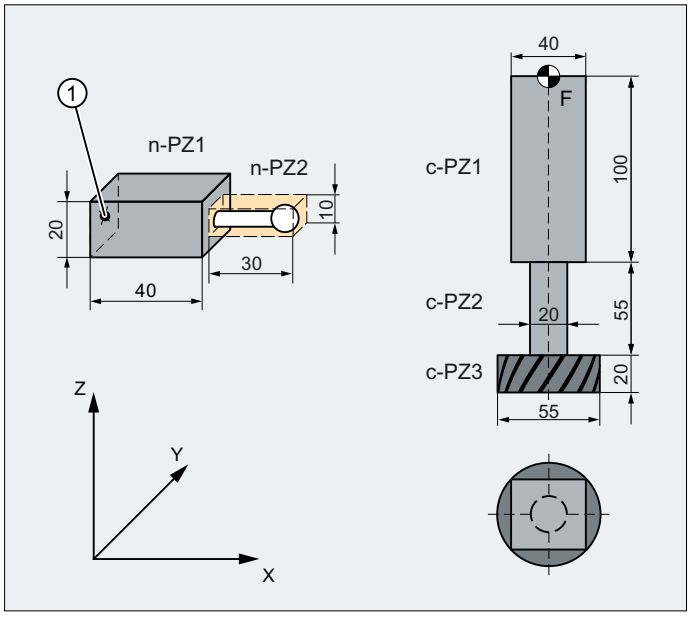
Para una fresadora se quiere vigilar una posible colisión de la fresa con el palpador. La posición del palpador se indicará en la activación mediante un decalaje.

Para este fin se definen las siguientes zonas protegidas:

- Una zona protegida específica de la máquina y otra específica de la pieza para el soporte del palpador (n-PZ1) y para el palpador mismo (n-PZ2).
- Una zona protegida específica del canal y otra específica de la herramienta para el portafresa (c-PZ1), el vástago de la fresa (c-PZ2) y la fresa misma (c-PZ3).

La orientación de todas las zonas protegidas corresponde a la dirección Z.

La posición del punto de referencia del palpador en la activación debe ser X = -120, Y = 60 y Z = 80.



① Punto de referencia para la zona protegida del palpador

F Punto de referencia del portaherramientas

Código de programa	**Comentarios**
DEF INT PROTZONE	; Definición de una variable auxiliar
G17	; Plano de trabajo XY
; Definición de las zonas protegidas:	
NPROTDEF(1,FALSE,3,10,-10)	; Zona protegida n-PZ1
G01 X0 Y-10	
X40	
Y10	
X0	
Y-10	
EXECUTE(PROTZONE)	
NPROTDEF(2,FALSE,3,5,-5)	; Zona protegida n-PZ2
G01 X40 Y-5	
X70	
Y5	
X40	
Y-5	
EXECUTE(PROTZONE)	
CPROTDEF(1,TRUE,3,0,-100)	; Zona protegida c-PZ1
G01 X-20 Y-20	
X20	
Y20	
X-20	
Y-20	
EXECUTE(PROTZONE)	

```
Código de programa                          Comentarios
CPROTDEF(2,TRUE,3,-100,-150)                ; Zona protegida c-PZ2
G01 X0 Y-10
G03 X0 Y10 J10
X0 Y-10 J-10
EXECUTE(PROTZONE)
CPROTDEF(3,TRUE,3,-150,-170)                ; Zona protegida c-PZ3
G01 X0 Y-27.5
G03 X0 Y27.5 J27.5
X0 Y27.5 J-27.5
EXECUTE(PROTZONE)

; Activación de las zonas protegidas:
NPROT(1,2,-120,60,80)                       ; Activar la zona protegida n-PZ1 con decalaje
NPROT(2,2,-120,60,80)                       ; Activar la zona protegida n-PZ2 con decalaje
CPROT(1,2,0,0,0)                            ; Activar la zona protegida c-PZ1
CPROT(2,2,0,0,0)                            ; Activar la zona protegida c-PZ2
CPROT(3,2,0,0,0)                            ; Activar la zona protegida c-PZ3
```

Información adicional

Estado de activación después de arrancar el control

Una zona protegida puede estar activa nada más arrancar el control y referenciar los ejes. Este es el caso si para la zona protegida está ajustada en TRUE la siguiente variable de sistema:

- $SN_PA_ACTIV_IMMED[<n>] (para zona protegida específica de la máquina) o
- $SC_PA_ACTIV_IMMED[<n>] (para zona protegida específica del canal)
  El índice "<n>" corresponde al número de la zona protegida: 0 = 1. Zona protegida

La zona protegida se activa con el estado = 2 y sin decalaje.

Activación múltiple de una zona protegida

Una zona protegida específica de la máquina puede ser efectiva simultáneamente en varios canales (p. ej., zona protegida de pinola con dos carros opuestos). La vigilancia de las zonas protegidas solo se lleva a cabo si están referenciados todos los ejes geométricos.

Una zona protegida no se puede activar simultáneamente en un canal con decalajes distintos.

Vigilancia de zonas protegidas con corrección activa de radio de herramienta

Cuando está activa la corrección de radio de herramienta, la vigilancia de zonas protegidas solo funciona si el plano de la corrección del radio de herramienta es idéntico al plano de las definiciones de zona protegida.

3.6.3 Comprobación de vulneración de zonas protegidas, limitación del campo de trabajo y finales de carrera de software (CALCPOSI)

Función

La función CALCPOSI comprueba en el sistema de coordenadas de pieza (WKS), si, partiendo de la posición inicial, los **ejes geométricos** pueden ejecutar un trayecto especificado sin vulnerar limitaciones activas. En caso de que, debido a la existencia de limitaciones, no sea posible ejecutar por completo el trayecto, se devuelve un valor de estado positivo en codificación decimal, así como el trayecto máximo posible.

Definición

```
INT CALCPOSI(VAR REAL[3] <Start>, VAR REAL[3] <Dist>, VAR REAL[5]
<Limit>, VAR REAL[3] <MaxDist>, BOOL <MeasSys>, INT <TestLim>)
```

Sintaxis

```
<Status> = CALCPOSI(VAR <Start>, VAR <Dist>, VAR <Limit>, VAR
<MaxDist>, <MeasSys>, <TestLim>)
```

Descripción

`CALCPOSI(...)`:	Función predefinida para la comprobación de vulneración de límites en referencia a los ejes geométricos	
	Parada de decodificación previa:	no
	Único elemento de la secuencia:	sí

<Estado>: (Parte 1)	Valor de retorno de la función. Los valores negativos indican estados de error.		
	Tipo de dato:		INT
	Rango de valores:		-8 ≤ x ≤ 100000
	Valor:	0	El trayecto puede recorrerse por completo
		-1	En <Límite> existe por lo menos un componente negativo.
		-2	Error durante un cálculo de transformación. Ejemplo: El trayecto atraviesa una singularidad, por lo que las posiciones de los ejes no están definidas.
		-3	El trayecto indicado <Dist> y el máximo trayecto posible <MaxDist> son linealmente dependientes. **Nota** Solo puede producirse en combinación con <TestLim>, bit 4 == 1.
		-4	La proyección del sentido de desplazamiento contenido en <Dist> sobre la superficie límite es el vector nulo, o el sentido de desplazamiento es perpendicular a la superficie límite vulnerada. **Nota** Solo puede producirse en combinación con <TestLim>, bit 5 == 1.
		-5	En <TestLim>, bit 4 == 1 Y Bit 5 == 1
		-6	Está sin referenciar por lo menos un eje de máquina que debe tenerse en cuenta para la comprobación de los límites de desplazamiento.
		-7	Función Prevención de colisiones: Definición no válida de la cadena cinemática o de las zonas protegidas.
		-8	Función Prevención de colisiones: No puede ejecutarse la función por falta de memoria.
<Estado>: (Parte 2)	Unidades **Nota** Si se vulneran varios límites al mismo tiempo, se notificará el que provoque una mayor limitación del trayecto especificado.		
	Valor:	1	El trayecto está limitado por finales de carrera de software
		2	El trayecto está limitado por limitación del campo de trabajo
		3	El trayecto está limitado por zonas protegidas
		4	Función Prevención de colisiones: El trayecto está limitado por zonas protegidas
	Decenas		
	Valor:	1x	El valor inicial viola el límite
		2x	La recta especificada viola el límite. Este valor se devuelve también cuando el punto final propiamente dicho no vulnera ningún límite, pero en el trayecto del punto inicial al punto final sí se vulneraría un límite (p. ej., al atravesar una zona protegida, finales de carrera de software curvados en el WKS en caso de transformadas no lineales, p. ej., Transmit).

<Estado>: (Parte 3)	Centenas		
	Valor:	1xx	Y unidades == 1 o 2: Se vulnera el límite positivo.
			Y unidades == 3 [1]: Se vulnera una zona protegida específica de CN.
		2xx	Y unidades == 1 o 2: Se vulnera el límite negativo.
			Y unidades == 3 [1]: Se ha vulnerado una zona protegida específica de canal.
<Estado>: (Parte 4)	Millares		
	Valor:	1xxx	Y unidades == 1 o 2: Factor por el que se multiplica el número de eje que vulnera el límite. El recuento de los ejes comienza por 1. Referencia: • Final de carrera de software: Ejes de máquina • Limitación del campo de trabajo: Ejes geométricos Y unidades == 3 [1]: Factor por el que se multiplica el número de la zona protegida vulnerada.
<Estado>: (Parte 5)	Centenas de millar		
	Valor:	0xxxxx	Centenas de millar == 0: <Dist> no varía
		1xxxxx	En <Dist> se devuelve un vector de dirección que define el sentido ulterior del movimiento sobre la superficie límite. Solo puede producirse bajo las siguientes condiciones: • Se ha vulnerado un final de carrera de software o una limitación del campo de trabajo (no en el punto inicial) • Una transformada **no** está activa • <TestID>, bit 4 o bit 5 == 1
<Inicio>:	Referencia a un vector con las posiciones iniciales: • <Inicio> [0]: 1er eje geométrico • <Inicio> [1]: 2o eje geométrico • <Inicio> [2]: 3er eje geométrico		
	Tipo de parámetro:		Entrada
	Tipo de dato:		VAR REAL [3]
	Rango de valores:		-máx. Valor REAL ≤ x[<n>] ≤ +máx. valor REAL

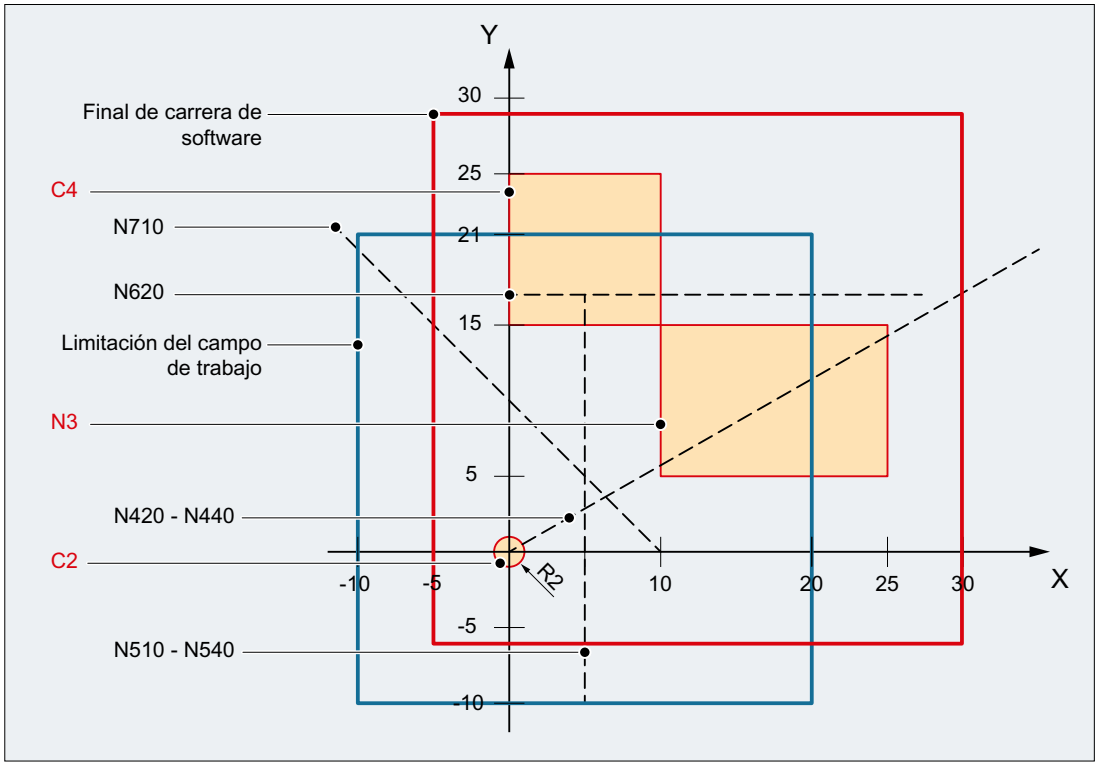
<Dist>:	Referencia a un vector.	
	Entrada: trayecto incremental • <Dist> [0]: 1er eje geométrico • <Dist> [1]: 2o eje geométrico • <Dist> [2]: 3er eje geométrico	
	Salida (solo con centenas de millar definidas en el <Estado>):	
	<Dist> contiene como valor de salida un vector de unidad **v** que define el sentido ulterior de desplazamiento en el WKS.	
	Caso 1: formación del vector **v** para <TestID>, bit 4 == 1 Los vectores de entrada <Dist> y <MaxDist> definen el plano de desplazamiento. Este plano se corta con la superficie límite vulnerada. La línea de intersección de los dos planos define la dirección del vector **v**. La orientación (signo) se escoge de tal modo que el ángulo entre el vector de entrada <MaxDist> y **v** no sea mayor que 90 grados.	
	Caso 2: Formación del vector **v** para <TestID>, bit 5 == 1 El vector **v** es el vector de unidad en sentido de la proyección del vector de desplazamiento contenido en <Dist> sobre la superficie límite. Si la proyección del vector de desplazamiento sobre la superficie límite es el vector nulo, se devuelve un error.	
	Tipo de parámetro:	Entrada/salida
	Tipo de dato:	VAR REAL [3]
	Rango de valores:	-máx. Valor REAL ≤ x[<n>] ≤ +máx. valor REAL
<Límite>:	Referencia a un campo de la longitud 5. • <Límite> [0 - 2]: distancias mínimas de los ejes geométricos a los límites: – <Límite> [0]: 1er eje geométrico – <Límite> [1]: 2o eje geométrico – <Límite> [2]: 3er eje geométrico Las distancias mínimas se mantienen en los siguientes casos: – Limitación del campo de trabajo: Sin limitaciones – Final de carrera de software: Si no hay ninguna transformada activa, o si hay una transformada activa para la que es posible una asignación unívoca de los ejes geométricos a ejes de máquina lineales, p. ej., transformadas de 5 ejes. • <Límite> [3]: Contiene la distancia mínima para los ejes de máquina lineales que no pueden asignarse a ningún eje geométrico, p. ej., a causa de una transformada no lineal. Además, este valor se utiliza como límite para la vigilancia de las zonas protegidas convencionales y de las zonas protegidas de prevención de colisiones. • <Límite> [4]: Contiene la distancia mínima para los ejes de máquina giratorios que no pueden asignarse a ningún eje geométrico, p. ej., a causa de una transformada no lineal. **Nota** Este valor solo es válido para la vigilancia de los finales de carrera de software de transformadas especiales.	
	Tipo de parámetro:	Entrada
	Tipo de dato:	VAR REAL [5]
	Rango de valores:	-máx. valor REAL ≤ x[n] ≤ +máx. valor REAL

<MaxDist>:	Referencia a un vector con un trayecto incremental en el que ningún límite de eje vulnera la distancia mínima especificada en los ejes de máquina afectados: • <Dist> [0]: 1er eje geométrico • <Dist> [1]: 2o eje geométrico • <Dist> [2]: 3er eje geométrico Si el trayecto no está limitado, el contenido de ese parámetro de retorno es igual al contenido de <Dist>.		
	Con <TestID>, bit 4 == 1: <Dist> y <MaxDist> <MaxDist> y <Dist>, como valores de entrada, deben contener vectores que definen un plano de desplazamiento. Ambos vectores deben ser linealmente independientes entre si. <MaxDist> puede tener cualquier valor. Para calcular el sentido del movimiento, ver la descripción de <Dist>.		
	Tipo de parámetro:		Salida
	Tipo de dato:		VAR REAL [3]
	Rango de valores:		-máx. Valor REAL ≤ x[<n>] ≤ +máx. valor REAL
<MeasSys>:	Sistema de medida (pulgada/métrico) para los datos de posición y longitud (**opcional**)		
	Tipo de dato:		BOOL
	Valor:	FALSE (Predeterminado)	Sistema de medida de acuerdo con el comando G actualmente activo del grupo G 13 (G70, G71, G700, G710). **Nota** Con G70 activo y sistema base métrico, o con G71 activo y sistema base en pulgadas, las variables del sistema $AA_IW y $AA_MW se devuelven en el sistema base y deberán convertirse en caso de usarse para CALCPOSI.
		TRUE	Sistema de medida según el sistema básico ajustado: DM52806 $MN_ISO_SCALING_SYSTEM
<TestLim>:	Selección codificada al bit de los límites que deberán vigilarse (**opcional**)		
	Tipo de dato:		INT
	Valor por defecto:		Bit 0, 1, 2, 3, 6, 7 == 1 (207)
	Bit	**Decimal**	**Descripción**
	0	1	Final de carrera de software
	1	2	Limitación del campo de trabajo
	2	4	Zonas protegidas convencionales activadas
	3	8	Zonas protegidas convencionales preactivadas
	4	16	En caso de vulnerarse finales de carrera de software o limitaciones del campo de trabajo, devolver en <Dist> el sentido de desplazamiento de acuerdo con el **caso 1** (ver arriba).
	5	32	En caso de vulnerarse finales de carrera de software o limitaciones del campo de trabajo, devolver en <Dist> el sentido de desplazamiento de acuerdo con el **caso 2** (ver arriba).
	6	64	Zonas protegidas activadas de prevención de colisiones
	7	128	Zonas protegidas preactivadas de prevención de colisiones
	8	256	Pares de zonas protegidas activadas y preactivadas de prevención de colisiones

[1] Si se vulneran varias zonas protegidas, se devuelve la zona protegida que provoca una mayor limitación del trayecto especificado.

Ejemplo

Limitaciones



En el ejemplo se muestran los finales de carrera de software y limitaciones del campo de trabajo efectivos en el plano X-Y y las tres zonas protegidas siguientes:

- C2: zona protegida relativa a la herramienta específica de canal, activa, circular, radio = 2 mm
- C4: zona protegida relativa a la pieza, específica de canal, preactivada, cuadrada, lado = 10 mm
- N3: zona protegida específica de la máquina, activa, rectangular, lados = 10 mm x 15 mm

Programa CN

En el programa CN se definen en primer lugar las zonas protegidas y las limitaciones del campo de trabajo. A continuación se llama la función `CALCPOSI()` con distintas parametrizaciones.

Código de programa

```
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
```

Código de programa

```
; Zona protegida relativa a la herramienta C2
N70 CPROTDEF(2, TRUE, 0)
N80 G17 G1 X-2 Y0
N90 G3 I2 X2
N100 I-2 X-2
N110 EXECUTE(_PA)
; Zona protegida relativa a la pieza C4
N120 CPROTDEF(4, FALSE, 0)
N130 G17 G1 X0 Y15
N140 X10
N150 Y25
N160 X0
N170 Y15
N180 EXECUTE(_PA)
; Zona protegida específica de la máquina N3
N190 NPROTDEF(3, FALSE, 0)
N200 G17 G1 X10 Y5
N210 X25
N220 Y15
N230 X10
N240 Y5
N250 EXECUTE(_PA)
; Activar o preactivar zonas protegidas
N260 CPROT(2, 2, 0, 0, 0)
N270 CPROT(4, 1, 0, 0, 0)
N280 NPROT(3, 2, 0, 0, 0)
; Definir limitaciones de campo de trabajo
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.
```

Código de programa

```
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)
; Eliminar los frames de N690 y N700
N720 TRANS
N730 _START[0] = 0.
N740 _START[1] = 10.
N750 _START[2] = 0.
; Los vectores _DIST y _MAXDIST definen el plano de desplazamiento
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
```

Resultados de CALCPOSI()

N...	<Estado>	<MaxDist>[0] ≙ X	<MaxDist>[1] ≙ Y	Notas
420	3123	8.040	4.594	Se vulnera N3.
430	1122	20.000	11.429	Ninguna vigilancia de las zonas protegidas, se vulnera la limitación de campo de trabajo
440	1121	30.000	17.143	Solo permanece activa la vigilancia de los finales de carrera de software.
510	4213	0.000	0.000	Punto inicial vulnerado C4
520	0000	0.000	-27.000	No se vigila la zona protegida preactivada C4. Puede recorrerse por completo el trayecto especificado.
540	2222	0.000	-25.000	Debido a _LIMIT[1] = 2, el trayecto está restringido por la limitación del campo de trabajo.

N...	<Estado>	<MaxDist>[0] ≙ X	<MaxDist>[1] ≙ Y	Notas
620	4223	-13.000	0.000	Distancia total respecto a C4: 4 mm debido a C2 y _LIMIT[3]. La distancia de C2 → N3 de 0,1 mm no provoca limitación del trayecto.
710	1221	0.000	21.213	Frame activo con traslación y rotación. El trayecto admisible en _DIST es válido en el WKS desplazado y girado.
820	102121	18.000	18.000	Se vulnera el final de carrera de software del eje Y. Con <_TESTLIM> = 17 se solicita el cálculo de una dirección de trayecto ulterior. Esta dirección está especificada en <_DIST> (0.707, 0.0, 0.707). Es válida porque en <_STATUS> se especifican las centenas de millar.

Información adicional

Estado de eje "referenciado"

Todos los ejes de máquina considerados por `CALCPOSI()` deben estar referenciados.

Indicaciones de desplazamiento referidas a la circunferencia

Todas las indicaciones de desplazamiento referidas a la circunferencia se interpretan **siempre** como indicaciones de radio. Esto debe tenerse en cuenta especialmente en el caso de los ejes de refrentado con programación por diámetro activada (`DIAMON`/`DIAM90`).

Reducción del trayecto

Si se limita el trayecto especificado de un eje, en el valor de retorno `<MaxDist>` se reducen también proporcionalmente los trayectos de los demás ejes. De este modo, el punto final resultante sigue hallándose en el trayecto especificado.

Ejes giratorios

Los ejes giratorios solo se vigilan si no se trata de ejes giratorios con módulo.

Es admisible que para uno o varios de los ejes que intervienen no se hayan definido finales de carrera de software, limitaciones de campo de trabajo o zonas protegidas.

Estado de los finales de carrera de software y de la limitación del campo de trabajo

Los finales de carrera de software y las limitaciones del campo de trabajo solo se tienen en cuenta si están activos durante la ejecución de `CALCPOSI()`. Puede influirse en el estado, p. ej., mediante:

- Datos de máquina: DM21020 $MC_WORKAREA_WITH_TOOL_RADIUS
- Datos de operador: $AC_WORKAREA_CS_...
- Señales de la interfaz CN/PLC: DB31, ... DBX12.2 / 3
- Comandos: `WALIMON` / `WALIMOF`

Finales de carrera de software y transformadas

En el caso de algunas transformadas cinemáticas (p. ej., TRANSMIT) y debido a ambigüedades en determinados puntos de la zona de desplazamiento, con CALCPOSI() no es posible determinar siempre de forma unívoca las posiciones de los ejes de máquina (MKS) a partir de las posiciones de los ejes geométricos (WKS). En los desplazamientos normales, la univocidad suele derivarse de los antecedentes y de la condición según la cual un desplazamiento continuo en el WKS debe corresponder siempre a un desplazamiento continuo de los ejes de máquina en el MKS. En esos casos, durante la vigilancia de los finales de carrera de software, para deshacer la ambigüedad se utiliza la posición de la máquina en el momento de ejecución de CALCPOSI().

Nota

Parada de decodificación previa

Si se utiliza CALCPOSI() en combinación con transformadas, es responsabilidad exclusiva del usuario programar una parada de decodificación previa (STOPRE) para sincronizar las posiciones de la máquina con el avance antes de CALCPOSI().

Distancia a la zona protegida y zonas protegidas convencionales

En el caso de las zonas protegidas convencionales **no** se garantiza que, en caso de desplazamiento por el trayecto especificado, se mantenga la distancia de seguridad indicada en <Límite>[3] respecto a todas las zonas protegidas. Solo se garantiza que no se vulnerará ninguna zona protegida en caso de prolongación del punto final devuelto en <Dist> por un valor equivalente a la distancia de seguridad en el sentido de desplazamiento. Sin embargo, la trayectoria de la recta puede pasar tan cerca de una zona protegida como se desee.

Distancia a las zonas protegidas y zonas protegidas de prevención de colisiones

Para las zonas protegidas de prevención de colisiones se garantiza que, en caso de desplazamiento por el trayecto especificado, se mantendrá la distancia de seguridad indicada en el parámetro <Límite>[3] respecto a todas las zonas protegidas.

La distancia de seguridad especificada en el parámetro <Límite>[3] solo tendrá efecto si se cumple:

<Límite>[3] > (DM10619 $MN_COLLISION_TOLERANCE)

Si en el parámetro <TestLim> está ajustado el bit 4 (cálculo del sentido de desplazamiento ulterior), el vector de dirección contenido en <DIST> solo será válido si están definidas las centenas de millar en el valor de retorno de la función (<Estado>). Si no es posible determinar el sentido porque se han vulnerado zonas protegidas o porque hay una transformada activa, el valor de entrada de <DIST> no varía. No se realiza ningún otro aviso de error.

3.7 Órdenes de desplazamiento especiales

3.7.1 Desplazamiento a posiciones codificadas (CAC, CIC, CDC, CACP, CACN)

Con los comandos de desplazamiento para "Desplazamiento a posiciones codificadas", es posible desplazar ejes lineales y giratorios a posiciones de eje fijas guardadas en tablas de datos de máquina indicando números de posición.

Sintaxis

```
CAC(<n>)
CIC(<n>)
CACP(<n>)
CACN(<n>)
```

Descripción

`CAC(<n>)`:	Desplazar a la posición codificada desde el número de posición n
`CIC(<n>)`:	Desplazar a la posición codificada n posiciones hacia delante (+) o hacia atrás (–n), partiendo del número de posición actual
`CDC(<n>)`:	Desplazar a la posición codificada desde el número de posición n por el trayecto más corto (solo para ejes giratorios)
`CACP(<n>)`:	Desplazar a la posición codificada desde el número de posición n en la dirección positiva (solo para ejes giratorios)
`CACN(<n>)`:	Desplazar a la posición codificada desde el número de posición n en la dirección negativa (solo para ejes giratorios)
`<n>`:	Número de posición en la tabla de datos de máquina Rango de valores: 0, 1, … (número máx. de puestos en la tabla - 1)

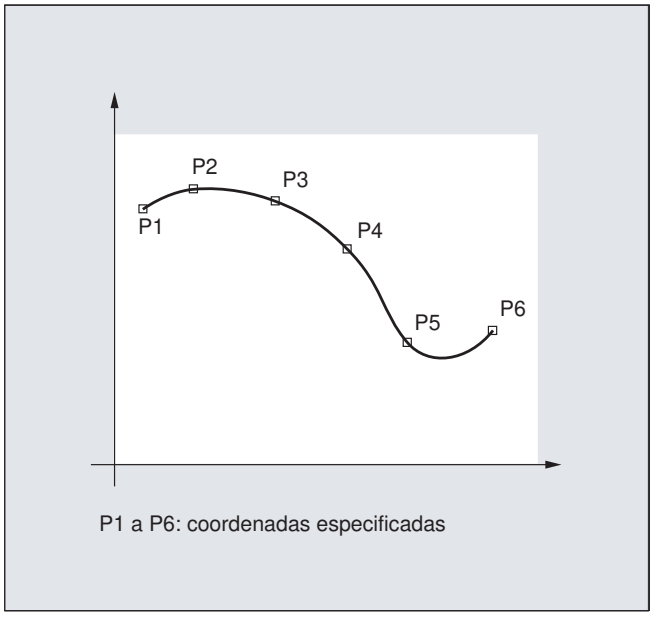
Ejemplo: Desplazamiento a posiciones codificadas de un eje de posicionado

Código de programación	**Comentarios**
`N10 FA[B]=300`	`; Avance para eje de posicionado B`
`N20 POS[B]=CAC(10)`	`; Desplazar a la posición codificada desde el número de posición 10`
`N30 POS[B]=CIC(-4)`	`; Desplazar a la posición codificada desde el "número de posición actual" - 4`

3.7.2 Interpolación spline (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Los contornos de curvatura irregular de las piezas no se pueden describir analíticamente con exactitud. Por eso ese tipo de contornos se describen de modo aproximado por medio de un número limitado de puntos de interpolación, p. ej. para la digitalización de superficies. A fin de generar la superficie digitalizada en una pieza, deben asociarse los puntos de interpolación para formar una descripción de contorno. Para lograrlo se usa la interpolación spline.

Un spline define una curva compuesta por polinomios de 2.º o 3.er grado. Las propiedades de los puntos de interpolación de un spline pueden definirse **dependiendo del tipo de spline utilizado**.



En SINUMERIK solution line están disponibles los siguientes tipos de splines:

- Spline A
- Spline B
- Spline C

Sintaxis

General:

```
ASPLINE X... Y... Z... A... B... C...
BSPLINE X... Y... Z... A... B... C...
CSPLINE X... Y... Z... A... B... C...
```

Programable adicionalmente para splines B:

```
PW=<n>
SD=2
PL=<Valor>
```

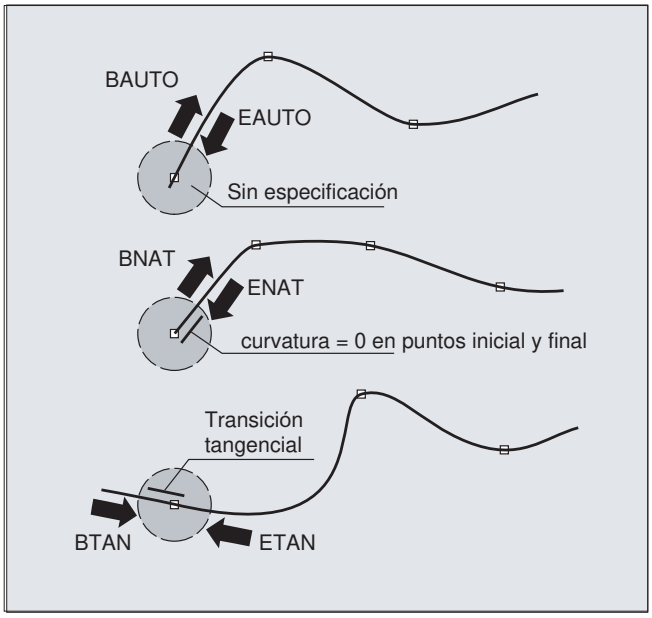
Programable adicionalmente para splines A y C:

BAUTO/BNAT/BTAN

EAUTO/ENAT/ETAN

Descripción

Tipo de interpolación spline:			
ASPLINE:	Comando para activar la interpolación de splines A		
BSPLINE:	Comando para activar la interpolación de splines B		
CSPLINE:	Comando para activar la interpolación de splines C		
	Los comandos ASPLINE, BSPLINE y CSPLINE son de tipo modal y pertenecen al grupo de los comandos de desplazamiento.		
Puntos de interpolación o puntos de control:			
X... Y... Z... A... B... C...	Posiciones en coordenadas cartesianas		
Peso del punto (sólo para splines B):			
PW:	El comando PW permite programar un "peso del punto" para cada punto de interpolación.		
<n>:	"Peso del punto"		
	Rango de valores:	0 ≤ n ≤ 3	
	Paso:	0.0001	
	Efecto:	n > 1	El punto de control atrae más a la curva.
		n < 1	El punto de control atrae menos a la curva.
Grado del spline (sólo para splines B):			
SD:	De forma estándar se utilizan polinomios de tercer grado. Sin embargo, programando SD=2 puede utilizarse también un polinomio de 2.º grado.		
Distancia entre nodos (sólo para splines B):			
PL:	Las distancias a los nodos se calculan adecuadamente a nivel interno. Pero el control también puede procesar distancias entre nodos especificadas, que se indican como "longitud de intervalo de parámetros" con el comando PL.		
<Valor>:	Longitud de intervalo de parámetros		
	Rango de valores:	como la cota	
Comportamiento de transición al inicio de la curva spline (sólo para splines A o C):			
BAUTO:	No se especifica ningún valor para el comportamiento de transición. El inicio se obtiene de la posición del primer punto.		
BNAT:	Curvatura cero		
BTAN:	Transición tangencial a la secuencia anterior (posición de borrado)		
Comportamiento de transición al final de la curva spline (sólo para splines A o C):			
EAUTO:	No se especifica ningún valor para el comportamiento de transición. El final se obtiene de la posición del último punto.		

ENAT:	Curvatura cero
ETAN:	Transición tangencial a la secuencia anterior (posición de borrado)
	

Nota

El comportamiento de transición programable no afecta al spline B. En los puntos inicial y final la interpolación B-spline es siempre tangencial al polígono de referencia.

Condiciones

- Puede utilizarse la corrección de radio de herramienta.
- La vigilancia de colisión se efectúa en la proyección sobre el plano.

Ejemplos

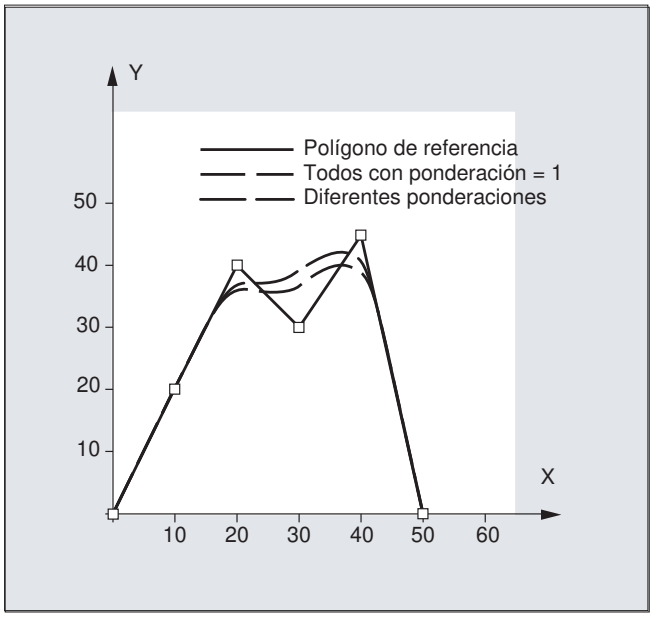
Ejemplo 1: Spline B

```
Código de programa 1 (todos los pesos 1)
N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20
N40 X20 Y40
N50 X30 Y30
N60 X40 Y45
N70 X50 Y0
```

Código de programa 2 (diversos pesos)

```
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
```

Código de programa 3 (polígono de referencia)	**Comentarios**
N10 G1 X0 Y0 F300 G64	
N20	; Se suprime
N30 X10 Y20	
N40 X20 Y40	
N50 X30 Y30	
N60 X40 Y45	
N70 X50 Y0	

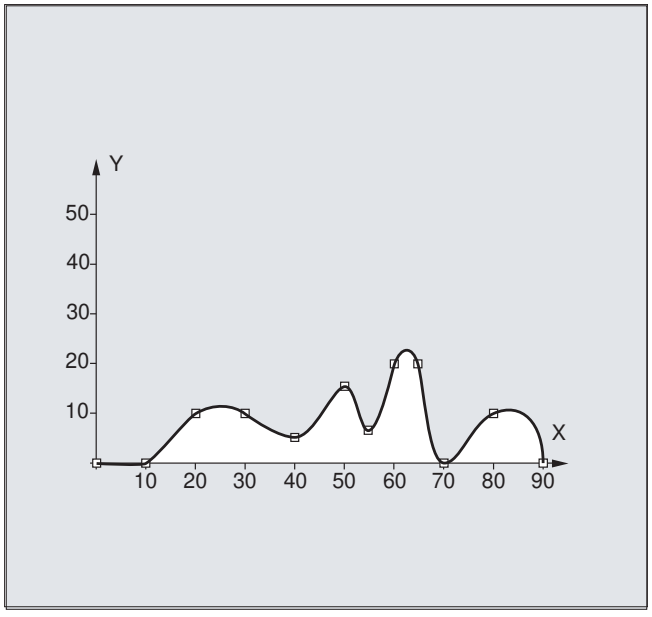


Ejemplo 2: spline C con radio de curvatura cero en puntos inicial y final

Código de programa

```
N10 G1 X0 Y0 F300
N15 X10
N20 BNAT ENAT
N30 CSPLINE X20 Y10
```

```
Código de programa
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
```



Ejemplo 3: interpolación spline (spline A) y transformación de coordenadas (ROT)

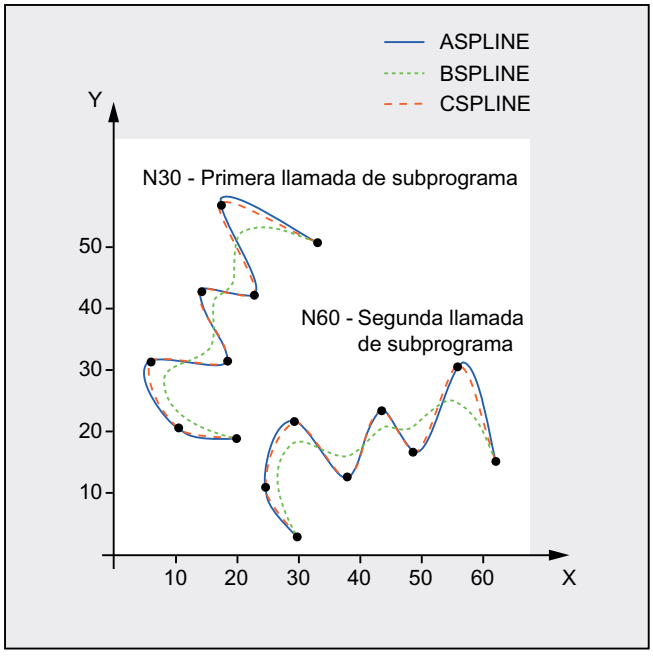
Programa principal:

Código de programa	Comentarios
N10 G00 X20 Y18 F300 G64	; Desplazamiento al punto inicial.
N20 ASPLINE	; Activar el tipo de interpolación de spline A.
N30 CONTORNO	; Primera llamada al subprograma.
N40 ROT Z-45	; Transformación de coordenadas: Giro de -45° de WKS alrededor del eje Z.
N50 G00 X20 Y18	; Desplazamiento al punto inicial del contorno.
N60 CONTORNO	; Segunda llamada al subprograma.
N70 M30	; Fin del programa

Subprograma "Contorno" (contiene las coordenadas de los puntos de interpolación):

Código de programa
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

En la figura siguiente se muestran, además de la curva spline resultante del ejemplo de programación (ASPLINE), las curvas spline que se habrían obtenido en caso de activar una interpolación de spline B o C (BSPLINE, CSPLINE):



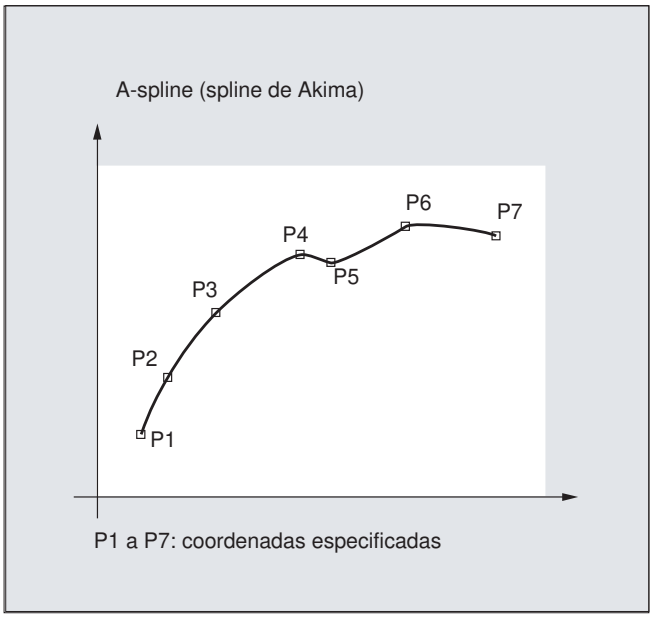
Información adicional

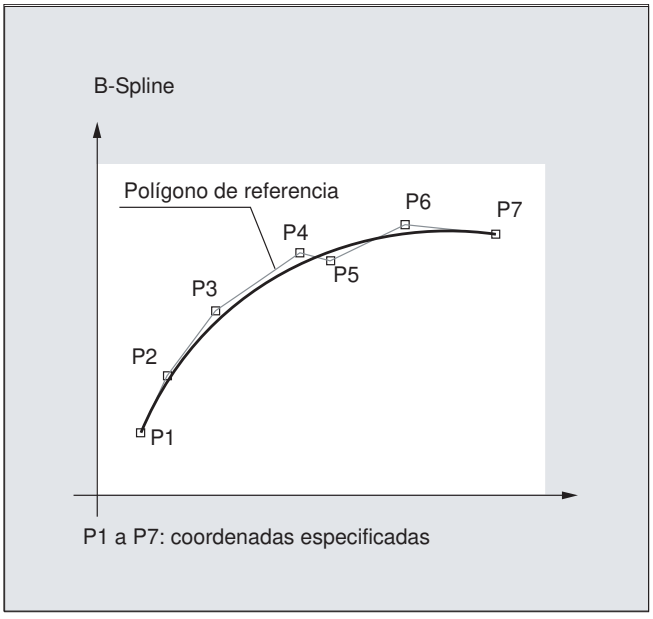
Ventajas de la interpolación spline

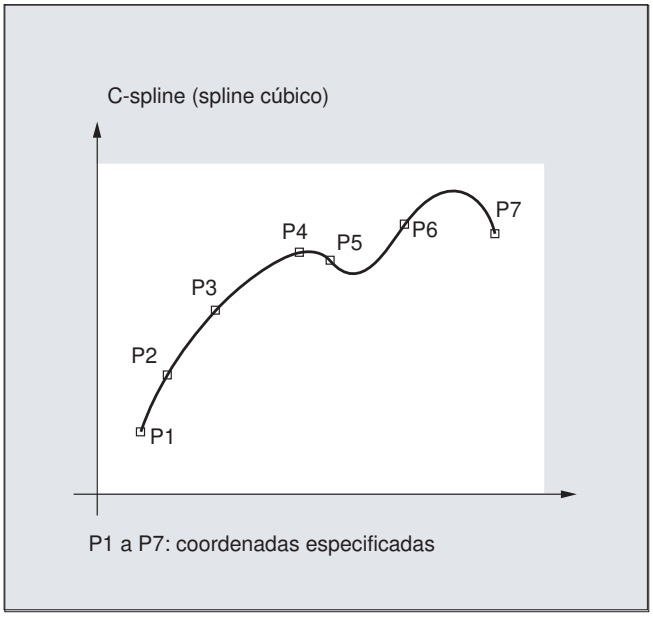
Utilizando la interpolación spline se obtienen las siguientes ventajas respecto a la utilización de secuencias de rectas G01:

- Menor número de secuencias de programa de pieza necesarias para la descripción del contorno
- Trazado de las curvas en la transición entre secuencias de programa de pieza suave y sin sobrecargas mecánicas

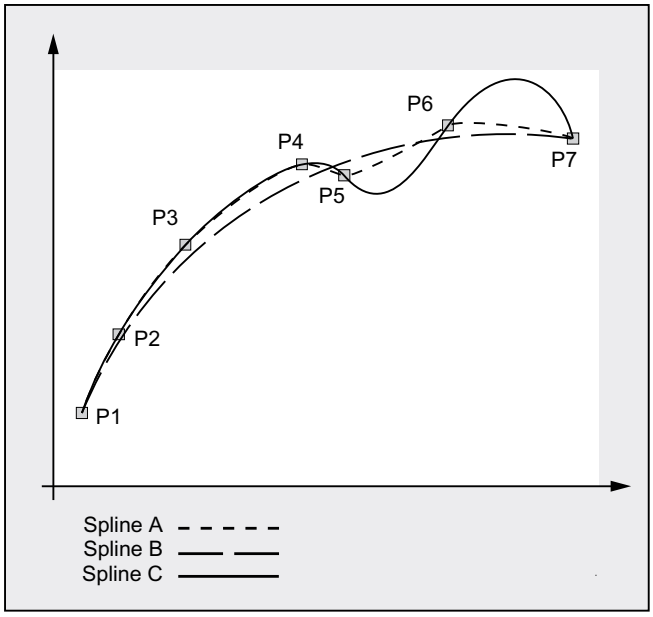
Características y aplicación de los diferentes tipos de splines

Tipo de spline	Características y aplicación
Spline A	 **Características:** • Pasa exactamente por entre los puntos de interpolación especificados. • El trazado de la curva es continuo en la tangente pero no en la curvatura. • Apenas genera oscilaciones no deseadas. • El área de influencia de las modificaciones de los puntos de interpolación es local, es decir: la modificación de uno de los puntos afecta como máximo a 6 puntos adyacentes. **Aplicación:** El spline A es idóneo para la interpolación de trazados de curva con grandes cambios de paso (p. ej. trazados curvos en forma de escalera).

Tipo de spline	Características y aplicación
Spline B	 **Características:** • No pasa por entre los puntos de interpolación especificados, sino sólo cerca de ellos. La curva es atraída por los puntos de interpolación. Otra manera de modificar el trazado de la curva es ponderando los puntos de interpolación. • El trazado de la curva es continuo en la tangente y en la curvatura. • No genera oscilaciones no deseadas. • El área de influencia de las modificaciones de los puntos de interpolación es local, es decir: la modificación de uno de los puntos afecta como máximo a 6 puntos adyacentes. **Aplicación:** La función principal del spline B es servir de interfaz con los sistemas CAD.

Tipo de spline	Características y aplicación
Spline C	 **Características:** • Pasa exactamente por entre los puntos de interpolación especificados. • El trazado de la curva es continuo en la tangente y en la curvatura. • Suele generar oscilaciones no deseadas, sobre todo en los puntos con grandes cambios de paso. • El área de influencia de las modificaciones de los puntos de interpolación es global; es decir, la modificación de un punto de interpolación afecta a todo el trazado de la curva. **Aplicación:** El spline C resulta de utilidad cuando los puntos de interpolación se encuentran en una curva previamente analizada (círculo, parábola, hipérbola)

Comparación de los tres tipos de interpolación spline para los mismos puntos predefinidos



Cantidad mínima de secuencias spline

Los comandos G ASPLINE, BSPLINE B y CSPLINE unen los puntos finales de la secuencia con splines. Para ello se deben de preprocesar una serie de secuencias (puntos finales) y calcularlos simultáneamente. El tamaño por defecto del búfer para el cálculo es de 10 secuencias. No toda información de secuencia es un punto final de spline. Sin embargo, el control precisa para 10 secuencias una cantidad determinada de secuencias de punto final de spline:

Tipo de spline	Cantidad mínima de secuencias spline
A-spline:	Por lo menos **4** secuencias de cada 10 deben ser secuencias spline. No cuentan secuencias con comentarios y cálculo de parámetros.
B-spline:	Por lo menos **6** secuencias de cada 10 deben ser secuencias spline. No cuentan secuencias con comentarios y cálculo de parámetros.
C-spline:	La cantidad mínima necesaria de secuencias spline resulta de la siguiente suma: valor de MD20160 $MC_CUBIC_SPLINE_BLOCKS + 1 En el MD20160 se indica la cantidad de puntos mediante la que se calcula la sección spline. El ajuste estándar es 8. Por tanto, de cada 10 secuencias, en el caso estándar, deben ser secuencias spline como mínimo **9**.

Nota

Si se rebasa por efecto el valor de tolerancia, se genera una alarma. También se emite una alarma si un eje que forma parte de la interpolación spline se programa simultáneamente como un eje de posicionado.

Resumen de secuencias spline breves

En la interpolación spline pueden aparecer secuencias spline breves que llevan a una reducción innecesaria de la velocidad de contorneado. Con la función "Resumen de secuencias spline breves" se pueden resumir estas secuencias de modo que la longitud de secuencia resultante sea suficientemente grande y no conduzca a una disminución de la velocidad de contorneado.

La función se activa mediante el dato de máquina específico del canal:

MD20488 $MC_SPLINE_MODE (ajuste de interpolación spline)

Información adicional: Manual de funciones Funciones básicas

3.7.3 Conjunto spline (SPLINEPATH)

Para seleccionar los ejes que se interpolarán en el conjunto spline se usa el comando `SPLINEPATH`. En una interpolación spline pueden intervenir hasta 8 ejes de contorneado.

Nota

Si no se programa explícitamente el comando `SPLINEPATH`, se desplazarán en forma de conjunto spline los tres primeros ejes del canal.

Sintaxis

El conjunto spline se define mediante una secuencia independiente:

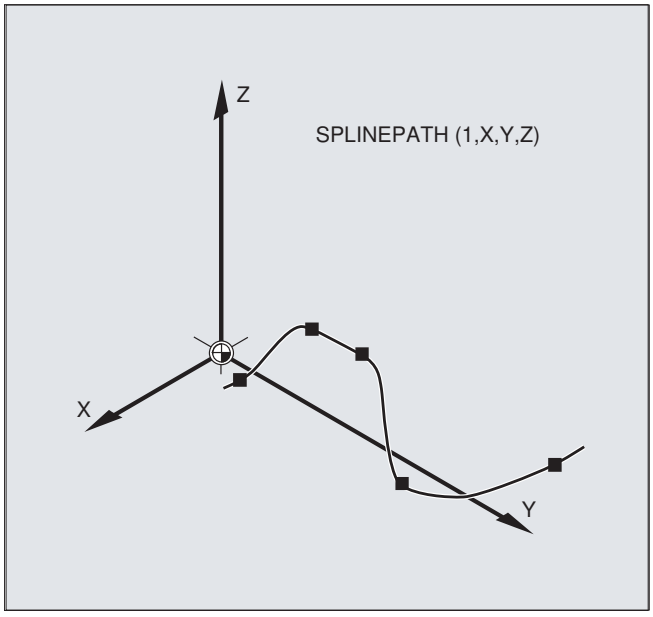
```
SPLINEPATH(n,X,Y,Z,…)
```

Descripción

`SPLINEPATH`:	Comando para definir un conjunto spline
`n`:	=1 (valor fijo)
`X,Y,Z,…` :	Identificadores de los ejes de contorneado que se interpolarán en el conjunto spline

Ejemplo: Conjunto spline con tres ejes de contorneado

Código de programa	**Comentarios**
`N10 G1 X10 Y20 Z30 A40 B50 F350`	
`N11 SPLINEPATH(1,X,Y,Z)`	`; Conjunto spline`
`N13 CSPLINE BAUTO EAUTO X20 Y30 Z40 A50 B60`	`; Spline C`
`N14 X30 Y40 Z50 A60 B70`	`; Puntos de interpolación`
`…`	
`N100 G1 X… Y…`	`; Cancelar interpolación spline`



3.7.4 Activar/desactivar compresión de secuencias CN (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF)

Las funciones de compresión de secuencias lineales (y también secuencias circulares y en rápido, según la parametrización) se activan/desactivan con los comandos G del grupo G 30. Los comandos son de tipo modal.

Sintaxis

```
COMPON/COMPCURV/COMPCAD/COMPSURF
...
COMPOF
```

Descripción

COMPON:	Activación de la función de compresor COMPON
COMPCURV:	Activación de la función de compresor COMPCURV
COMPCAD:	Activación de la función de compresor COMPCAD
COMPSURF:	Activación de la función de compresor COMPSURF
COMPOF:	Desconexión de la función del compresor actualmente activa

Nota

Para una mejora aún mayor de la calidad de superficie, pueden utilizarse las funciones de matado de esquinas G642 y limitación de tirones SOFT. Estos comandos deben escribirse al inicio del programa.

Ejemplo: COMPCAD

Código de programa	Comentarios
N10 G00 X30 Y6 Z40	
N20 G1 F10000 G642	; Activar: función de matado de esquinas G642
N30 SOFT	; Activar: limitación de tirones SOFT
N40 **COMPCAD**	; Activar: función de compresor COMPCAD
N50 FIFOCTRL	
N24050 Z32.499	; 1.ª secuencia de desplazamiento
N24051 X41.365 Z32.500	; 2.ª secuencia de desplazamiento
...	
N99999 X... Z...	; Última secuencia de desplazamiento
COMPOF	; Desactiva la función de compresor.
...	

3.7.5 Interpolación polinómica (POLY, POLYPATH, PO, PL)

La interpolación polinómica (POLY) no es un tipo de interpolación spline propiamente dicha. El primer lugar, está prevista como interfaz para la programación de curvas spline generadas a nivel externo. Las secciones spline se pueden programar directamente.

Este tipo de interpolación descarga al CN del cálculo de los coeficientes polinómicos. Esta función es especialmente apropiada cuando los coeficientes polinómicos se introducen directamente desde un sistema CAD o bien desde un postprocesador.

Sintaxis

Polinomio de 3.er grado:
```
POLY PO[X]=(xe,a2,a3) PO[Y]=(ye,b2,b3) PO[Z]=(ze,c2,c3) PL=n
```

Polinomio de 5.º grado y nueva sintaxis polinómica:
```
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")
```

Nota

La suma de los coeficientes polinómicos y ejes programados en una secuencia CN no debe superar el número máximo de ejes permitido por secuencia.

Descripción

POLY:	Activar la interpolación polinómica con una secuencia con POLY.
POLYPATH:	Interpolación polinómica seleccionable para los dos grupos de ejes AXIS o VECT
PO[Identificador de eje/variable]:	Puntos finales y coeficientes de polinomio

`X, Y, Z`:	Identificador de eje
`xe, ye, ze` :	Indicar la posición final para el eje correspondiente; margen de valores como en cota
`a2, a3, a4, a5`:	Los coeficientes a_2, a_3, a_4 y a_5 se escriben con su valor; margen de valores como en la cota. Se puede prescindir en cada caso del último coeficiente si tiene el valor cero.
`PL`:	Longitud del intervalo de parámetros en el que se definen los polinomios (rango de definición de la función f(p)). El intervalo empieza siempre con 0, p puede tomar valores entre 0 y PL. Margen de valores teórico para PL: 0,0001 … 99 999,9999 **Nota:** El valor de PL es válido para la secuencia en la que se encuentra. Si no se ha programado ningún PL, entonces PL=1.

Activar/desactivar la interpolación polinómica

La interpolación polinómica se activa en el programa de pieza con el comando G `POLY`.

El comando G `POLY`, junto con `G0`, `G1`, `G2`, `G3`, `ASPLINE`, `BSPLINE` y `CSPLINE`, pertenece al 1.er grupo G.

Los ejes que se programan solo con su nombre y punto final (p. ej. `X10`), se desplazan de modo lineal. Si todos los ejes de una secuencia CN están programados de este modo, el control se comporta como con G1.

La interpolación polinómica se vuelve a desactivar de forma implícita programando otro comando del 1.er grupo G (p. ej. `G0`, `G1`).

Coeficiente de polinomio

El valor de `PO (PO[])` o `...=PO(...)` indica todos los coeficientes de polinomio correspondientes a un eje. De acuerdo con el grado del polinomio, se especifican varios valores separados por comas. Dentro de una secuencia puede haber diferentes grados de polinomio, correspondientes a diferentes ejes.

Subprograma POLYPATH

Con `POLYPATH(...)` se puede habilitar la interpolación polinómica de forma selectiva para determinados grupos de ejes:

Solo ejes de contorneado y ejes adicionales:	`POLYPATH("AXES")`
Solo ejes de orientación: (en el desplazamiento con transformada de orientación)	`POLYPATH("VECT")`

Los ejes que no están habilitados en cada caso se desplazan de modo lineal.

Por defecto está habilitada la interpolación polinómica para ambos grupos de ejes.

Programando sin parámetros `POLYPATH( )` se desactiva la interpolación polinómica para todos los ejes.

Ejemplo

Código de programa	**Comentarios**
`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`	`; Activa la interpolación polinómica`
`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`	`; Datos mixtos para los ejes`
`N27 PO[X]=(10,2.5) PO[Y]=(2.3)`	`; No hay PL programado; PL = 1`
`N30 G1 X… Y… Z.`	`; Desactiva la interpolación polinómica`
…	

Ejemplo: Nueva sintaxis polinómica

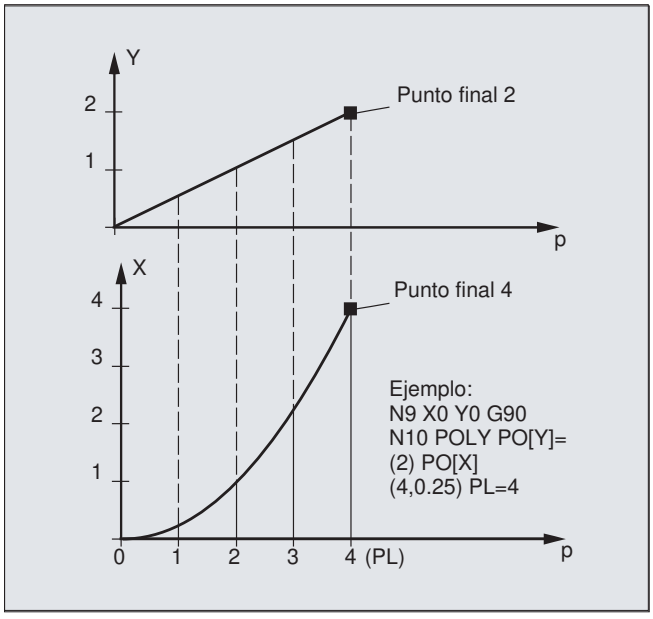
Sintaxis polinómica todavía válida	Nueva sintaxis polinómica
`PO[Identificador de eje]=(.. , ..)`	`Identificador de eje=PO(.. , ..)`
`PO[PHI]=(.. , ..)`	`PHI=PO(.. , ..)`
`PO[PSI]=(.. , ..)`	`PSI=PO(.. , ..)`
`PO[THT]=(.. , ..)`	`THT=PO(.. , ..)`
`PO[]=(.. , ..)`	`PO(.. , ..)`
`PO[variable]=IC(.. , ..)`	`variable=PO IC(.. , ..)`

Ejemplo: Curva en el plano X/Y

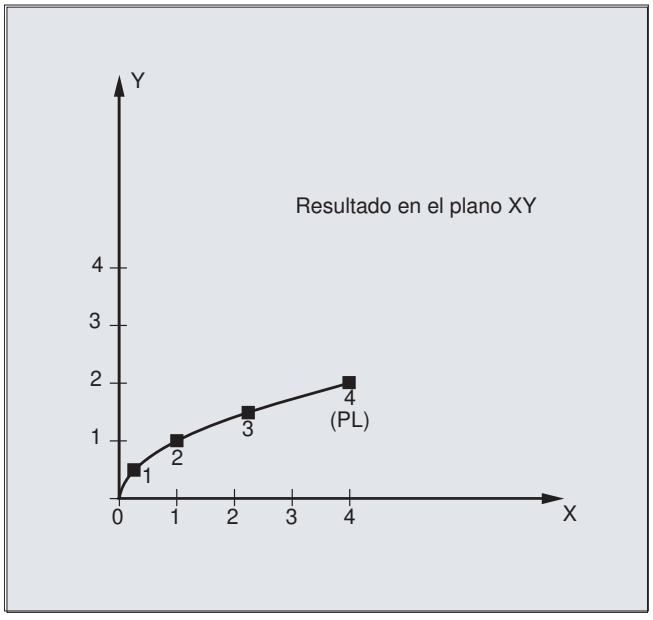
Programación

Código de programa
`N9 X0 Y0 G90 F100`
`N10 POLY PO[Y]=(2) PO[X]=(4,0.25) PL=4`

Forma de las curvas X(p) e Y(p)



Forma de la curva en el plano XY



Descripción

La expresión general para un polinomio de tercer grado es:

$f(p)= a_0 + a_1p + a_2p^2 + \ldots + a_np^n$

don- a_i: coeficientes constantes (i = 0, 1, ..., n)
de: p: Parámetros

En el control pueden programarse como máximo polinomios de 5.º grado:

$$f(p)= a_0 + a_1p + a_2p^2 + a_3p^3 + a_4p^4 + a_5p^5$$

Se pueden generar diferentes formas de curva tales como rectas, parábolas y funciones exponenciales mediante la asignación de valores determinados a los coeficientes.

Una recta se genera con $a_2 = a_3 = a_4 = a_5 = 0$:

$$f(p) = a_0 + a_1p$$

Donde:

$a_{0:}$ posición de eje al final de la secuencia precedente
p = PL
$a_1 = (x_E - a_0 - a_2*p^2 - a_3*p^3) / p$

Es posible programar polinomios **sin** que se haya activado la interpolación polinómica con el comando G `POLY`. En tal caso no se interpolan los polinomios programados, sino que se produce un desplazamiento lineal a los puntos finales programados de los ejes (`G1`). Solo tras activar explícitamente la interpolación polinómica en el programa de pieza (`POLY`) se desplazan también los polinomios programados como tales.

Particularidad: Denominador polinómico

`PO[]=(…)` permite programar un denominador polinómico común para ejes geométricos sin especificar el nombre del eje; esto quiere decir que el desplazamiento de ejes geométricos se interpola como el cociente entre dos polinomios.

De este modo se pueden mecanizar de forma exacta, p. ej., secciones cónicas (circunferencias, elipses, parábolas, hipérbolas).

Ejemplo:

Código de programa	**Comentarios**
`POLY G90 X10 Y0 F100`	`; Los ejes geométricos se desplazan de forma lineal a la posición X10 Y0.`
`PO[X]=(0,-10) PO[Y]=(10) PO[]=(2,1)`	`; Los ejes geométricos se desplazan a lo largo del cuadrante de circunferencia a X0 Y10.`

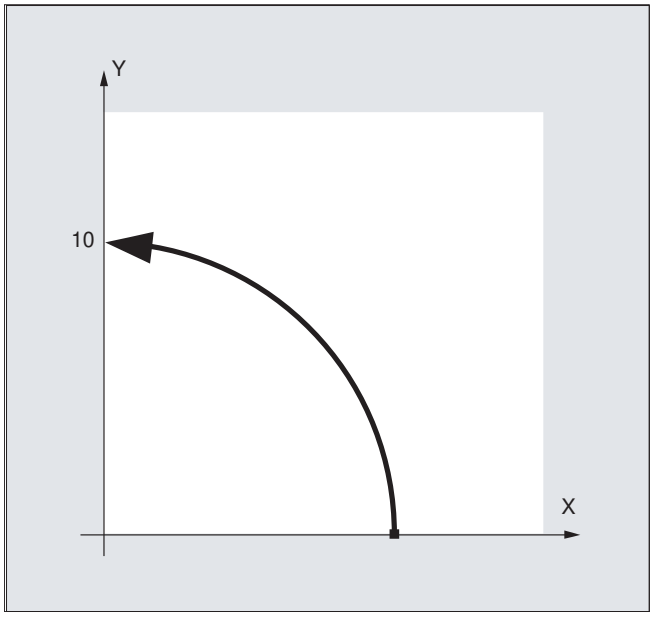
El coeficiente constante (a_0) del denominador polinómico siempre se toma como 1. El punto final programado es independiente de `G90`/`G91`.

A partir de los valores programados se obtienen X(p) e Y(p):

$$X(p) = (10 - 10 * p^2) / (1 + p^2)$$
$$Y(p) = 20 * p / (1 + p^2)$$
con $0 \le p \le 1$

Los siguientes resultados parciales se obtienen a partir de los puntos iniciales y finales programados, del coeficiente a_2 y PL=1:

Numerador (X) =	$10 + 0 * p - 10 * p^2$
Numerador (Y) =	$0 + 20 * p + 0 * p^2$
Denominador =	$1 + p^2$



Cuando se tiene una interpolación polinómica activa, la programación de un denominador polinómico con ceros en el intervalo `[0,PL]` se rechaza, y se presenta una alarma. El denominador polinómico carece de efecto en desplazamientos de ejes adicionales.

Nota

La corrección del radio de herramienta también se puede activar para interpolaciones polinómicas mediante `G41` y `G42`, y se utiliza de la misma manera que para interpolaciones lineales o circulares.

3.7.6 Referencia de trayectoria ajustable (SPATH, UPATH)

En la interpolación polinómica (`POLY`, `ASPLINE`, `BSPLINE`, `CSPLINE`, `COMP...`), las posiciones de los ejes de contorneado i se especifican mediante polinomios $p_i(U)$. El parámetro de curva U se desplaza dentro de una secuencia CN desde 0 hasta 1.

Mediante `FGROUP` se eligen los ejes (ejes FGROUP) a los que hace referencia el avance de contorneado `F`. Sin embargo, una interpolación con velocidad constante en la trayectoria S de los ejes FGROUP normalmente significa durante la interpolación polinómica una modificación

no constante del parámetro de curva U. Por ello, para los ejes no contenidos en FGROUP puede escogerse entre dos modos de seguimiento de los ejes FGROUP:

- De modo síncrono a la trayectoria S (SPATH)
- De modo síncrono al parámetro de curva U (UPATH)

Sintaxis

```
SPATH
UPATH
```

Descripción

SPATH:	Los ejes no contenidos en FGROUP se desplazan en referencia a la trayectoria S
UPATH:	Los ejes no contenidos en FGROUP se desplazan en referencia al parámetro de curva U

Nota

UPATH y SPATH determinan también la relación del polinomio de palabra F (FPOLY, FCUB, FLIN) con el desplazamiento de trayectoria.

Condiciones

SPATH o UPATH no tienen ningún significado en:

- Interpolación lineal (G1)
- Interpolación circular (G2, G3)
- Secuencias de roscado (G33, G34, G35, G33x, G63)
- Todos los ejes de contorneado están contenidos en FGROUP

Ejemplo

El siguiente ejemplo muestra la diferencia entre las dos clases de guiado de movimiento.

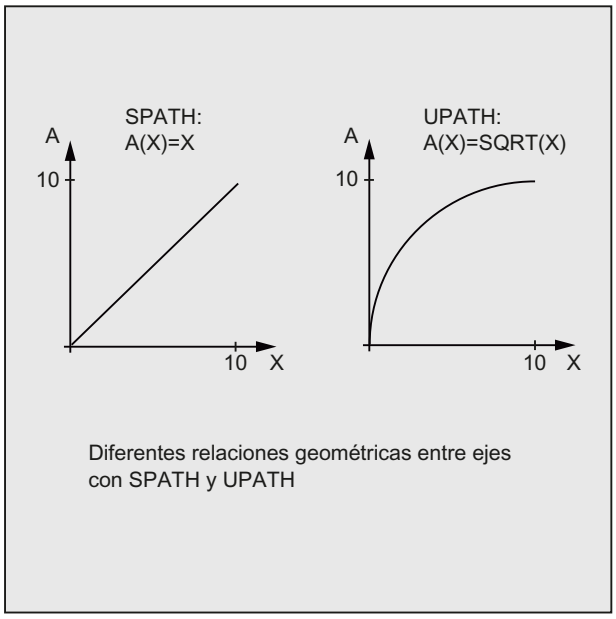
Código de programa

```
N10 FGROUP(X,Y,Z)
N15 G1 X0 A0 F1000 SPATH                ; SPATH
N20 POLY PO[X]=(10,10) A10
```

Código de programa

```
N10 FGROUP(X,Y,Z)
N15 G1 X0 A0 F1000 UPATH                ; UPATH
N20 POLY PO[X]=(10,10) A10
```

En los dos segmentos de programa, en `N20`, el recorrido S de los ejes FGROUP depende del cuadrado del parámetro de curva U. Por ello, a lo largo del recorrido de X se obtienen diferentes posiciones del eje síncrono A, según si está activo `SPATH` o `UPATH`.



Diferentes relaciones geométricas entre ejes con SPATH y UPATH

Información adicional

Comportamiento del control con Reset y datos de máquina/opciones

Después del reset está activo el comando G definido por MD20150 $MC_GCODE_RESET_VALUES[44] (grupo G 45).

El ajuste básico para el tipo de matado de esquinas se define con MD20150 $MC_GCODE_RESET_VALUES[9] (grupo G 10).

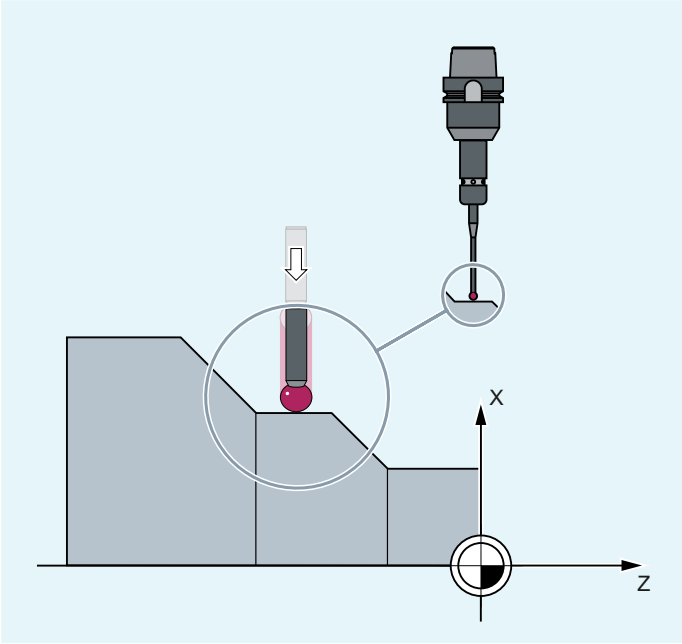
El dato de máquina específico de eje MD33100 $MA_COMPRESS_POS_TOL[<n>] tiene otro significado: contiene las tolerancias para la función de compresor y el matado de esquinas con G642.

3.7.7 Medición específica de canal (MEAS, MEAW)

En la medición específica de canal, la activación del proceso de medición de un canal CN se realiza siempre desde el programa de pieza que esté funcionando en el canal en cuestión. En una secuencia de medición se programa **un** evento de disparo (flanco ascendente o descendente del palpador) y **un** modo de medición con borrado de trayecto residual (MEAS) o sin borrado de trayecto residual (MEAW). En el proceso de medición participan **todos** los ejes programados en la secuencia de medición.

En cuanto se activa una secuencia de medición, se aproxima el palpador a la pieza. Con el flanco de conmutación del palpador se miden las posiciones para todos los ejes programados en la secuencia de medición y se escriben en la correspondiente celda de memoria para cada eje.

Los resultados de medición se pueden leer en el programa de pieza o, por medio de acciones síncronas, en el sistema de coordenadas de máquina o en el sistema de coordenadas de pieza.



Sintaxis

```
MEAS=<TE> G... X... Y... Z...
MEAW=<TE> G... X... Y... Z...
```

Descripción

MEAS:	Medición **con** borrado de trayecto residual		
	Efecto:		Secuencia a secuencia
MEAW:	Medición **sin** borrado de trayecto residual		
	Efecto:		Secuencia a secuencia
	Aplicación:		Para procesos de medición en los que se debe alcanzar la posición programada en cada caso.
<TE>:	Evento de disparo para activar la medición		
	Tipo:	INT	
	Rango de valores:		-2, -1, 1, 2
	Valor:	(+)1	Flanco ascendente del palpador 1 (en entrada de medición 1)
		-1	Flanco descendente del palpador 1 (en entrada de medición 1)
		(+)2	Flanco ascendente del palpador 2 (en entrada de medición 2)
		-2	Flanco descendente del palpador 2 (en entrada de medición 2)
	Nota: Existen 2 palpadores como máximo (según configuración).		

`G...:`	Tipo de interpolación, p. ej., `G0`, `G1`, `G2` o `G3`
`X... Y... Z...:`	Puntos finales en coordenadas cartesianas

Nota

MEAS y MEAW se aplican por secuencias y se programan junto con comandos de desplazamiento. Tanto el avance y el tipo de interpolación (G0, G1...) como el número de ejes deben adaptarse en cada caso al problema de medición en cuestión.

Ejemplo

Código de programa	**Comentarios**
`...`	
`N10 MEAS=1 G1 F1000 X100 Y730 Z40`	`; Secuencia de medición con palpador por la primera entrada de medición e interpolación lineal. Se realiza una parada de decodificación previa de forma automática.`
`...`	

Información adicional

Consulta del estado

Si el programa exige evaluar si un palpador se ha deflectado o ha conmutado, puede consultarse el estado mediante las siguientes variables de sistema:

Variable de sistema	Descripción	Tipo de datos	Valor	
$A_PROBE[<n>]	Estado del palpador	INT	0	Palpador no deflectado.
			1	Palpador deflectado.
$AC_MEA[<n>]	Estado de conmutación del palpador $AC_MEA[<n>] se resetea automáticamente al comienzo de la medición.	INT	0	El palpador no ha conmutado.
			1	El palpador ha conmutado.

<n> = Número de palpador

Registro de medidas

En la medición específica de canal, se capturan las posiciones de todos los ejes de contorneado y de posicionado desplazados de la secuencia (número máximo de ejes según

configuración del control). Con MEAS se frena el desplazamiento de modo definido cuando el detector conmuta.

Nota

Si en una secuencia de medición está programado un eje geométrico, se guardan las medidas de todos los ejes geométricos actuales.

Si en una secuencia de medición está programado un eje que participa en una transformada, se guardan las medidas de todos los ejes que participan en esa transformada.

Lectura de resultados de medición

Los valores medidos del palpador para los ejes capturados se pueden leer en el programa de pieza y en las acciones síncronas mediante las siguientes variables de sistema:

Variable de sistema	Descripción
$AA_MM[<Axis>]	Valor medido del palpador en el sistema de coordenadas de máquina.
$AA_MW[<Axis>]	Valor medido del palpador en el sistema de coordenadas de pieza.

3.7.8 Medición específica de eje (MEASA, MEAWA, MEAC) (opcional)

En la medición específica de eje, la activación puede tener lugar en el programa de pieza **o** en las acciones síncronas. Si hay dos sistemas de medida disponibles para el eje, pueden emplearse ambos para la medición.

Se dispone de los siguientes métodos de medición:

- Medición con borrado de trayecto residual (MEASA)
- Medición sin borrado de trayecto residual (MEAWA)
- Medición continua sin borrado de trayecto residual (MEAC)

Con MEASA o MEAWA se capturan, para cada eje programado, hasta cuatro valores medidos por medición, que se guardan en variables de sistema en función del evento de disparo.

En la medición continua con MEAC, los resultados de medición se almacenan en variables FIFO.

Sintaxis

```
MEASA[<Axis>]=(<Mode>,<TE1>,...,<TE4>)
MEAWA[<Axis>]=(<Mode>,<TE1>,...,<TE4>)
MEAC[<Axis>]=(<Mode>,<MeasMem>,<TE1>,...,<TE4>)
```

Nota

MEASA y MEAWA se aplican por secuencias y se pueden programar juntos en una secuencia. Por el contrario, si MEASA/MEAWA se programa en una misma secuencia con MEAS/MEAW, se emite un aviso de error.

Descripción

MEASA:	Medición específica de eje con borrado de trayecto residual	
	Efecto:	Secuencia a secuencia
MEAWA:	Medición específica de eje sin borrado de trayecto residual	
	Efecto:	Secuencia a secuencia
MEAC:	Medición continua específica de eje sin borrado de trayecto residual	
	Efecto:	Secuencia a secuencia
<Axis>:	Nombre del eje de canal utilizado para la medición	
<Mode>:	Número de dos dígitos (xx) para indicar el modo de operación (modo de medición y sistema de medida)	
	Década de unidades: modo de medición Define si los eventos de disparo deben activarse en orden cronológico o programado.	
	x**0**	Interrumpir la orden de medición.
	x**1**	Hasta 4 eventos de disparo **diferentes** activables simultáneamente.
	x**2**	Hasta 4 eventos de disparo activables sucesivamente.
	x**3**	Hasta 4 eventos de disparo activables sucesivamente, pero sin vigilancia del evento de disparo 1 en el arranque (se suprimen las alarmas 21700/21703). **Nota:** Este modo no es posible con MEAC.
	Década de decenas: Sistema de medida Define con qué sistema de medida debe realizarse la medición.	
	0x (o sin datos)	Sistema de medida activo
	1x	Sistema de medida 1
	2x	Sistema de medida 2
	3x	Ambos sistemas de medida
<TE>:	Evento de disparo para activar la medición	
	Tipo:	INT
	Rango de valores:	-2, -1, 1, 2
	(+)1	Flanco ascendente del palpador 1
	-1	Flanco descendente del palpador 1
	(+)2	Flanco ascendente del palpador 2
	-2	Flanco descendente del palpador 2
	Nota: Si la medición se efectúa con dos sistemas de medida, pueden programarse como máximo dos eventos de disparo (flanco ascendente o descendente). En cada uno de los dos eventos de disparo se capturan los valores medidos de ambos sistemas de medida.	
<MeasMem>:	Número del FIFO (memoria cíclica)	

Ejemplos

Ejemplo 1: Medición específica de eje con borrado de trayecto residual en el modo 1 (evaluación en orden cronológico)

a) Medición con un sistema de medida

Código de programa	Comentarios
...	
N100 **MEASA[X]=(1,1,-1)** G01 X100 F100	; Medición en modo 1 con el sistema de medida activo. Esperar la señal de medición con flanco ascendente/descendente del palpador 1 en el recorrido hacia X=100.
N110 IF $AC_MEA[1]==FALSE GOTOF ENDE	; Comprobar si la medición se ha realizado correctamente.
N120 R10=$AA_MM1[X]	; Guardar la medida correspondiente al primer evento de disparo programado (flanco ascendente).
N130 R11=$AA_MM2[X]	; Guardar la medida correspondiente al segundo evento de disparo programado (flanco descendente).
N140 ENDE:	

b) Medición con dos sistemas de medida

Código de programa	Comentarios
...	
N200 **MEASA[X]=(31,1,-1)** G01 X100 F100	; Medición en modo 1 con ambos sistemas de medida. Esperar la señal de medición con flanco ascendente/descendente del palpador 1 en el recorrido hacia X=100.
N210 IF $AC_MEA[1]==FALSE GOTOF ENDE	; Comprobar si la medición se ha realizado correctamente.
N220 R10=$AA_MM1[X]	; Guardar la medida del sistema de medida 1 con flanco ascendente.
N230 R11=$AA_MM2[X]	; Guardar la medida del sistema de medida 2 con flanco ascendente.
N240 R12=$AA_MM3[X]	; Guardar la medida del sistema de medida 1 con flanco descendente.
N250 R13=$AA_MM4[X]	; Guardar la medida del sistema de medida 2 con flanco descendente.
N260 ENDE:	

Ejemplo 2: Medición específica de eje con borrado de trayecto residual en el modo 2 (evaluación en orden programado)

Código de programa	Comentarios
...	

Código de programa	**Comentarios**
N100 **MEASA[X]=(2,1,-1,2,-2)** G01 X100 F100	; Medición en modo 2 con el sistema de medida activo. Esperar la señal de medición en el orden: flanco ascendente del palpador 1; flanco descendente del palpador 1, flanco ascendente del palpador 2, flanco descendente del palpador 2 en el recorrido hacia X=100.
N110 IF $AC_MEA[1]==FALSE GOTOF MESSTASTER2	; Comprobar si la medición se ha realizado correctamente con el palpador 1.
N120 R10=$AA_MM1[X]	; Guardar la medida correspondiente al primer evento de disparo programado (flanco ascendente palpador 1).
N130 R11=$AA_MM2[X]	; Guardar la medida correspondiente al segundo evento de disparo programado (flanco ascendente palpador 1).
N140 MESSTASTER2:	
N150 IF $AC_MEA[2]==FALSE GOTOF ENDE	; Comprobar si la medición se ha realizado correctamente con el palpador 2.
N160 R12=$AA_MM3[X]	; Guardar la medida correspondiente al tercer evento de disparo programado (flanco ascendente palpador 2).
N170 R13=$AA_MM4[X]	; Guardar la medida correspondiente al cuarto evento de disparo programado (flanco ascendente palpador 2).
N180 ENDE:	

Ejemplo 3: Medición continua específica de eje en el modo 1 (evaluación en orden cronológico)

a) Medición de hasta 100 valores medidos

Código de programa	**Comentarios**
...	
N110 DEF REAL VALOR MEDIDO[100]	
N120 DEF INT Bucle=0	
N130 **MEAC[X]=(1,1,-1)** G01 X1000 F100	; Medición en modo 1 con el sistema de medida activo, guardar las medidas en $AC_FIFO1, esperar a señal de medición con flanco descendente del palpador 1 en el recorrido hacia X = 1000.
N135 STPRE	
N140 MEAC[X]=(0)	; Interrumpir la medición una vez alcanzada la posición del eje.
N150 R1=$AC_FIFO1[4]	; Guardar el número de valores medidos acumulados en el parámetro R1.
N160 FOR Bucle=0 TO R1-1	
N170 MEDIDA[Bucle]=$AC_FIFO1[0]	; Leer las medidas en el $AC_FIFO1 y guardarlas.
N180 ENDFOR	

b) Medición con borrado de trayecto residual al cabo de 10 valores medidos

```
Código de programa                                              Comentarios
...
N10 WHEN $AC_FIFO1[4]>=10 DO MEAC[x]=(0) DELDTG(x)      ; Borrar trayecto residual.
N20 MEAC[x]=(1,1,1,-1) G01 X100 F500
N30 MEAC[X]=(0)
N40 R1=$AC_FIFO1[4]                                     ; Número de medidas.
...
```

c) Medida de un flanco de dientes ascendente/descendente con 2 palpadores

```
Código de programa                          Comentarios
...
N110 DEF REAL VALOR MEDIDO[16]
N120 DEF INT Bucle=0
N130 MEAC[X]=(1,1,-1,2) G01 X100 F100       ; Medir en modo 1 con el sistema de medida
                                            activo, guardar las medidas en $AC_FIFO1,
                                            esperar a la señal de medición con la se-
                                            cuencia flanco descendente del palpador
                                            1, flanco ascendente del palpador 2, en
                                            el recorrido hacia X = 100.
N140 STOPRE                                 ; Parada de decodificación previa.
N150 MEAC[X]=(0)                            ; Interrumpir la medición una vez alcan-
                                            zada la posición del eje.
N160 R1=$AC_FIFO1[4]                        ; Guardar el número de valores medidos
                                            acumulados en el parámetro R1.
N170 FOR Bucle=0 TO R1-1
N180 MEDIDA[Bucle]=$AC_FIFO1[0]             ; Leer las medidas en el $AC_FIFO1 y guar-
                                            darlas.
N190 ENDFOR
```

Información adicional

Tarea de medición

La programación de una tarea de medición se puede realizar en el programa de pieza o desde una acción síncrona. Por cada eje solo puede estar activa una tarea de medición a la vez.

Nota

El avance se tiene que adaptar al problema de medición en cuestión.

Con MEASA y MEAWA, solo se pueden garantizar resultados correctos con avances en los que no se produzca más de un evento de disparo igual ni más de 4 eventos de disparo distintos por cada ciclo de regulador de posición.

En la medición continua con MEAC, la relación entre el ciclo de interpolador y el ciclo de regulador de posición no debe ser superior a 1:8.

Evento de disparo

Un evento de disparo se compone del número de palpador y del criterio de disparo (flanco ascendente o descendente) de la señal de medición.

Para cada medición se pueden procesar hasta 4 eventos de disparo de los palpadores activados, es decir, hasta dos palpadores con dos flancos de medición cada uno. El orden de procesamiento así como el número máximo de eventos de disparo depende del modo elegido.

Si la medición se efectúa con dos sistemas de medida, pueden programarse como máximo dos eventos de disparo (flanco ascendente o descendente). En cada uno de los dos eventos de disparo se capturan los valores medidos de ambos palpadores.

Cuando se utiliza el telegrama PROFIBUS 391 (ajuste estándar para la comunicación PROFIBUS), solo es posible un valor medido por cada evento de disparo y ciclo de regulador de posición.

Al utilizar el telegrama PROFIBUS 395 y con MEAC en el modo de operación 1, el número de valores medidos por cada evento de disparo puede incrementarse a un total de 8 valores medidos con flanco ascendente y 8 valores medidos con flanco descendente por cada evento de disparo y ciclo de regulador de posición:

- Un palpador: 8 valores medidos con flanco ascendente y 8 con flanco descendente
- Dos palpadores: 4 valores medidos con flanco ascendente y 4 con flanco descendente

De este modo con un telegrama PROFIBUS 395 pueden implementarse avances y velocidades de giro mayores.

Modo de operación

Con el primer número (década de decenas) del modo de operación se selecciona el sistema de medida deseado. Si sólo existe un sistema de medida y se programa el segundo, se utiliza automáticamente el existente.

Con el segundo número (década de unidades) se selecciona el modo de medición deseado. Con ello, el proceso de medida se adapta a las posibilidades del correspondiente control:

- Modo 1
  La evaluación de los eventos de disparo tiene lugar en el orden cronológico de su aparición.
- Modo 2
  La evaluación de los eventos de disparo tiene lugar en el orden programado.
- Modo 3
  La evaluación de los eventos de disparo tiene lugar en el orden programado, pero sin vigilancia del evento de disparo 1 en el arranque.

Medición con y sin borrado de trayecto residual

En la programación de MEASA, el borrado de trayecto residual solo se realiza una vez que se han capturado todos los valores medidos necesarios.

Para los procesos de medición especiales en los que se debe alcanzar la posición final programada en cada caso, se utiliza MEAWA.

Nota

MEASA no se puede programar en acciones síncronas. Alternativamente, MEAWA con borrado de trayecto residual se puede programar como acción síncrona.

Si la orden de medición con MEAWA se inicia desde las acciones síncronas, los valores medidos solo están disponibles en el sistema de coordenadas de máquina.

Ejes geométricos/transformadas

Para iniciar la medición específica de eje para un eje geométrico, se debe programar la misma orden de medición de forma explícita para todos los ejes geométricos restantes. Lo mismo rige para los ejes que participan en una transformada.

Ejemplos:
```
N10 MEASA[Z]=(1,1) MEASA[Y]=(1,1) MEASA[X]=(1,1) G0 Z100
```

o bien
```
N10 MEASA[Z]=(1,1) POS[Z]=100
```

Consulta del estado

Si el programa exige evaluar si un palpador se ha deflectado o ha conmutado, puede consultarse el estado mediante las siguientes variables de sistema:

Variable de sistema	Descripción	Tipo de datos	Valor	
$A_PROBE[<n>]	Estado del palpador	INT	0	Palpador no deflectado.
			1	Palpador deflectado.
$AC_MEA[<n>]	Estado de conmutación del palpador $AC_MEA[<n>] se resetea automáticamente al comienzo de la medición.	INT	0	El palpador no ha conmutado.
			1	El palpador ha conmutado (todos los eventos de disparo programados en la secuencia de medición se han producido).

<n> = Número de palpador

Nota

Si la medición se inicia desde acciones síncronas, $AC_MEA ya no se actualiza. En este caso, debe consultarse la señal de interfaz CN/PLC DB31, … DBX62.3 o la variable equivalente $AA_MEAACT[<Axis>]:

$AA_MEAACT==1: Medición activa

$AA_MEAACT==0: Medición no activa

Limitación del palpador

La variable de sistema $A_PROBE_LIMITED permite leer el estado de la limitación de palpador en el programa CN o en la acción síncrona mediante el telegrama PROFIBUS 395:

$A_PROBE_LIMITED[<n>] == 0: Limitación de palpador inactiva/reseteada

$A_PROBE_LIMITED[<n>] == 1: Limitación de palpador activa

<n> = Número del palpador

Resultados de medición para MEASA/MEAWA

Los valores medidos del palpador se pueden leer en el programa de pieza y en las acciones síncronas mediante las siguientes variables de sistema:

Variable de sistema	Descripción
$AA_MM1[<Axis>] ... $AA_MM4[<Axis>]	Valor medido del palpador con el evento de disparo 1 en el sistema de coordenadas de máquina ... Valor medido del palpador con el evento de disparo 4 en el sistema de coordenadas de máquina
$AA_MW1[<Axis>] ... $AA_MW4[<Axis>]	Valor medido del palpador con el evento de disparo 1 en el sistema de coordenadas de pieza ... Valor medido del palpador con el evento de disparo 4 en el sistema de coordenadas de pieza

<Axis> = Eje de medición

Si se ejecuta la orden de medición con dos sistemas de medida, ambos sistemas de medida capturan cada uno de los dos posibles eventos de disparo. La asignación de las variables de sistema es como sigue:

$AA_MM1[<Axis>]	o bien	$AA_MW1[<Axis>]	Valor medido del sistema de medida 1 con el evento de disparo 1
$AA_MM2[<Axis>]	o bien	$AA_MW2[<Axis>]	Valor medido del sistema de medida 2 con el evento de disparo 1
$AA_MM3[<Axis>]	o bien	$AA_MW3[<Axis>]	Valor medido del sistema de medida 1 con el evento de disparo 2
$AA_MM4[<Axis>]	o bien	$AA_MW4[<Axis>]	Valor medido del sistema de medida 2 con el evento de disparo 2

Medición continua (MEAC)

Con MEAC, los valores medidos están disponibles en el sistema de coordenadas de máquina y se guardan en la memoria FIFO[<n>] indicada (memoria cíclica). Si se han configurado dos palpadores para la medición, los valores medidos del segundo palpador se guardan por separado en la memoria FIFO[<n>+1] configurada adicionalmente al efecto (ajustable con DM).

La memoria FIFO es una memoria cíclica en la cual se introducen valores medidos en variables $AC_FIFO según el principio de ciclo.

Nota

El contenido FIFO solo se puede leer una vez de la memoria cíclica. Para el uso múltiple de los datos de medición, éstos se tienen que guardar en los datos de usuario.

Si el número de valores medidos para la memoria FIFO sobrepasa el máximo establecido en el dato de máquina, la medición se termina automáticamente.

Una medición infinita se puede realizar mediante la lectura cíclica de valores medidos. La lectura se tiene que realizar, por lo menos, con la misma frecuencia que la entrada de nuevos valores medidos.

Información adicional: Manual de funciones Acciones síncronas

Protección contra errores de programación

Las siguientes programaciones erróneas se detectan y se señalizan con un error:

- MEASA/MEAWA programado en una misma secuencia con MEAS/MEAW
  Ejemplo:
  ```
  N01 MEAS=1 MEASA[X]=(1,1) G01 F100 POS[X]=100
  ```
- MEASA/MEAWA con número de parámetros <2 o >5
  Ejemplo:
  ```
  N01 MEAWA[X]=(1) G01 F100 POS[X]=100
  ```
- MEASA/MEAWA con evento de disparo distinto de 1/–1/2/–2
  Ejemplo:
  ```
  N01 MEASA[B]=(1,1,3) B100
  ```
- MEASA/MEAWA con modo incorrecto
  Ejemplo:
  ```
  N01 MEAWA[B]=(4,1) B100
  ```
- MEASA/MEAWA con doble programación del evento de disparo
  Ejemplo:
  ```
  N01 MEASA[B]=(1,1,-1,2,-1) B100
  ```
- MEASA/MEAWA y falta de eje geométrico
  Ejemplo:
  ```
  N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) G01 X50 Y50 Z50 F100 ;Eje
  geométrico X/Y/Z
  ```
- Orden de medición no uniforme en ejes geométricos
  Ejemplo:
  ```
  N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) MEASA[Z]=(1,1,2) G01 X50 Y50 Z50
  F100
  ```

Consulte también

Acciones síncronas (Página 983)

3.7.9 Funciones especiales para el usuario de OEM (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829)

Direcciones OEM

El usuario de OEM determina el significado de las direcciones OEM. La funcionalidad se aporta a través de ciclos de compilación. 5 direcciones OEM están reservadas (`OMA1` ... `OMA5`). Los identificadores de dirección son ajustables. Las direcciones OEM se admiten en cada secuencia.

Llamadas reservadas a comandos G

Para el usuario de OEM están reservadas las siguientes llamadas a comandos G:

- `OEMIPO1`, `OEMIPO2` (del grupo G 1)
- `G810` ... `G819` (grupo G 31)
- `G820` ... `G829` (grupo G 32)

La funcionalidad se aporta a través de ciclos de compilación.

Funciones y subprogramas

Adicionalmente, los usuarios OEM pueden crear también funciones y subprogramas predefinidos con transferencia de parámetros.

Nota

Simulación de pieza

Hasta la versión de SW 4.4, no se admite ningún ciclo de compilación (CC) en la simulación de pieza; a partir de SW 4.4, solo ciclos de compilación seleccionados.

En consecuencia, los comandos de lenguaje del programa de pieza pertenecientes a ciclos de compilación no admitidos (`OMA1` ... `OMA5`, `OEMIPO1/2`, `G810` ... `G829`, funciones y procedimientos propios) provocan la emisión de un aviso de alarma y la cancelación de la simulación sin tratamiento individual.

Solución: tratar individualmente los elementos de lenguaje específicos de CC en el programa de pieza (consulta de $P_SIM).
Ejemplo:

```
N1 G01 X200 F500
IF (1==$P_SIM)
N5 X300 ; no activo en simulación CC
ELSE
N5 X300 OMA1=10
ENDIF
```

3.7.10 Reducción del avance con deceleración en los dos vértices (FENDNORM, G62, G621)

En la deceleración automática en vértices, el avance se reduce en forma de campana poco antes de la esquina en cuestión. Además, la medida del comportamiento de herramienta relevante para el mecanizado se puede parametrizar a través de datos de operador:

- Inicio y fin de la reducción del avance
- Corrección con la cual se reduce el avance
- Detección de la esquina relevante

Se consideran esquinas relevantes aquellas cuyo ángulo interior es menor que la esquina parametrizada a través del dato de operador.

Con el valor predeterminado FENDNORM se desactiva la función de la corrección automática de esquinas.

Sintaxis

```
FENDNORM
G62 G41
G621
```

Descripción

FENDNORM:	Deceleración automática en vértices DES
G62:	Deceleración en vértices **en esquinas interiores** con la corrección activa del radio de herramienta
G621:	Deceleración en vértices **en todas las esquinas** con la corrección activa del radio de herramienta

G62 solo actúa en las esquinas interiores con corrección activa del radio de herramienta G41/G42 y modo de contorneado activo G64/G641.

El posicionado en la esquina en cuestión se realiza con el avance reducido, que se calcula del modo siguiente:

F * (Corrección para la reducción del avance) * Corrección de avance

La máxima reducción de avance posible se alcanza exactamente cuando la herramienta tiene que realizar, con relación a la trayectoria del centro, el cambio de dirección en la esquina en cuestión.

G621 actúa de forma análoga a G62 en cada esquina de los ejes definidos por FGROUP.

3.7.11 Criterio de fin del movimiento programable (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)

De un modo similar al criterio de cambio de secuencia con la interpolación de la trayectoria (G601, G602 y G603), el criterio de fin del movimiento en la interpolación de eje individual se puede programar en un programa de pieza o en acciones síncronas para ejes de comando/ PLC.

Dependiendo del criterio de fin del movimiento que se haya establecido, se finalizarán con mayor o menor rapidez las secuencias de programa de pieza o las secuencias de ciclo tecnológico con desplazamientos del eje individual. Lo mismo se aplica con respecto al PLC a través de FC15/16/18.

Sintaxis

```
FINEA[<Eje>]
COARSEA[<Eje>]
IPOENDA[<Eje>]
IPOBRKA(<Eje>[,<Instante>])
ADISPOSA(<Eje>[,<Modo>,<Tamaño ventana>])
```

Descripción

FINEA:	Criterio de fin del movimiento: "Parada precisa fina"		
	Eficacia:	modal	
COARSEA:	Criterio de fin del movimiento: "Parada precisa basta"		
	Eficacia:	modal	
IPOENDA:	Criterio de fin de movimiento: "Parada interpolador"		
	Eficacia:	modal	
IPOBRKA:	Criterio de cambio de secuencia: Rampa de frenado		
	Eficacia:	modal	
ADISPOSA:	Ventana de tolerancia con relación al criterio de fin del movimiento		
	Eficacia:	modal	
<Eje>:	Nombre de eje de canal (X, Y, ...)		
<Instante>:	Instante del cambio de secuencia, referido a la rampa de frenado en %: • 100% = Inicio de la rampa de frenado • 0% = Fin de la rampa de frenado, con el mismo significado que IPOENDA		
	Tipo:	REAL	
<Modo>:	Referencia de la ventana de tolerancia		
	Rango de valores:	0	Ventana de tolerancia no activa
		1	Ventana de tolerancia con relación a la posición teórica
		2	Ventana de tolerancia con relación a la posición real
	Tipo:	INT	

<Tamaño ventana>:	Tamaño de la ventana de tolerancia	
	Tipo:	REAL

Ejemplos

Ejemplo 1: Criterio de fin del movimiento: "Parada interpolador"

Código de programa

```
; Desplazar eje de posicionado X a 100, velocidad 200 m/
min, aceleración 90%,
; Criterio de fin del movimiento: Parada interpolador
N110 G01 POS[X]=100 FA[X]=200 ACC[X]=90 IPOENDA[X]

; Acción síncrona:
; SIEMPRE QUE: la entrada 1 esté ajustada,
; ENTONCES Desplazar eje de posicionado X a 50, veloci-
dad 200 m/min, aceleración 140%,
; Criterio de fin del movimiento: Parada interpolador
N120 EVERY $A_IN[1] DO POS[X]=50 FA[X]=200 ACC[X]=140
IPOENDA[X]
```

Ejemplo 2: Criterio de cambio de secuencia: "Rampa de frenado"

Código de programa	Comentarios
	; Ajuste por defecto activo
N40 POS[X]=100	; Desplazamiento de posicionado de X a la posición 100
	Criterio de cambio de secuencia: Parada precisa fina
N20 IPOBRKA(X,100)	; Criterio de cambio de secuencia: "Rampa de frenado",
	100% = Inicio de la rampa de frenado
N30 POS[X]=200	; El cambio de secuencia se efectúa en cuanto el eje X empieza a frenar
N40 POS[X]=250	; El eje X ya no frena en la posición 200, sino que continúa hasta la posición 250.
	En cuanto el eje empieza a frenar, se efectúa el cambio de secuencia.
N50 POS[X]=0	; El eje X frena y vuelve a la posición 0.
	El cambio de secuencia se produce en la posición 0 y la "parada precisa fina"
N60 X10 F100	; El eje X se desplaza a la posición 10 como eje de contorneado

Información adicional

Variable de sistema para el criterio de fin del movimiento

El criterio efectivo de fin del movimiento puede leerse a través de la variable de sistema $AA_MOTEND.

Criterio de cambio de secuencia: "Rampa de frenado" (IPOBRKA)

Si durante la activación del criterio de cambio de secuencia "Rampa de frenado" se programa un valor para el instante del cambio de secuencia opcional, este valor es efectivo para el siguiente desplazamiento de posicionado y se escribe en el dato de operador de forma sincronizada con el proceso principal. Si no se indica ningún valor para el instante del cambio de secuencia, es efectivo el valor actual del dato de operador.

SD43600 $SA_IPOBRAKE_BLOCK_EXCHANGE

Con la próxima programación de un criterio de fin del movimiento axial (FINEA, COARSEA, IPOENDA) se desactiva IPOBRKA para el eje correspondiente.

Criterio adicional de cambio de secuencia: "Ventana de tolerancia" (ADISPOSA)

Con ADISPOSA puede definirse como criterio adicional de cambio de secuencia una ventana de tolerancia alrededor del punto final de la secuencia (posición real o teórica, a elección). En ese caso, para el cambio de secuencia deben cumplirse las dos condiciones:

- Criterio de cambio de secuencia: "Rampa de frenado"
- Criterio de cambio de secuencia: "Ventana de tolerancia"

3.8 Transformaciones de coordenadas (frames)

3.8.1 Transformada de coordenadas a través de variables frame

Para transformar el sistema de coordenadas de pieza (WKS), además de las instrucciones frame (Página 316) como, p. ej., ROT, AROT o SCALE, pueden usarse también las variables frame $P_...FR (frames de gestión de datos) y $P_...FRAME (frames activos).

La siguiente figura muestra una vista general de la estructura de las variables frame:

- Frames de gestión de datos
- Frames activos
- Frame total activo: concatenación de todos los frames activos
- Frames globales NCU
- Frames específicos del canal

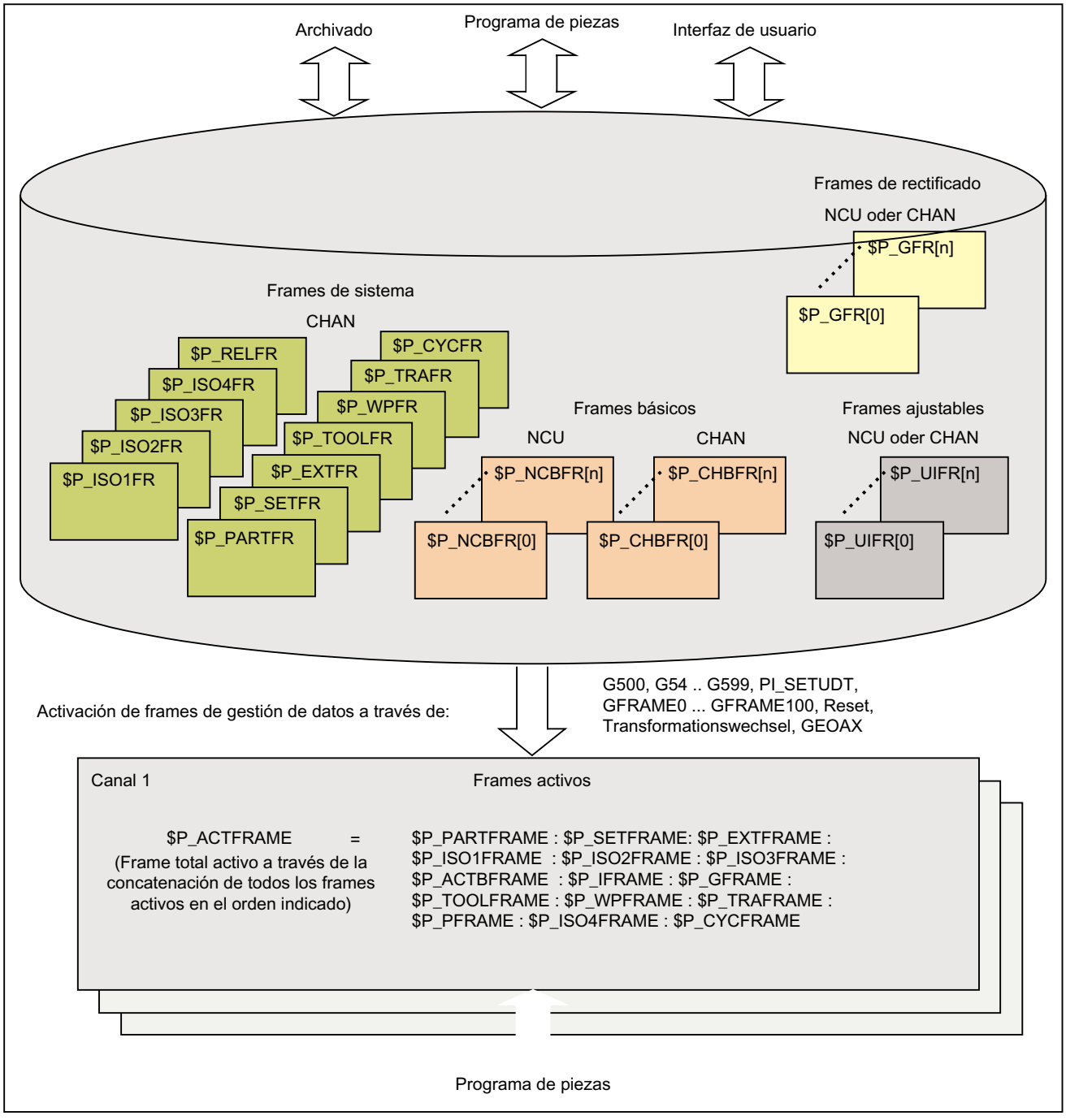


Figura 3-1 Vista general de las variables frame

3.8.1.1 Variable frame predefinida ($P_CHBFRAME, $P_IFRAME, $P_PFRAME, $P_ACTFRAME)

Activo: frames básicos específicos de canal $P_CHBFRAME[<n>] ($P_BFRAME)

Nota

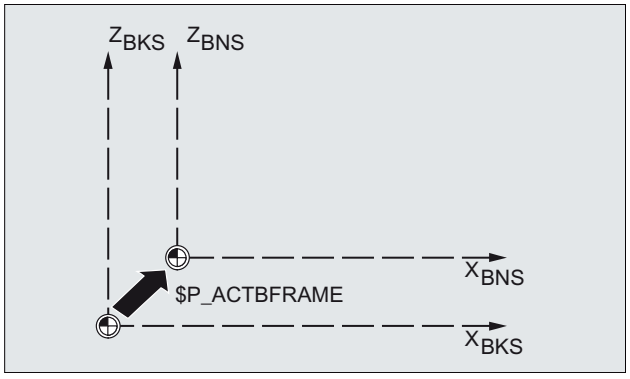
El frame básico actual `$P_BFRAME` y el frame básico de gestión de datos `$P_UBFR` se mantienen por motivos de compatibilidad.

- `$P_BFRAME` ≙ `$P_CHBFRAME[0]`
- `$P_UBFR` ≙ `$P_CHBFR[0]`.

Las variables frame $P_CHBFRAME[<n>] definen la relación entre el sistema de coordenadas básico (BKS) y el sistema de origen básico (BNS).

Si se desea activar el frame básico actual específico de canal $P_CHBFRAME[`<n>`] de manera inmediata en el programa CN, existen las siguientes posibilidades:

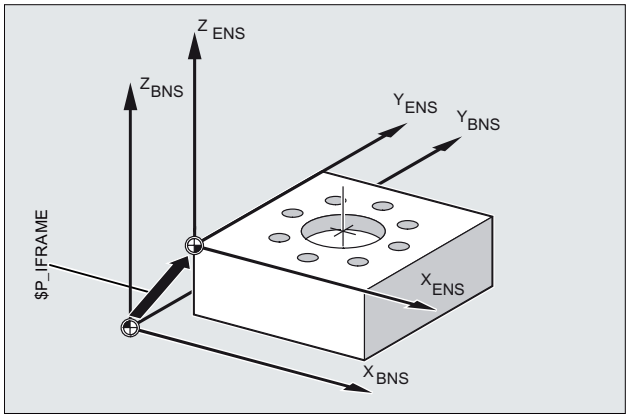
- Comandos:
  - `G500` (desactivar todos los frames ajustables; los frames básicos permanecen activos)
  - `G54` ... `G599` (decalajes de origen ajustables)
- Asignación de un frame básico de gestión de datos específico de canal a un frame básico actual específico de canal:
  `$P_CHBFRAME[<n>] = $P_CHBFR[<m>]`



Activo: frame ajustable específico de canal $P_IFRAME

La variable frame $P_IFRAME define la relación entre el sistema de origen básico (BNS) y el sistema de origen ajustable (ENS).

- `$P_IFRAME` corresponde a `$P_UIFR[$P_IFRNUM]`
- `$P_IFRAME` contiene después de la programación (p. ej., de `G54`) la traslación, rotación, escala y simetría especular definidas por `G54`.

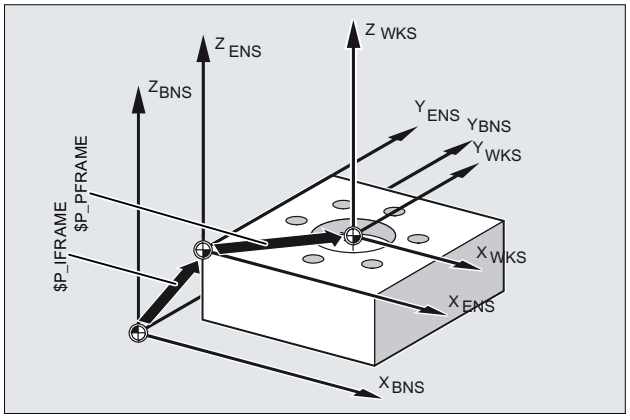


Activo: frame programable específico de canal $P_PFRAME

La variable frame $P_PFRAME define la relación entre el sistema de origen ajustable (ENS) y el sistema de coordenadas de pieza (WKS).

`$P_PFRAME` contiene el frame resultante

- **de la programación** de `TRANS/ATRANS`, `ROT/AROT`, `SCALE/ASCALE`, `MIRROR/AMIRROR` o
- **de la asignación** de `CTRANS`, `CROT`, `CMIRROR`, `CSCALE` al FRAME programable



Activo: frame total $P_ACTFRAME

El frame total activo en el canal se obtiene de la concatenación de todos los frames activos en el 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` describe el origen del sistema de coordenadas de pieza actual.

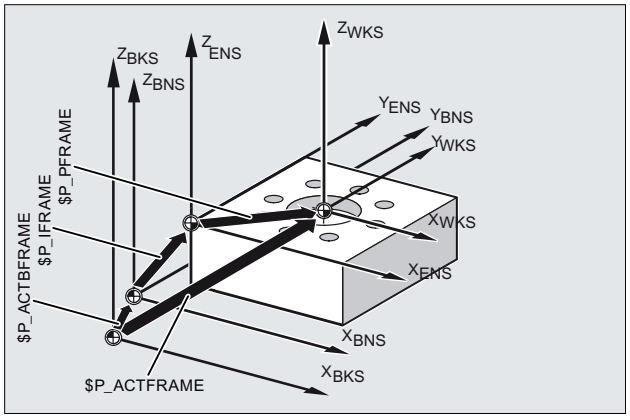
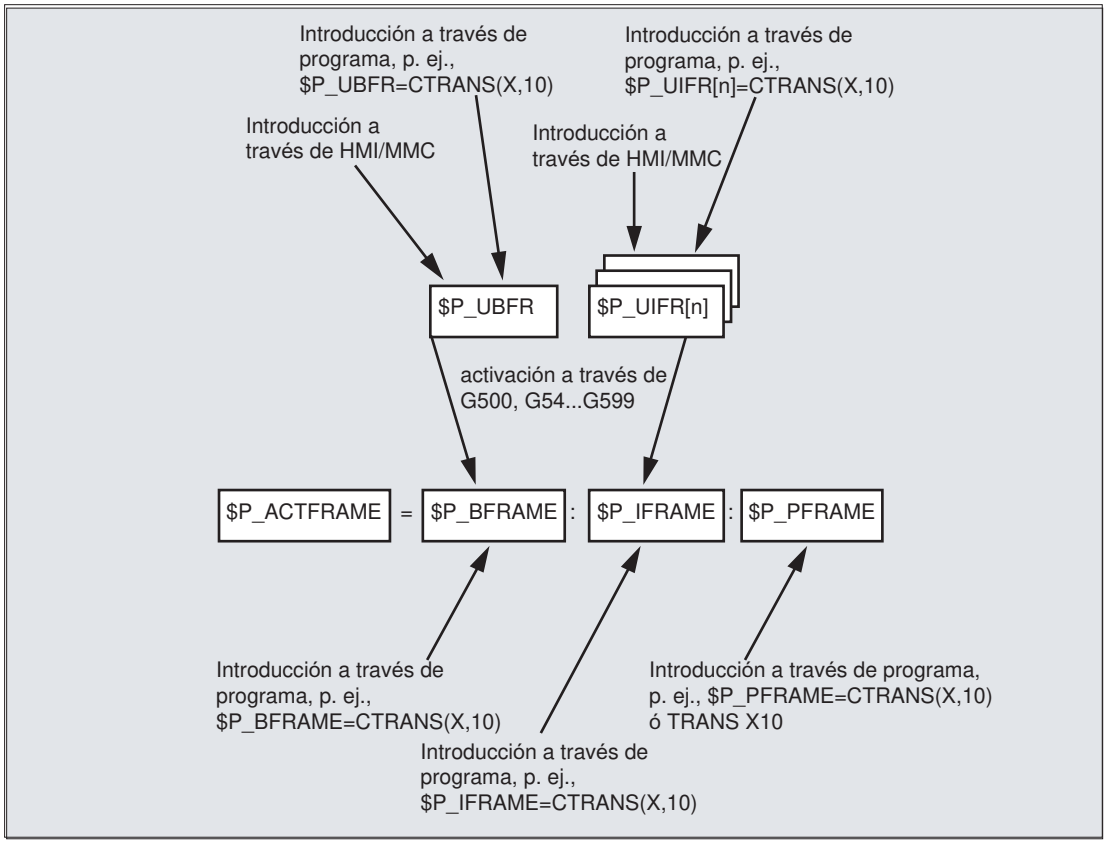


Figura 3-2 Variable frame $P_ACTFRAME

Si se modifica uno de los frames `$P_BFRAME`/`$P_CHBFRAME[<n>]`, `$P_IFRAME` o `$P_PFRAME`, se recalcula el frame total actual `$P_ACTFRAME`.



El frame básico y el frame ajustable actúan tras el reset si el MD20110 RESET_MODE_MASK ha sido ajustado de la siguiente manera:

Bit0=1, bit14=1 --> `$P_UBFR` (frame básico) activo

Bit0=1, bit5=1 --> `$P_UIFR[$P_UIFRNUM]` (frame ajust.) activo

Gestión de datos: frames básicos específicos de canal $P_CHBFR[<n>]

Las variables frame $P_CHBFR[<n>] permiten leer y escribir los frames básicos en la gestión de datos. El frame de gestión de datos no se activa inmediatamente en el canal al escribirlo. La activación del frame escrito se realiza en:

- Reset del canal y MD20110 $MC_RESET_MODE_MASK, Bit0 == 1 y Bit14 == 1
- Comandos `G500`, `G54` ... `G57`, `G505` ... `G599` (activación y desactivación de los frames básicos con nuevo cálculo inmediato del frame total actual)

Gestión de datos: frames ajustables específicos de canal $P_UIFR[<n>]

Las variables frame $P_UIFR[<n>] permiten leer y escribir los frames ajustables en la gestión de datos. El frame no se activa inmediatamente en el canal al escribirlo. La inclusión del frame escrito en el canal se realiza en:

- Comando `G500` (desactivación de todos los frames o decalajes de origen ajustables)
- Comandos `G54` ... `G57`, `G505` ... `G599` (activación de un frame ajustable o un decalaje de origen)

Frame ajustable activo	Frame de gestión de datos	(equivale a comando)
`$P_IFRAME` =	`$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 Asignación de valores a frames

3.8.2.1 Asignar valores directos (valor de eje, ángulo, escala)

Se pueden asignar valores directamente en el programa CN a frames o variables frame.

Sintaxis

Sintaxis

```
$P_PFRAME = CTRANS(X, <Valor de decalaje>, Y, <Valor de decalaje>, Z,
<Valor de decalaje>, ...)
$P_PFRAME = ROT(X, <Ángulo>, Y, <Ángulo>, Z, <Ángulo>, ...)
$P_UIFR[..] = CROT(X, <Ángulo>, Y, <Ángulo>, Z, <Ángulo>, ...)
```

```
$P_PFRAME = CSCALE(X, <Escala>, Y, <Escala>, Z, <Escala>, ...)
$P_PFRAME = CMIRROR(X, Y, Z)
```

La sintaxis de `$P_CHBFRAME[<n>]` es idéntica a la de `$P_PFRAME`.

Descripción

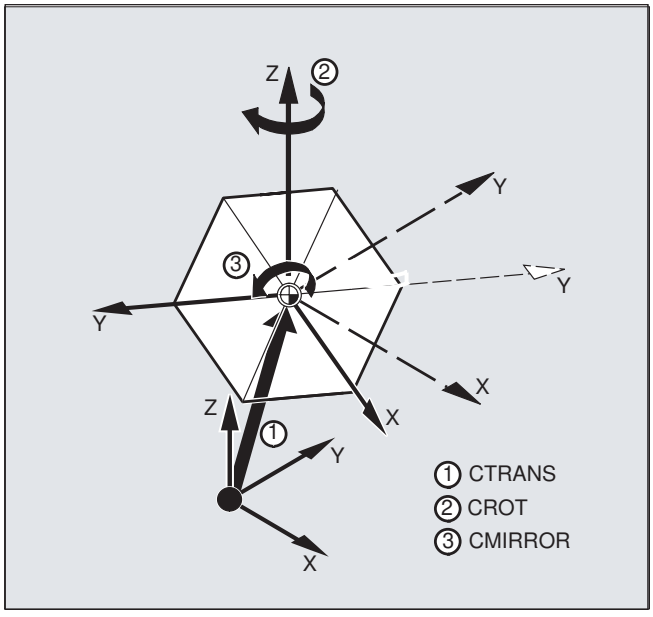
`CTRANS`:	Desplazamiento en los ejes indicados
`CROT`:	Rotación alrededor del eje indicado
`CSCALE`:	Cambio de escala en los ejes indicados
`CMIRROR`:	Inversión de sentido de los ejes indicados
`X, Y, Z`:	Valores de decalaje a lo largo de los ejes geométricos indicados
`<Valor de decalaje>`:	Valor de decalaje
`<Ángulo>`:	Ángulo que se gira
`<Escala>`:	Indicación de escala

Ejemplos

Asignaciones de valor a componentes del frame programable actual

Asignación de valor a los componentes de frame traslación, rotación y simetría especular del frame programable actual:

```
$P_PFRAME = CTRANS(X,10,Y,20,Z,5) : CROT(Z,45) : CMIRROR(Y)
```



Escribir los componentes de rotación de un frame

Asignación de valores a los tres ejes del componente de rotación del frame de gestión de datos ajustable `$P_UIFR` con `CROT`:

```
$P_UIFR[5] = CROT(X, 0, Y, 0, Z, 0)
```

También es posible asignar directamente los valores individuales al eje correspondiente del componente de rotación del frame de gestión de datos:

```
$P_UIFR[5, Y, RT]=0
$P_UIFR[5, X, RT]=0
$P_UIFR[5, Z, RT]=0
```

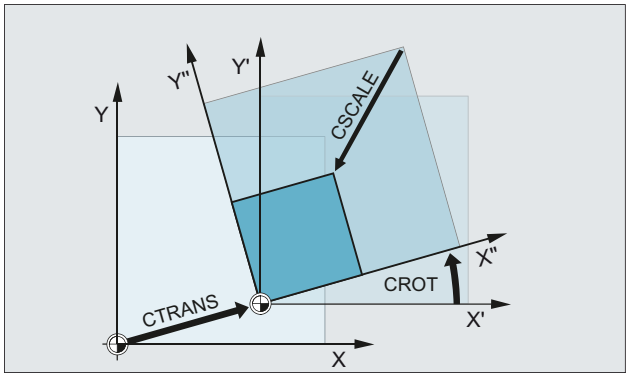
Descripción

Es posible usar el operador de concatenación : para unir entre sí varias operaciones en un frame. Tras ello, las operaciones se ejecutarán consecutivamente de izquierda a derecha.

Ejemplo

Encadenar operaciones en $P_PFRAME con decalaje, rotación y escala:

```
$P_PFRAME = CTRANS(...) : CROT(...) : CSCALE...
```



3.8.2.2 Lectura y modificación de componentes de frame (TR, FI, RT, SC, MI)

Tiene la posibilidad de acceder a datos **individuales** de un frame, p. ej., a un determinado valor de decalaje o ángulo de giro. Estos valores se pueden modificar o asignar a otra variable.

Sintaxis

`R10=$P_UIFR[$P_UIFNUM,X,RT]`	El ángulo de giro RT alrededor del eje X desde el decalaje de origen ajustable actualmente válido $P_UIFRNUM se debe asignar a la variable R10.
`R12=$P_UIFR[25,Z,TR]`	El valor de decalaje TR en Z del registro de datos del frame ajustado núm. 25 se debe asignar a la variable R12.
`R15=$P_PFRAME[Y,TR]`	El valor de decalaje TR en Y del frame programable actual se debe asignar a la variable R15.
`$P_PFRAME[X,TR]=25`	El valor de decalaje TR en X del frame programable actual debe modificarse. A partir de ahora es X25.

Descripción

$P_UIFRNUM:	Con esta variable se establece automáticamente la relación con el decalaje de origen ajustable actualmente válido.
P_UIFR[n,…,…]:	Especificando el número de frame n se accede al frame ajustable número n.
	Especificación del componente que se desea leer o modificar:
TR:	TR Translation
FI:	FI Translation Fine
RT:	RT Rotation
SC:	SC Scale cambio de escala
MI:	MI Simetría especular
X, Y, Z:	Además (ver ejemplos) se indica el eje correspondiente X, Y, Z.

Rango de valores para rotación RT

Giro alrededor del 1er eje geométri- co: -180° a +180°

Giro alrededor del 2º eje geométrico: -90° a +90°

Giro alrededor del 3.er eje geométri- co: -180° a +180°

Descripción

Llamar un frame

Indicando la variable de sistema $P_UIFRNUM se puede acceder directamente al decalaje de origen ajustado actualmente con $P_UIFR o G54, G55, …
($P_UIFRNUM contiene el número del frame actualmente ajustado).

Todos los demás frames ajustables almacenados $P_UIFR se llaman indicando el correspondiente número $P_UIFR[n].

Para variables frame predefinidas y frames de definición propia se indica el nombre, p. ej., $P_IFRAME.

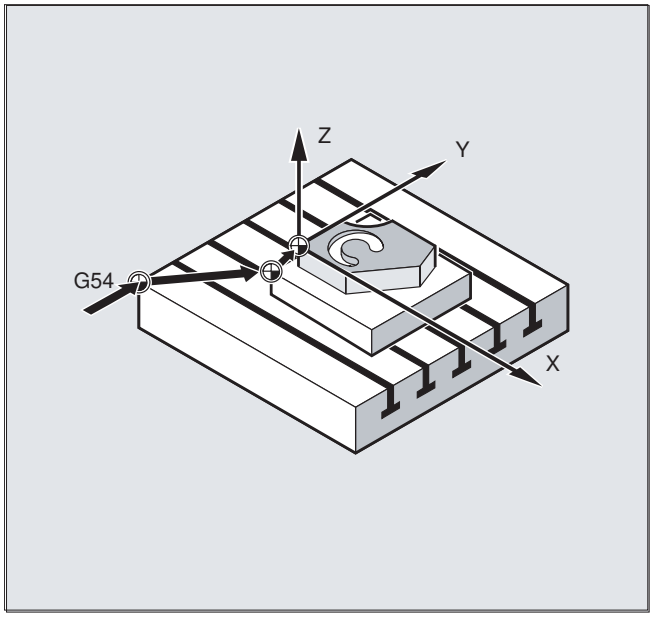
Llamar datos

Entre corchetes se encuentran el nombre del eje y el componente de frame del valor al cual quiere acceder o que quiere modificar, p. ej., [X, RT] o [Z, MI].

3.8.2.3 Calcular con frames

En el programa CN es posible asignar un frame a otro frame o concatenar frames.

Las concatenaciones de frames se prestan, por ejemplo, para definir varias piezas dispuestas sobre una paleta y que deben mecanizarse en un proceso de fabricación.



Para definir las operaciones de paleta, los componentes de frame podrían contener, p. ej., sólo determinados valores parciales, con cuya concatenación se generan diferentes decalajes de origen de las piezas.

Ejemplos

Asignaciones

Código de programa	**Comentarios**
`DEF FRAME AJUSTE_1`	`; Definición de una variable frame local`
`AJUSTE_1 = CTRANS(X,10)`	`; Asignación del resultado de una función a la variable frame`
`$P_PFRAME = AJUSTE_1`	`; Asignación de la variable frame al frame actual`
`DEF FRAME AJUSTE_4`	`; Definición de una variable frame local`
`AJUSTE_4 = $P_PFRAME`	`; Almacenamiento intermedio del frame actual en la variable frame`
`...`	
`$P_PFRAME = AJUSTE_4`	`; Lectura del frame actual desde la variable frame`

Concatenaciones

Los frames se concatenan mediante el operador : en el orden programado. Los componentes de frame, como, p. ej., decalajes o rotaciones, se ejecutan consecutivamente de modo aditivo.

Código de programa	Comentarios

`$P_IFRAME = $P_UIFR[15] : $P_UIFR[16]`	`; Asignación del frame de resultado de la concatenación de`
	`; los dos frames ajustables de gestión de datos al`
	`; frame total ajustable.`
	`; Ejemplo de aplicación:`
	`; $P_UIFR[15]: Decalaje`
	`; $P_UIFR[16]: Rotación`

`$P_UIFR[3] = $P_UIFR[4] : $P_UIFR[5]`	`; Asignación del frame de resultado de la concatenación de`
	`; los dos frames ajustables de gestión de datos a`
	`; otro frame ajustable de gestión de datos`

3.8.2.4 Definición de variables frame (DEF FRAME)

Además de las variables frame predefinidas, pueden definirse también variables frame propias. Las variables frame definidas por el usuario son variables del usuario del tipo FRAME. El nombre del frame puede asignarse libremente siempre que se cumplan las reglas válidas para las variables de usuario.

Las funciones CTRANS, CROT, CSCALE y CMIRROR permiten asignar las variables frame definidas por el usuario a valores.

Sintaxis

```
DEF FRAME <Nombre>
```

Descripción

`DEF FRAME`:	Definir variables de usuario del tipo FRAME.
<Nombre>:	Nombre de la variable frame

Ejemplo

Definición de una variable frame "PALET" y asignación de valores de decalaje y rotación:

Código de programa	Comentarios
`DEF FRAME PALET`	`; Definición de la variable frame PALET`
`PALET = CTRANS(...) : CROT(...)`	`; Asignación del frame de resultado de la concatenación de` `; decalaje y rotación a la variable frame PALET`

3.8.3 Decalaje basto y fino (CTRANS, CFINE)

Decalaje fino

Un decalaje fino `CFINE(...)` puede aplicarse a los siguientes frames:

- Frames ajustables: $P_UIFR o $P_IFRAME
- Frames básicos: $P_NCBFR[<n>], $P_CHBFR[<n>], $P_CHBFRAMES[<n>] o $P_ACTBFRAME
- Frame programable: $P_PFRAME

El decalaje fino de un frame se programa con el comando `CFINE(...)`.

Decalaje basto

El decalaje basto `CTRANS(...)` puede aplicarse a todos los frames.

Decalaje total

El decalaje total se obtiene sumando el decalaje basto y el decalaje fino.

Datos de máquina

Habilitación del decalaje fino

El decalaje fino se habilita con el dato de máquina:

MD18600 $MN_MM_FRAME_FINE_TRANS = 1

Sintaxis

Decalaje fino

- Frame total
  - `<Frame> = CFINE(<K_1>,<Valor>)`
  - `<Frame> = CFINE(<K_1>,<Valor>, <K_2>, <Valor>)`
  - `<Frame> = CFINE(<K_1>,<Valor>, <K_2>, <Valor>, <K_3>, <Valor>)`
- Componente de frame
  - `<Frame>[<n>, <K_1>, FI] = <Valor>`

Decalaje basto

- Frame total
  - `<Frame> = CTRANS(<K_1>,<Valor>)`
  - `<Frame> = CTRANS(<K_1>,<Valor>, <K_2>,<Valor>)`
  - `<Frame> = CTRANS(<K_1>,<Valor>, <K_2>,<Valor>, <K_3>,<Valor>)`
- Componente de frame
  - `<Frame>[<n>,<K_1>,TR] = <Valor>`

Especialmente para el frame programable `$P_PFRAME`:

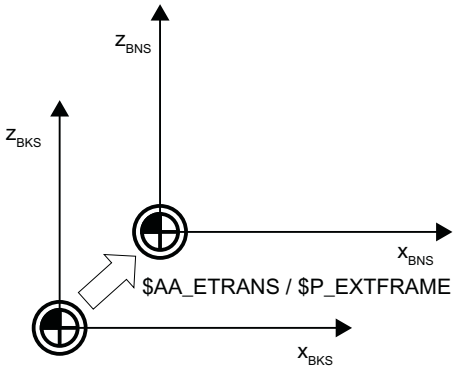
- `TRANS <K_1> <Valor>`
- `TRANS <K_1> <Valor> <K_2> <Valor>`
- `TRANS <K_1> <Valor> <K_2> <Valor> <K_3> <Valor>`

Descripción

`<Frame>`:	Frame, p. ej., frame ajustable de gestión de datos `$P_UIFR[<n>]`
`CFINE`:	Decalaje fino, decalaje aditivo.
`CTRANS`:	Decalaje basto, decalaje absoluto.
`TRANS`:	Solo frame programable: decalaje basto, decalaje absoluto.
`<K_n>`:	Ejes de coordenadas X, Y, Z
`<Valor>`:	Valor de decalaje

3.8.4 Decalaje de origen externo ($AA_ETRANS)

El decalaje de origen externo es un decalaje lineal entre el sistema de coordenadas básico (BKS) y el sistema de origen básico (BNS).



En función de la parametrización de los datos de máquina, el decalaje de origen externo mediante $AA_ETRANS actúa de dos maneras:

1. La variable de sistema $AA_ETRANS actúa directamente como valor de decalaje tras la activación a través de la señal de interfaz CN/PLC
2. Tras la activación a través de la señal de interfaz CN/PLC, el valor de la variable de sistema $AA_ETRANS se envía a los frames de sistema activos $P:EXTFRAME y al frame de gestión de datos $P_EXTFR. A continuación se calcula de nuevo el frame total activo $P_ACTFRAME.

Datos de máquina

En el caso de la variable de sistema $AA_ETRANS debe distinguirse entre dos procedimientos, que se eligen mediante el siguiente dato de máquina:

MD28082 $MC_MM_SYSTEM_FRAME_MASK,Bit1 = <Valor>

<Valor>	Descripción
0	Función: escritura directa de $AA_ETRANS[<Eje>] desde PLC, HMI o programa CN. Habilitación para la retirada del decalaje de origen de $AA_ETRANS[<Eje>] en la siguiente secuencia de desplazamiento posible: DB31, ... DBX3.0
1	Función: activación del frame de sistema activo $P:EXTFRAME y del frame de gestión de datos $P_EXTFR Habilitación para la retirada del decalaje de origen de $AA_ETRANS[<Eje>] mediante: DB31, ... DBX3.0. Tras ello, se produce en el canal: • la parada de todos los desplazamientos en el canal (excepto los ejes de comando y de PLC) • la parada de decodificación previa con posterior reorganización (STOPRE) • el decalaje basto de frame activo $P_EXTFRAME[<Eje>] = $AA_ETRANS[<Eje>] • el decalaje basto de frame de gestión de datos $P_EXTFR[<Eje>] = $AA_ETRANS[<Eje>] • Nuevo cálculo del frame total activo $P_ACTFRAME • Retirada del decalaje en los ejes programados. • Continuación del desplazamiento interrumpido o del programa CN

Programación

- Sintaxis
  `$AA_ETRANS[<Eje>] = <Valor>`
- Descripción

`$AA_ETRANS`:	Variable de sistema para el almacenamiento intermedio del decalaje de origen externo
`<Eje>`:	Eje de canal
`<Valor>`:	Valor de decalaje

Señal de la interfaz CN/PLC

DB31, ... DBX3.0 = 0 → 1 ⇒ $P_EXTFRAME[<Eje>] = $P_EXTFR[<Eje>] = $AA_ETRANS[<Eje>]

3.8.5 Ajuste de valor real con pérdida del estado de referenciado (PRESETON)

El procedimiento PRESETON() ajusta un nuevo valor real para uno o varios ejes en el sistema de coordenadas de la máquina (MKS), lo equivale a un decalaje de origen del MKS del eje. Esto no provoca el desplazamiento del eje.

Mediante PRESETON se realiza una parada de decodificación previa con sincronización. No se asigna la posición real al eje hasta que este está parado.

Si, con PRESETON, el eje no está asignado al canal, el procedimiento posterior dependerá de la configuración específica de eje del comportamiento de intercambio de eje:

MD30552 $MA_AUTO_GET_TYPE

Estado de referenciado

Al ajustar un nuevo valor real en el sistema de coordenadas de la máquina, se anula el estado de referenciado del eje de máquina:

DB31, ... DBX60.4/.5 = 0 (referenciado/sincronizado sistema de medida 1/2)

Se recomienda utilizar PRESETON solo con ejes que no requieran punto de referencia.

Para restablecer el sistema de coordenadas de máquina original, debe volver a referenciarse el sistema de medida del eje de máquina, p. ej., mediante una búsqueda del punto de referencia desde el programa de pieza (G74).

⚠ PRECAUCIÓN
Pérdida del estado de referenciado
Al ajustar un nuevo valor real en el sistema de coordenadas de máquina con PRESETON, el estado de referenciado del eje de máquina se restablece a "no referenciado/sincronizado".

Programación

Sintaxis

```
PRESETON(<Eje_1>, <Valor_1> [, <Eje_2>, <Valor_2>, ... <Eje_8>,
<Valor_8>])
```

Descripción

PRESETON:	Ajuste de valor real con pérdida del estado de referenciado	
	Parada de decodificación previa:	sí
	Único elemento de la secuencia:	sí
<Eje_x>:	Nombre del eje de máquina	
	Tipo:	AXIS
	Rango de valores:	Nombres de eje de máquina definidos en el canal
<Valor_x>:	Nuevo valor real del eje de máquina en el sistema de coordenadas de máquina (MKS) Se introduce en el sistema de medida válido actualmente (pulgadas/métrico) Si hay una programación por diámetro activa (DIAMON), se tendrá en cuenta	
	Tipo:	REAL

Información adicional

PRESETON en programas CN

Encontrará una descripción detallada de PRESETON en programas CN en el Manual de funciones Funciones básicas.

PRESETON en acciones síncronas

Encontrará una descripción detallada de PRESETON en acciones síncronas en el Manual de funciones Acciones síncronas.

3.8.6 Ajuste de valor real sin pérdida del estado de referenciado (PRESETONS)

El procedimiento PRESETONS() ajusta un nuevo valor real para uno o varios ejes en el sistema de coordenadas de la máquina (MKS), lo equivale a un decalaje de origen del MKS del eje. Esto no provoca el desplazamiento del eje.

Mediante PRESETONS se realiza una parada de decodificación previa con sincronización. No se asigna la posición real al eje hasta que este está parado.

Si, con PRESETONS, el eje no está asignado al canal, el procedimiento posterior dependerá de la configuración específica de eje del comportamiento de intercambio de eje:

MD30552 $MA_AUTO_GET_TYPE

Estado de referenciado

Al ajustar un nuevo valor real en el sistema de coordenadas de la máquina (MKS) con PRESETONS, el estado de referenciado del eje de máquina **no** varía.

Requisitos

- **Tipo de encóder**
  PRESETONS solo es posible con los siguientes tipos de encóder del sistema de medida activo:
  - MD30240 $MA_ENC_TYPE[<Sistema de medida>] = 0 (encóder simulado)
  - MD30240 $MA_ENC_TYPE[<Sistema de medida>] = 1 (encóder de señal bruta)
- **Modo de referenciado**
  PRESETONS solo es posible con los siguientes modos de referenciado del sistema de medida activo:
  - MD34200 $MA_ENC_REFP_MODE[<Sistema de medida>] = 0 (no es posible la búsqueda de punto de referencia)
  - MD34200 $MA_ENC_REFP_MODE[<Sistema de medida>] = 1 (referenciado de sistemas de medida incrementales, giratorios o lineales: impulso cero en la pista de encóder)

Programación

Sintaxis

```
PRESETONS(<Eje_1>, <Valor_1> [, <Eje_2>, <Valor_2>, ... <Eje_8>,
<Valor_8>])
```

Descripción

PRESETONS:	Ajuste de valor real sin pérdida del estado de referenciado	
	Parada de decodificación previa:	sí
	Único elemento de la secuencia:	sí
<Eje_x>:	Nombre del eje de máquina	
	Tipo:	AXIS
	Rango de valores:	Nombres de eje de máquina definidos en el canal
<Valor_x>:	Nuevo valor actual del eje de máquina en el sistema de coordenadas de máquina (MKS) Se introduce en el sistema de medida activo (pulgadas/métrico) Si hay una programación por diámetro activa (DIAMON), se tendrá en cuenta	
	Tipo:	REAL

Información adicional

PRESETONS en programas CN

Encontrará una descripción detallada de PRESETONS en programas CN en el Manual de funciones Funciones básicas.

PRESETONS en acciones síncronas

Encontrará una descripción detallada de PRESETONS en acciones síncronas en el Manual de funciones Acciones síncronas.

3.8.7 Cálculo del frame a partir de 3 puntos de medida en el espacio (MEAFRAME)

La función MEAFRAME se utiliza para facilitar los ciclos de medida. La función calcula el frame a partir de los tres puntos ideales y de los tres puntos medidos asociados.

Cuando se posiciona una pieza para su mecanizado, su posición respecto al sistema de coordenadas cartesiano de la máquina está generalmente caracterizada por desplazamientos y giros respecto a su posición ideal. Para un exacto mecanizado o medición es necesario o bien un ajuste físico, operación costosa, o modificar las instrucciones de movimiento del programa de pieza.

Un frame puede definirse palpando tres puntos en el espacio de los que se conocen sus posiciones ideales. Se palpa con un sensor de contacto u óptico que toca agujeros o esferas de medición fijados con precisión en la placa soporte.

Sintaxis

```
MEAFRAME(<Puntos_ideales>,<Puntos_medida>,<Calidad>)
```

Descripción

MEAFRAME:	Llamada de función		
<Puntos_ideales>:	Matriz REAL de 2 dimensiones que incluye las tres coordenadas de los puntos ideales		
<Puntos de medida>:	Matriz REAL de 2 dimensiones que incluye las tres coordenadas de los puntos medidos		
<Calidad>:	Variable con la que se devuelve información sobre la calidad del cálculo de FRAME		
	Tipo:	VAR REAL	
	Valor:	-1	Los puntos ideales se sitúan prácticamente en una recta. No ha sido posible calcular el frame. La variable FRAME devuelta contiene un frame neutro.
		-2	Los puntos de medida se sitúan prácticamente en una recta. No ha sido posible calcular el frame. La variable FRAME devuelta contiene un frame neutro.
		-4	El cálculo de la matriz de rotación fracasa por otra razón.
		≥ 0.0	Suma de distorsiones (distancias entre los puntos) que se necesitan para trasladar el triángulo medido congruente con el triángulo ideal.

Nota

Calidad de la medición

Para poder asignar las coordenadas medidas a las ideales a través de una rotación/translación combinada, el triángulo abierto desde los puntos de medida tiene que ser congruente con el triángulo ideal. Esto se consigue con un algoritmo de compensación que minimiza la suma de los cuadrados de las desviaciones que trasladan el triángulo medido al triángulo ideal.

La distorsión efectiva necesaria de los puntos de medida puede considerarse un indicador de la calidad de la medición; por ello la entrega también MEAFRAME en calidad de variable adicional.

Nota

El frame generado por MEAFRAME se puede transformar, con la función ADDFRAME, en otro frame en la cadena de frames (ver ejemplo "Concatenación con ADDFRAME").

Ejemplos

Ejemplo 1:

Programa de pieza 1:

Código de programa

```
...
DEF FRAME CORR_FRAME
```

Definir puntos de medida:

Código de programa	**Comentarios**
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)	; Para prueba.
DEF REAL FIT_QUALITY=0	
DEF REAL ROT_FRAME_LIMIT=5	; Permite un giro máx. de 5 grados de la posición de la pieza.
DEF REAL FIT_QUALITY_LIMIT=3	; Permite un decalaje máx. de 3 mm entre el triángulo ideal y el medido.
DEF REAL SHOW_MCS_POS1[3]	
DEF REAL SHOW_MCS_POS2[3]	
DEF REAL SHOW_MCS_POS3[3]	

Código de programa	**Comentarios**
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	; Limitación del 1.er ángulo RPY.
SETAL(65020)	
GOTOF NO_FRAME	
ENDIF	
N260 IF CORR_FRAME[Y,RT] > ROT_FRAME_LIMIT	; Limitación del 2.° ángulo RPY.
SETAL(65021)	
GOTOF NO_FRAME	
ENDIF	
N270 IF CORR_FRAME[Z,RT] > ROT_FRAME_LIMIT	; Limitación del 3.er ángulo RPY.
SETAL(65022)	
GOTOF NO_FRAME	
ENDIF	
N300 $P_IFRAME=CORR_FRAME	; Activar frame palpado con un frame definible.

Código de programa	**Comentarios**
	; Comprobar el frame posicionando los ejes geométricos en los puntos ideales.
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	; Desactivar frame ajustable porque está ajustado por defecto con frame cero (no se ha introducido ningún valor).
No_FRAME	; Desactivar frame ajustable porque está ajustado por defecto con frame cero (no se ha introducido ningún valor).
M0	
M30	

Ejemplo 2: concatenación de frames

Concatenación de MEAFRAME para correcciones

La función MEAFRAME suministra un frame de corrección. Si este frame de corrección se concatena con el frame ajustable $P_UIFR[1] que estaba activo en el momento de la llamada a la función (p. ej. G54), se obtiene un frame ajustable para conversiones adicionales para el desplazamiento o el mecanizado.

Concatenación con ADDFRAME

Si este frame de corrección tiene que actuar en la cadena de frames en otro punto, o si antes del frame ajustable existen otros frames activos, se puede utilizar la función ADDFRAME para la concatenación en uno de los frames básicos de canal o en un frame de sistema.

En los frames no deben estar activos:

- Simetría especular con MIRROR
- Escala con SCALE

Los parámetros de entrada para valores teóricos y reales son las coordenadas de pieza. En el sistema básico del control estas coordenadas son siempre métricas o en pulgadas (G71/G70) y deben indicarse como medida relativa al radio (DIAMOF).

Encontrará **más información** sobre ADDFRAME en el Manual de funciones Funciones básicas.

3.8.8 Frames globales

Los frames globales solo existen una vez por cada control para todos los canales. Los frames globales se pueden escribir y leer desde todos los canales. La activación de los frames globales se realiza en el correspondiente canal.

Con frames globales se pueden añadir decalajes, escalas y simetría a **ejes de canal y de máquina**.

Relaciones geométricas y cadenas de frames

Para los frames globales no existe ninguna relación geométrica entre los ejes. Por ello no se pueden ejecutar rotaciones ni programación de identificadores de ejes geométricos.

Las rotaciones no se pueden aplicar a frames globales. La programación de una rotación se rechaza con la alarma 18310 "Canal %1 Secuencia %2 Frame: Rotación no permitida".

La concatenación de frames globales y frames específicos del canal es posible. El frame resultante contiene todas las porciones de frame, incluyendo las rotaciones para todos los ejes. La asignación de un frame con porciones de rotación a un frame global se rechaza con la alarma "Frame: Rotación no permitida".

Frames globales

Frames básicos globales $P_NCBFR[n]

Se pueden configurar hasta 8 frames básicos globales:

Al mismo tiempo pueden existir frames básicos específicos del canal.

Los frames básicos globales se pueden escribir y leer desde todos los canales de un control. Al escribir en frames globales el usuario debe preocuparse por la coordinación de canales. Esto se puede realizar
p. ej. con metas de espera (WAITMC).

Nota

Fabricante de la máquina

La cantidad de frames básicos globales se configura mediante datos de máquina.

Información adicional: Manual de funciones Funciones básicas

Frames ajustables $P_UIFR[n]

Todos los frames ajustables G500, G54...G599 se pueden configurar como frames globales o específicos de canal.

Nota

Fabricante de la máquina

Se pueden reconfigurar todos los frames ajustables como frames globales mediante el dato de máquina MD18601 $MN_MM_NUM_GLOBAL_USER_FRAMES.

Como descriptores del eje para los comandos de programa frame se pueden utilizar descriptores de canal y descriptores de eje de máquina. La programación de descriptores de eje geométrico se rechaza con una alarma.

3.8.8.1 Frames específicos de canal ($P_CHBFR, $P_UBFR)

Por intervención del operador y desde el PLC se pueden escribir y leer los frames ajustables o frames básicos a través del programa de pieza y a través del BTSS.

El decalaje fino es posible también para los frames globales. La supresión de frames globales se realiza del mismo modo que los frames específicos de canal mediante G53, G153, SUPA y G500.

Fabricante de la máquina

Con el dato de máquina MD28081 $MC_MM_NUM_BASE_FRAMES se puede configurar el número de frames básicos en el canal. La configuración básica está definida de modo que existe al menos un frame básico por canal. Como máximo son posibles 8 frames básicos por cada canal. Además de los 8 frames básicos en el canal puede haber 8 frames básicos globales NCU.

Frames específicos del canal

$P_CHBFR[n]

Mediante la variable de sistema $P_CHBFR[n] se pueden leer y escribir los frames básicos. Al escribir un frame básico no se activa el frame básico total concatenado, sino que la activación se realiza solo con la ejecución de una instrucción G500, G54 ... G599. La variable sirve en primer lugar como memoria para los procesos de escritura en el frame básico de HMI o PLC. Las variables de tipo frame se aseguran con la protección de datos.

Primer frame básico en el canal

Una escritura en la variable predefinida $P_UBFR no activa al mismo tiempo el frame básico con el índice de matriz 0, sino que la activación solo se produce al ejecutarse una instrucción G500, G54 ... G599. La variable también puede ser escrita y leída directamente desde el programa.

$P_UBFR

$P_UBFR es idéntico a $P_CHBFR[0]. Como estándar existe siempre un frame básico en el canal, de modo que la variable de sistema es compatible con estados más antiguos. Si no

existe ninguna variable frame predefinida específica del canal, al escribir o leer se emite la alarma "Frame: Instrucción no permitida".

3.8.8.2 Frames activos en el canal

Frames activos en el canal son introducidos desde el programa de pieza a través de las variables de sistema correspondientes a dichos frames. Esto incluye los frames de sistema. A través de estas variables de sistema se puede leer y escribir en el programa de pieza el frame de sistema actual.

Frames activos en el canal actuales

Vista general

Frames de sistema actuales	para:
$P_PARTFRAME	TCARR y PAROT
$P_SETFRAME	Poner valor real y Aproximación con contacto
$P_EXTFRAME	Decalaje de origen externo
$P_NCBFRAME[n]	Frames básicos globales actuales
$P_CHBFRAME[n]	Frames básicos de canal actuales
$P_BFRAME	1.er frame básico actual en el canal
$P_ACTBFRAME	Frame básico total
$P_CHBFRMASK y $P_NCBFRMASK	Frame básico total
$P_IFRAME	Frame ajustable actual
Frames de sistema actuales	para:
$P_TOOLFRAME	TOROT y TOFRAME
$P_WPFRAME	Puntos de referencia de pieza
$P_TRAFRAME	Transformadas
$P_PFRAME	Frame programable actual
Frame de sistema actual	para:
$P_CYCFRAME	Ciclos
P_ACTFRAME	Frame total actual
Concatenación de FRAMES	El frame actual se compone del frame básico total

$P_NCBFRAME[n] Frames básicos globales actuales

Mediante la variable de sistema $P_NCBFRAME[n] se pueden leer y escribir los elementos de matriz de frames básicos globales actuales. El frame básico global resultante se incorpora a través del proceso de escritura en el canal.

El frame modificado solamente se activa en el canal, en el que el frame ha sido programado. Para modificar el frame para todos los canales de un control, se deben escribir al mismo tiempo $P_NCBFR[n] y $P_NCBFRAME[n]. Entonces los demás canales deben activar todavía el

frame, p. ej., con G54. Al escribir una variable frame predefinida se calcula de nuevo la variable frame predefinida total.

$P_CHBFRAME[n] Frames básicos de canal actuales

Mediante la variable de sistema $P_CHBFRAME[n] se pueden leer y escribir los elementos de matriz de frames básicos de canal actuales. La variable frame predefinida total resultante se tiene en cuenta en el canal con el proceso de escritura. Al escribir una variable frame predefinida se calcula de nuevo la variable frame predefinida total.

$P_BFRAME 1.er frame básico actual en el canal

Mediante la variable frame predefinida $P_BFRAME se puede leer y escribir en el programa de pieza el frame básico actual con el índice de matriz 0 que es válido en el canal. La variable frame predefinida escrita se tiene en cuenta a efecto inmediato.

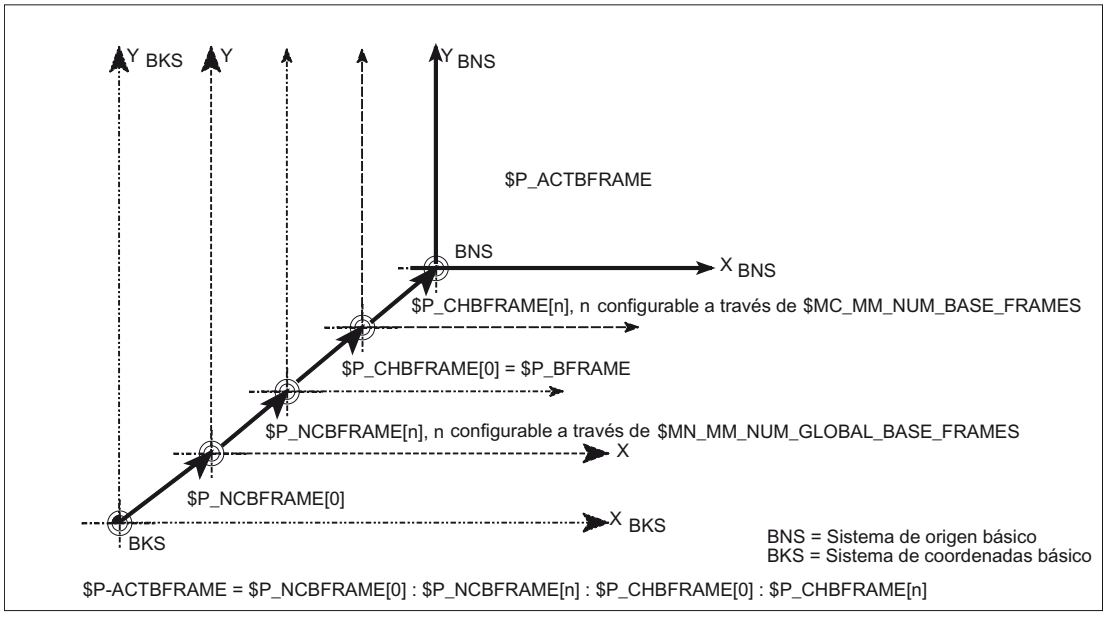
$P_BFRAME es idéntico a $P_CHBFRAME[0]. Como estándar, la variable de sistema tiene siempre un valor válido. Si no existe ninguna variable frame predefinida específica del canal, al escribir o leer se emite la alarma "Frame: Instrucción no permitida".

$P_ACTBFRAME Frame básico total

La variable $P_ACTFRAME determina el frame básico total concatenado. La variable sólo puede ser leída.

$P_ACTFRAME equivale a:

$P_NCBFRAME[0] : ... : $P_NCBFRAME[n] : $P_CHBFRAME[0] : ... : $P_CHBFRAME[n].



$P_CHBFRMASK y $P_NCBFRMASK Frame básico total

Mediante las variables de sistema $P_CHBFRMASK y $P_NCBFRMASK, el usuario puede elegir qué frame básico quiere incluir en el cálculo del frame básico "total". Las variables sólo pueden ser programadas en el programa y leídas vía la BTSS. El valor de las variables se

interpreta como máscara de bits e indica qué elemento de matriz de frame básico de $P_ACTFRAME se tiene en cuenta para el cálculo.

Con $P_CHBFRMASK se puede predefinir los frames básicos específicos del canal, y con $P_NCBFRMASK, los frames básicos globales, que se tendrán en cuenta para el cálculo.

Con la programación de las variables se calculan de nuevo el frame básico total y el frame total. Tras el Reset y con el ajuste base, el valor de $P_CHBFRMASK y $P_NCBFRMASK es el siguiente:

$P_CHBFRMASK = $MC_CHBFRAME_RESET_MASK

$P_NCBFRMASK = $MC_CHBFRAME_RESET_MASK

Ejemplo:

```
$P_NCBFRMASK = 'H81' ;$P_NCBFRAME[0] : $P_NCBFRAME[7]
$P_CHBFRMASK = 'H11' ;$P_CHBFRAME[0] : $P_CHBFRAME[4]
```

$P_IFRAME Frame ajustable actual

Mediante la variable frame predefinida $P_IFRAME se puede leer y escribir en el programa de pieza el frame ajustable actual válido en el canal. La variable frame ajustable escrita se tiene en cuenta a efecto inmediato.

En caso de frames globales ajustables, el frame modificado solo actúa en el canal en el que se ha programado el frame. Para modificar el frame para todos los canales de un control, se deben escribir al mismo tiempo $P_UIFR[n] y $P_IFRAME. Entonces los demás canales deben activar todavía el correspondiente frame, p. ej., con G54.

$P_PFRAME Frame programable actual

$P_PFRAME es el frame programable que resulta de la programación de TRANS/ATRANS, G58/G59, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR o de la asignación de CTRANS, CROT, CMIRROR, CSCALE al FRAME programable.

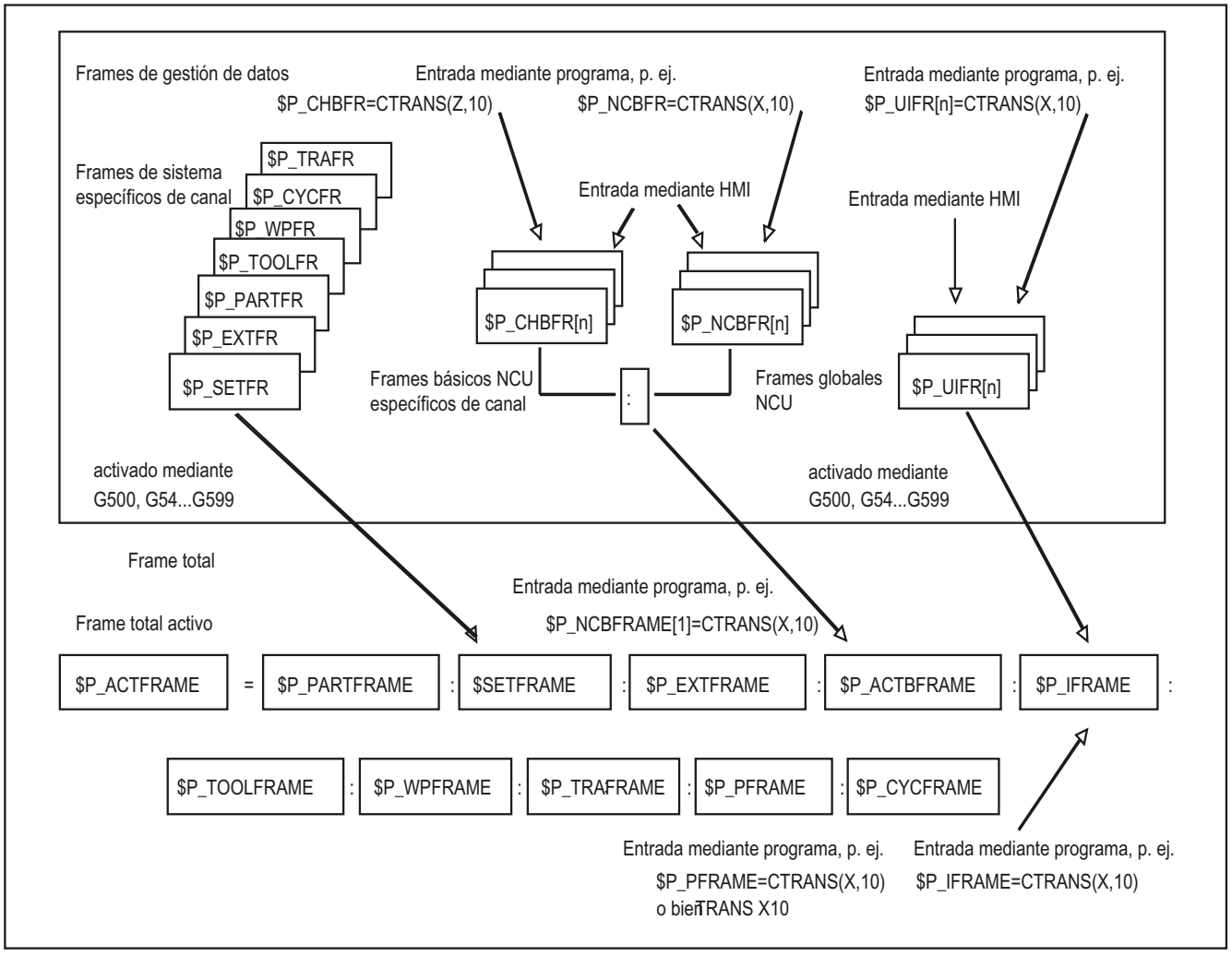
Variable frame programable actual que establece la relación entre el sistema de origen ajustable (ENS) y el sistema de coordenadas de pieza (WKS).

P_ACTFRAME Frame total actual

El frame total actual resultante $P_ACTFRAME se obtiene entonces como concatenación de todos los frames básicos, del frame ajustable actual y del frame programable. La variable frame actual se actualiza siempre, cuando se modifica una proporción de variable frame.

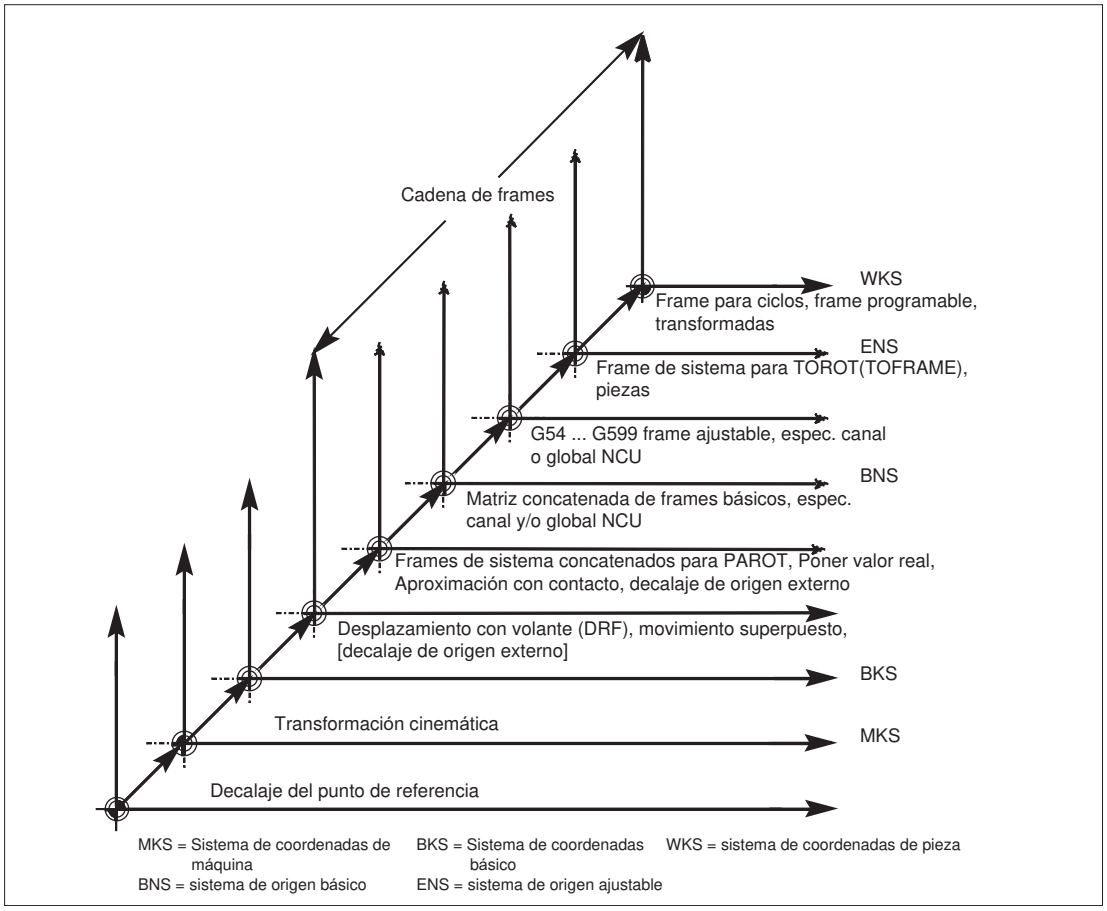
`$P_ACTFRAME` equivale a:

```
$P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME : $P_ACTBFRAME : $P_IFRAME :
$P_TOOLFRAME : $P_WPFRAME : $P_TRAFRAME : $P_PFRAME : $P_CYCFRAME
```


Frames de gestión de datos
Entrada mediante programa, p. ej.
$P_CHBFR=CTRANS(Z,10)
$P_NCBFR=CTRANS(X,10)
Entrada mediante programa, p. ej.
$P_UIFR[n]=CTRANS(X,10)
Frames de sistema específicos de canal
$P_TRAFR
$P_CYCFR
$P_WPFR
$P_TOOLFR
$P_PARTFR
$P_EXTFR
$P_SETFR
Entrada mediante HMI
Entrada mediante HMI
$P_CHBFR[n]
$P_NCBFR[n]
$P_UIFR[n]
Frames básicos NCU específicos de canal
:
Frames globales NCU
activado mediante G500, G54...G599
activado mediante G500, G54...G599
Frame total
Entrada mediante programa, p. ej.
Frame total activo
$P_NCBFRAME[1]=CTRANS(X,10)
$P_ACTFRAME = $P_PARTFRAME : $SETFRAME : $P_EXTFRAME : $P_ACTBFRAME : $P_IFRAME :
$P_TOOLFRAME : $P_WPFRAME : $P_TRAFRAME : $P_PFRAME : $P_CYCFRAME
Entrada mediante programa, p. ej.
$P_PFRAME=CTRANS(X,10)
o bien CTRANS X10
Entrada mediante programa, p. ej.
$P_IFRAME=CTRANS(X,10)

Concatenación de frames

El frame actual se compone del frame básico total, del frame ajustable, del frame de sistema y del frame programable según el frame total actual arriba indicado.



3.9 Transformadas

3.9.1 Programación general de los tipos de transformada

Función general

Para adaptar el control a distintas cinemáticas de máquina existe la selección de programar tipos de transformada con parámetros apropiados. A través de estos parámetros se pueden acordar, para la transformada seleccionada, tanto la orientación de la herramienta en el espacio como también los movimientos de orientación de los ejes giratorios.

En las transformadas de tres, cuatro y cinco ejes, los datos de posición programados se refieren siempre a la punta de la herramienta que sigue en posición ortogonal a la superficie de mecanizado situada en el espacio. Las coordenadas cartesianas se convierten del sistema de coordenadas básico al sistema de coordenadas de máquina y se refieren a los ejes geométricos. Éstos describen el punto de trabajo. Ejes giratorios virtuales describen las orientaciones de la herramienta en el espacio y se programan con TRAORI.

En la transformada cinemática se puede programar posiciones en el sistema de coordenadas cartesiano. El control transforma los movimientos de desplazamiento programados con TRANSMIT, TRACYL y TRAANG del sistema de coordenadas cartesiano a los movimientos de desplazamiento de los ejes de máquina reales.

Programación

Transformadas de tres, cuatro y cinco ejes TRAORI

La transformada de orientación acordada se activa con el comando TRAORI y los tres parámetros posibles para número de transformada, vector de orientación y correcciones de eje giratorio.

```
TRAORI(número de transformada, vector de orientación y correcciones
de eje giratorio)
```

Transformadas cinemáticas

A las transformadas cinemáticas pertenecen las transformadas acordadas
`TRANSMIT(número de transformada)`

`TRACYL(diámetro de trabajo, número de transformada)`

`TRAANG(ángulo del eje inclinado, número de transformada)`

Desactivar transformada activa

Con `TRAFOOF` se puede desactivar la transformada actualmente activa.

Transformada de orientación

Transformadas de tres, cuatro y cinco ejes TRAORI

Para el mecanizado óptimo de superficies conformadas en el espacio en el área de trabajo de la máquina, las máquinas herramienta necesitan ejes adicionales a los tres ejes lineales X, Y

y Z. Los ejes adicionales describen la orientación en el espacio y se denominarán a continuación como ejes de orientación. Están disponibles como ejes giratorios en cuatro tipos de maquinas con distintas cinemáticas.

1. Cabezal orientable de dos ejes, p. ej., cabezal tipo cardán con un eje giratorio paralelo a un eje lineal con mesa de herramienta fija.
2. Mesa giratoria de dos ejes, p. ej., cabezal orientable fijo con mesa de herramientas giratoria alrededor de dos ejes.
3. Cabezal orientable de un eje y mesa giratoria de un eje, p. ej., un cabezal orientable con herramienta girada con mesa de herramienta giratoria alrededor de un eje.
4. Cabezal orientable de dos ejes y mesa giratoria de un eje, p. ej., con mesa de herramienta giratoria alrededor de un eje y un cabezal orientable con herramienta que alrededor de sí misma.

Las **transformadas de 3 y 4 ejes** son formas especiales de la transformada de 5 ejes y se programan de forma análoga a las transformadas de 5 ejes.

La "**transformada genérica de 3/4/5/6 ejes**" cubre con su volumen de funciones para ejes giratorios dispuestos en ángulo recto las transformadas para el cabezal portafresas cardánico y se puede activar, como cualquier otra transformada de orientación, también para estos cuatro tipos de máquina con TRAORI. En la transformada genérica de 5/6 ejes, la orientación de herramienta tiene un tercer grado de libertad en el cual la herramienta se puede girar alrededor de su propio eje, en posición libre en el espacio frente a la dirección de la herramienta.

Información adicional

Manual de funciones Transformadas; transformadas multieje

Estado inicial de la orientación de herramienta independiente de la cinemática

ORIRESET

Si hay una transformada de orientación activa con TRAORI, pueden indicarse con ORIRESET los estados iniciales de hasta 3 ejes de orientación con parámetros opcionales A, B y C. La asignación del orden de los parámetros programados a los ejes giratorios se realiza según el orden de los ejes de orientación establecido mediante la transformada. La programación de ORIRESET(A, B, C) produce un desplazamiento lineal y síncrono de los ejes de orientación desde su posición instantánea hasta la posición de estado inicial indicada.

Transformadas cinemáticas

TRANSMIT y TRACYL

Para fresados en tornos se puede programar, para la transformada acordada,

1. el mecanizado mediante herramienta motorizada en superficies de refrentado con TRANSMIT o
2. el mecanizado de ranuras con cualquier orientación en piezas cilíndricas con TRACYL.

TRAANG

Si el eje de penetración tiene que poder pasar también en posición inclinada (p. ej., para la tecnología Rectificación) se puede programar con TRAANG un ángulo parametrizable para la transformada acordada.

Desplazamiento PTP cartesiano

La transformada cinemática comprende también el "Desplazamiento PTP cartesiano" en el cual se pueden programar hasta 8 distintas posiciones de articulación STAT=. Las posiciones se programan en el sistema de coordenadas cartesiano, realizándose el desplazamiento de la máquina en coordenadas de máquina.

Información adicional
Manual de funciones Transformadas; transformada cinemática

Transformadas concatenadas

Se pueden conectar dos transformadas en serie. En la segunda transformada concatenada de esta manera se adoptan las partes de movimiento para los ejes de la primera transformada.

Para la primera transformada son posibles:

- Transformada de orientación TRAORI
- Transformada polar TRANSMIT
- Transformada cilíndrica TRACYL
- Transformada eje oblicuo TRAANG

La segunda transformada tiene que ser Eje oblicuo TRAANG

3.9.1.1 Movimientos de orientación en las transformadas

Movimientos de desplazamiento y movimientos de orientación

Los movimientos de desplazamiento de las orientaciones programables dependen principalmente del tipo de máquina. En la transformada de tres, cuatro y cinco ejes con TRAORI, los ejes rotatorios o los ejes lineales basculables describen los movimientos de orientación de la herramienta.

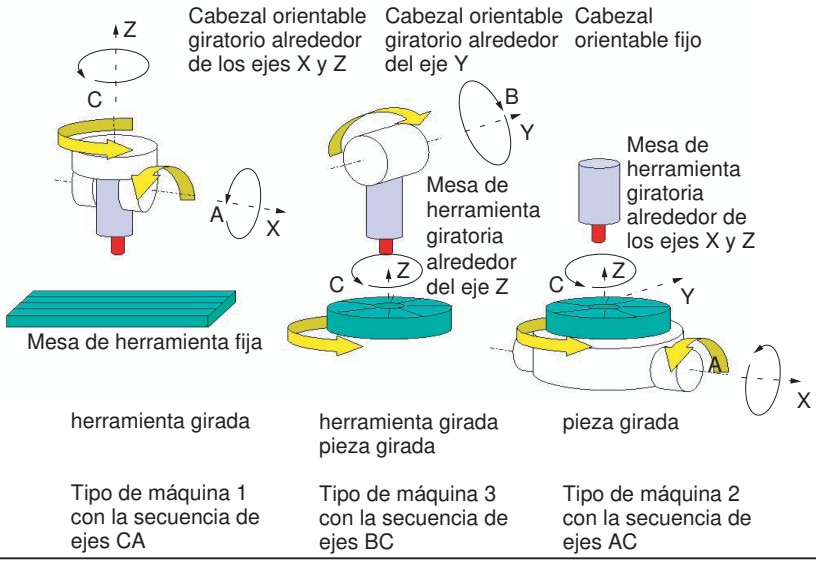
Modificaciones de las posiciones de los ejes giratorios que participan en la transformada de orientación producen movimientos de compensación de los demás ejes de máquina. La posición de la punta de la herramienta permanece inalterada.

Según la aplicación, los movimientos de orientación de la herramienta se pueden programar a través de los descriptores de eje giratorio A…, B…, C… de los ejes virtuales, indicando ángulos de Euler o RPY o vectores de dirección o normales a la superficie, vectores normalizados para el eje giratorio de un cono o para la orientación intermedia en una superficie de cono.

En la transformada cinemática con TRANSMIT, TRACYL y TRAANG, el control transforma los movimientos de desplazamiento programados del sistema de coordenadas cartesiano a los movimientos de desplazamiento de los ejes de máquina reales.

Cinemática de máquina en transformada de tres, cuatro y cinco ejes TRAORI

La herramienta o la mesa de herramienta pueden ser giratorias con hasta dos ejes giratorios. También es posible una combinación de un cabezal orientable de un eje y una mesa giratoria.



Tipo de máquina	Programación de la orientación
Transformada de tres ejes tipos de máquina 1 y 2	Programación de la orientación de herramienta únicamente en el plano perpendicular al eje rotatorio. Existen **dos** ejes de translación (ejes lineales) y **un** eje rotatorio (eje giratorio)
Transformada de cuatro ejes tipos de máquina 1 y 2	Programación de la orientación de herramienta únicamente en el plano perpendicular al eje rotatorio. Existen **tres** ejes de translación (ejes lineales) y **un** eje rotatorio (eje giratorio)
Transformada de cinco ejes tipos de máquina 3 Cabezal orientable de un eje y mesa giratoria de un eje	Programación de la transformada de orientación Cinemática con **tres** ejes lineales y **dos** ejes giratorios ortogonales. Los ejes giratorios son paralelos a dos de los tres ejes lineales. El primer eje giratorio es movido por dos ejes lineales cartesianos. Gira el tercer eje lineal con la herramienta. El segundo eje giratorio gira la pieza.

Transformadas genéricas de 5/6 ejes

Tipo de máquina	Programación de la transformada de orientación
Transformada genérica de cinco/seis ejes tipos de máquina 4 Cabezal orientable de dos ejes con herramienta giratoria sobre sí misma y mesa giratoria de un eje	Programación de la transformada de orientación Cinemática con **tres** ejes lineales y **tres** ejes giratorios ortogonales. Los ejes giratorios son paralelos a dos de los tres ejes lineales. El primer eje giratorio es movido por dos ejes lineales cartesianos. Gira el tercer eje lineal con la herramienta. El segundo eje giratorio gira la pieza. La orientación básica de la herramienta puede programarse mediante un giro adicional sobre sí misma con el ángulo de giro THETA.

Al llamar a la "transformada genérica de tres, cuatro y cinco/seis ejes" se puede consignar adicionalmente la orientación básica de la herramienta. Ya no se aplican las limitaciones con respecto a las direcciones de los ejes giratorios. Si los ejes giratorios no son exactamente perpendiculares entre sí o los ejes giratorios existentes no son exactamente paralelos a los ejes lineales, la "transformada genérica de cinco/seis ejes" puede suministrar mejores resultados para la orientación de herramienta.

Transformadas cinemáticas TRANSMIT, TRACYL y TRAANG

Para fresados en tornos o en un eje que se puede pasar en posición inclinada en la rectificación, se aplican, en función de la transformada, las siguientes disposiciones de ejes en el caso estándar:

TRANSMIT	Activación de la transformada polar
mecanizado mediante herramienta motorizada en superficies de refrentado	un eje giratorio un eje de penetración perpendicular al eje giratorio un eje longitudinal paralelo al eje giratorio

TRACYL	Activación de la transformada de la envolvente del cilindro
Mecanizado de ranuras con cualquier orientación en piezas cilíndricas	un eje giratorio un eje de penetración perpendicular al eje giratorio un eje longitudinal paralelo al eje giratorio

TRAANG	Activación de la transformada Eje oblicuo
Mecanizado con eje de penetración inclinado	un eje giratorio un eje de penetración con un ángulo parametrizable un eje longitudinal paralelo al eje giratorio

Desplazamiento PTP cartesiano

El movimiento de la máquina se realiza en coordenadas de máquina y se programa con:

TRAORI	Activación de la transformada
PTP, desplazamiento punto a punto	Desplazamiento a la posición en el sistema de coordenadas cartesiano (MKS)
CP	Desplazamiento de trayectoria de los ejes cartesianos en el (BKS)
STAT	La posición de las articulaciones depende de la transformada
TU	En qué ángulo se desplazan los ejes por el trayecto más corto

Desplazamiento PTP en transformada genérica de 5/6 ejes

El movimiento de la máquina se realiza en coordenadas de máquina y la orientación de herramienta puede programarse tanto con posiciones de ejes giratorios como con los vectores de Euler o los ángulos RPY, independientes de la cinemática, o con los vectores de dirección.

Además, son posibles la interpolación de ejes giratorios, la interpolación de vectores con interpolación circular de gran radio o la interpolación del vector de orientación sobre una superficie de cono.

Ejemplo: transformada de tres a cinco ejes con un cabezal de fresado cardánico

La máquina herramienta tiene al menos 5 ejes, de ellos:

- Tres ejes de translación para movimientos en línea recta que desplazan el punto de trabajo a cualquier posición en el área de trabajo.
- Dos ejes rotatorios dispuestos en un ángulo configurable (habitualmente 45 grados) permiten a la herramienta ocupar orientaciones en el espacio que, en la disposición en 45 grados, se limitan a una semiesfera.

3.9.1.2 Resumen de la transformada de orientación TRAORI

Posibles tipos de programación en relación con TRAORI

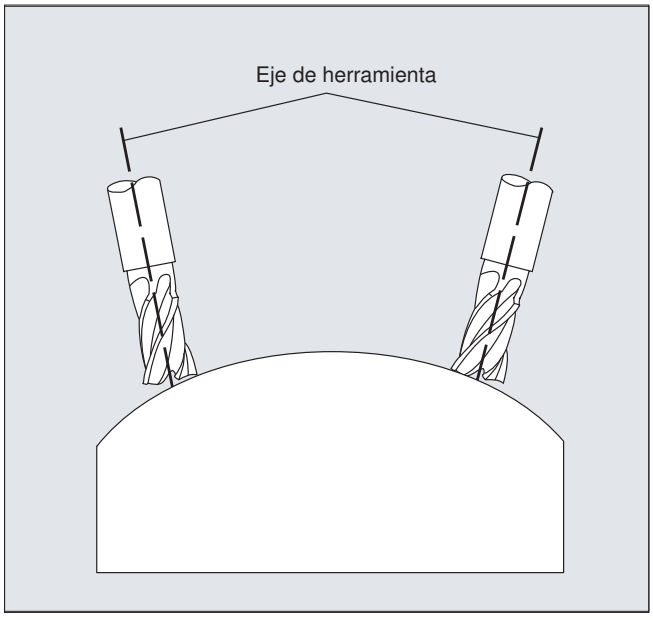
Tipo de máquina	Programación con transformada TRAORI activa
Tipos de máquina 1, 2 ó 3, cabezal orientable de dos ejes o mesa giratoria de dos ejes o una combinación de un cabezal orientable y una mesa giratoria de un eje.	La secuencia de ejes de los ejes de orientación y la dirección de orientación de la herramienta se pueden configurar **relativas a la máquina** a través de datos de máquina de forma dependiente de la cinemática de la máquina o **relativas a la pieza** con orientación programable de forma independiente de la cinemática de la máquina El sentido de giro de los ejes de orientación en el sistema de referencia se programa con: - ORIMKS sistema de referencia = sistema de coordenadas de máquina - ORIWKS sistema de referencia = sistema de coordenadas de pieza El ajuste básico es ORIWKS. Programación de los ejes de orientación con: A, B, C indicación directa de las posiciones de ejes A2, B2, C2 programación de ángulos de ejes virtuales mediante - ORIEULER a través de ángulos de Euler (estándar) - ORIRPY a través de ángulos RPY - ORIVIRT1 a través de ejes de orientación virtuales de 1.ª definición - ORIVIRT2 a través de ejes de orientación virtuales de 2.ª definición con distinción del tipo de interpolación: **Interpolación lineal** - ORIAXES de ejes de orientación o ejes de máquina **Interpolación circular de gran radio**(interpolación del vector de orientación) - ORIVECT de ejes de orientación Programación de los ejes de orientación por indicación A3, B3, C3 de las componentes vectoriales (normal de dirección/ superficie) Programación de la orientación de herramienta resultante A4, B4, C4 del vector normal de superficie al inicio de la secuencia A5, B5, C5 del vector normal de superficie al final de la secuencia LEAD ángulo de avance para la orientación de la herramienta TILT ángulo lateral para la orientación de la herramienta

Tipo de máquina	Programación con transformada TRAORI activa
	Interpolación del vector de orientación en una superficie de cono Cambios de orientación en una superficie de cono situada **libremente en el espacio** por interpolación: - ORIPLANE en el plano (interpolación circular de gran radio) - ORICONCW en una superficie de cono en sentido horario - ORICONCCW en una superficie de cono en sentido antihorario A6, B6, C6 vector de dirección (eje de giro de cono) -OICONIO Interpolación en una superficie de cono con: A7, B7, C7 vectores intermedios (orientación inicial y final) o - ORICONTO en una superficie de cono, transición tangencial Cambios de orientación relativos**a una trayectoria** con - ORICURVE especificación del desplazamiento de dos puntos de contacto a través de PO[XH]=(xe, x2, x3, x4, x5) polinomios de orientación de hasta 5º grado PO[YH]=(ye, y2, y3, y4, y5) polinomios de orientación de hasta 5º grado PO[ZH]=(ze, z2, z3, z4, z5) polinomios de orientación de hasta 5º grado - ORIPATHS Alisado del desarrollo de la orientación con A8, B8, C8 fase de reorientación de la herramienta corresponde a: dirección y longitud del trayecto de la herramienta en el movimiento de levantamiento
Tipos de máquina 1 y 3 Otros tipos de máquina con giro adicional de la herramienta sobre sí misma requieren un tercer eje giratorio Transformada de orientación, como p. ej. transformada genérica de 6 ejes. Giros del vector de orientación.	Programación **de los giros** de la orientación de herramienta con LEAD ángulo de avance Ángulo relativo al vector normal a la superficie PO[PHI] Programación de un polinomio de hasta 5º grado TILT ángulo lateral giro alrededor de la trayectoria tangente (dirección Z) PO[PSI] Programación de un polinomio de hasta 5º grado THETA Ángulo de giro (giro alrededor de la dirección de la herramienta en Z) THETA= valor que se alcanza al final de la secuencia THETA=AC(...) Conmutar secuencia a secuencia a acotado absoluto THETA=IC(...) Conmutar secuencia a secuencia a acotado incremental THETA=Θ_e Interpolar ángulo programado G90/G91 PO[THT]=(..) Programación de un polinomio de hasta 5º grado Programación del vector de giro - ORIROTA Giro absoluto - ORIROTR vector de giro relativo - ORIROTT vector de giro tangencial
Orientación relativa a la trayectoria para cambios de orientación relativos a la trayectoria o giro del vector de giro tangencialmente a la trayectoria	Cambios de orientación **relativo a la trayectoria** con - ORIPATH orientación de herramienta relativa a la trayectoria - ORIPATHS adicionalmente con acodamiento en el desarrollo de la orientación Programación del vector de giro - ORIROTC vector de giro tangencial, giro frente a la trayectoria tangente

3.9.2 Transformada de tres, cuatro y cinco ejes (TRAORI)

3.9.2.1 Relaciones generales, cabezal de herramienta cardánico

Para conseguir condiciones de corte óptimas en el mecanizado de superficies curvadas en el espacio, el ángulo de ataque de la herramienta se tiene que poder modificar.

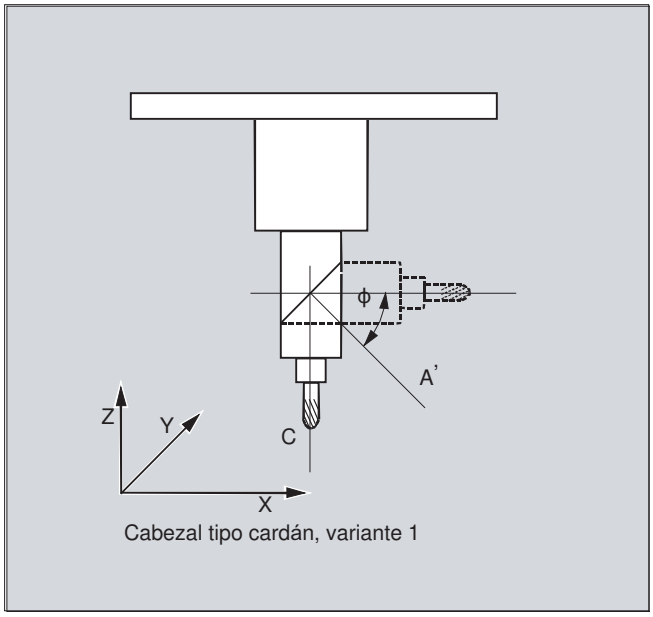


La configuración mecánica de la máquina así como sus parámetros elementales quedan almacenados en los datos de máquina de ejes.

Transformada de 5 ejes

Cabezal tipo cardan

En este caso, tres ejes lineales (X, Y, Z) y dos ejes de orientación (C, A) definen el punto de trabajo de la herramienta así como el ángulo de ataque. Uno de los dos ejes de orientación (en el ejemplo de la figura A') se define como eje inclinado, en la mayoría de los casos a 45°.

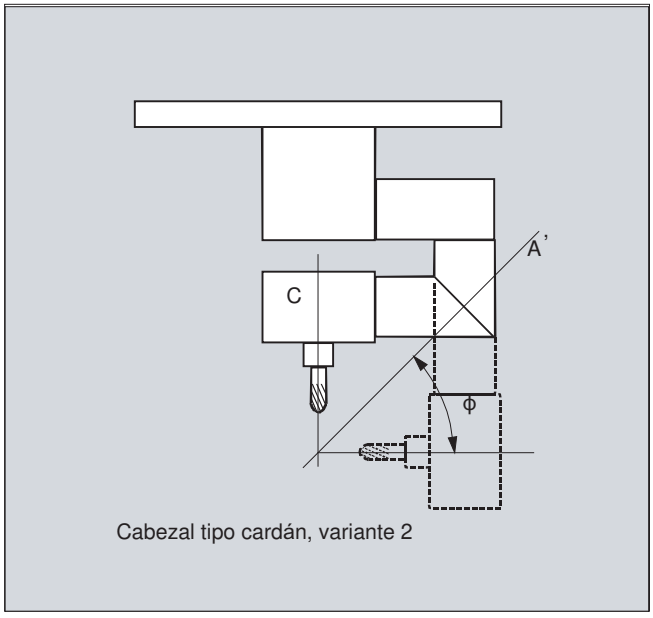


Cabezal tipo cardán, variante 1

En los ejemplos de las figuras se observa la asignación a la cinemática de la máquina para los ejes CA en el ejemplo del cabezal tipo cardán.

Fabricante de la máquina

La secuencia de movimiento para los ejes de orientación y la dirección para la orientación de la herramienta se pueden definir dependiendo de la cinemática de la máquina mediante datos de máquina.



Cabezal tipo cardán, variante 2

En este ejemplo, A' se sitúa en el ángulo φ frente al eje X

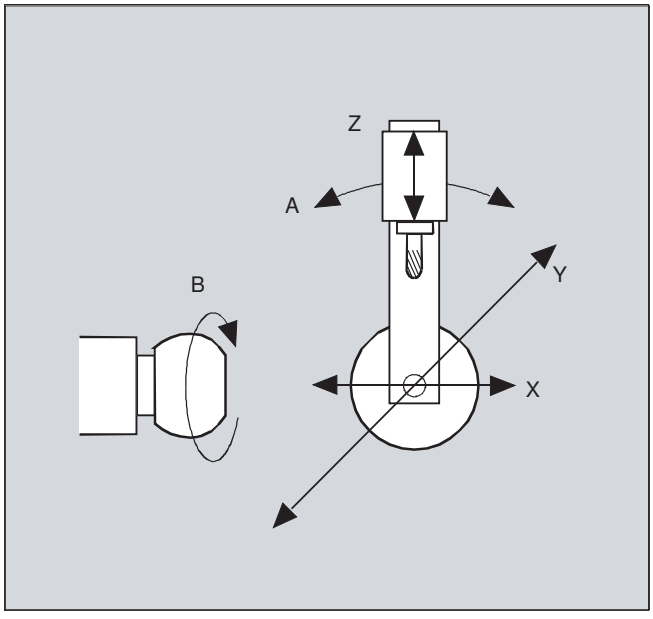
A nivel general rigen las siguientes relaciones:

A' se sitúa en el ángulo φ frente al	Eje X
B' se sitúa en el ángulo φ frente al	Eje Y
C' se sitúa en el ángulo φ frente al	Eje Z

El ángulo φ se puede parametrizar mediante datos de máquina en el rango entre 0° y +89°.

Con eje lineal basculable

En este caso se trata de una configuración en la cual se mueven tanto la herramienta como la pieza. La cinemática se compone de tres ejes lineales
(X, Y, Z) y dos ejes giratorios a 90°. El primer eje giratorio se desplaza mediante el movimiento cruzado de dos ejes lineales; la herramienta se encuentra paralela al tercer eje lineal. El segundo eje giratorio gira la herramienta. El tercer eje lineal (eje basculante) se encuentra en el plano perpendicular formado por los otros dos ejes lineales.



La secuencia de movimiento para los ejes rotatorios y la dirección para la orientación de la herramienta se pueden definir dependiendo de la cinemática de la máquina mediante datos de máquina.

Se tienen las siguientes relaciones:

Ejes:	**Secuencia de ejes:**
1.er eje giratorio	A A B B C C
2.º eje giratorio	B C A C A B
Eje lineal basculando	Z Y Z X Y X

Encontrará **más información** sobre las secuencias de ejes configurables para la dirección de orientación de la herramienta en el Manual de funciones Transformadas.

3.9.2.2 Transformada de tres, cuatro y cinco ejes (TRAORI)

El usuario puede configurar dos o tres ejes de traslación y un eje de rotación. Las transformadas requieren que el eje de rotación sea ortogonal al plano de orientación.

La orientación de la herramienta únicamente es posible en el plano perpendicular al eje de rotación. La transformada admite los tipos de máquina con herramienta móvil y pieza móvil.

La configuración y programación de las transformadas de tres y cuatro ejes son análogas a las de cinco ejes.

Información adicional: Manual de funciones Transformadas

Sintaxis

```
TRAORI(<n>)
TRAORI(<n>,<X>,<Y>,<Z>,<A>,<B>)
TRAFOOF
```

Descripción

`TRAORI`:	Activa la primera transformada de orientación declarada	
`TRAORI(<n>)`:	Activa la transformada de orientación declarada con n	
`<n>`:	Número de transformada	
	Valor:	1 ó 2
	Ejemplo: TRAORI(1) activa transformada de orientación 1	
`<X>,<Y>,<Z>`:	Componente del vector de orientación en la que apunta la herramienta.	
`<A>,<B>`:	Offset programable para ejes giratorios	
`TRAFOOF`:	Desactivar transformada	

Orientación de la herramienta

Según la dirección elegida para la orientación de la herramienta, en el programa CN debe ajustarse el plano de trabajo activo (G17, G18, G19) de tal modo que la corrección de la longitud de la herramienta actúe en esa dirección.

Nota

Tras activar la transformada, los datos de posición (X, Y, Z) se refieren siempre a la punta de la herramienta. Una modificación de las posiciones de los ejes giratorios que participan en la transformada produce los movimientos de compensación en los demás ejes de máquina, con lo cual permanece inalterada la posición de la punta de la herramienta.

La transformada de orientación siempre se ajusta desde la punta de la herramienta hasta el portaherramientas.

Offset para ejes de orientación

Al activar la transformada de orientación puede programarse directamente un offset adicional para ejes de orientación.

Es posible omitir parámetros siempre que se siga el orden correcto en la programación.

Ejemplo:

`TRAORI(, , , ,A,B)` ; si se tiene que introducir un único offset

Como alternativa a la programación directa, puede aplicarse automáticamente el offset adicional para ejes de orientación desde el decalaje de origen activo en ese momento. Esta aplicación se configura a través de datos de máquina.

Ejemplos

```
TRAORI(1,0,0,1)          ; La orientación básica de la herramienta apunta en direc-
                           ción Z
TRAORI(1,0,1,0)          ; La orientación básica de la herramienta apunta en direc-
                           ción Z
TRAORI(1,0,1,1)          ; La orientación básica de la herramienta apunta en direc-
                           ción Y/Z (corresponde a la posición -45°)
```

3.9.2.3 Variantes de la programación de la orientación y el estado inicial (ORIRESET)

Programación de la orientación de herramienta con TRAORI

Además de los ejes lineales X, Y, Z, en combinación con una transformada de orientación programable TRAORI pueden programarse posiciones de eje o ejes virtuales con ángulos o componentes vectoriales mediante los identificadores de ejes giratorios A..., B..., C.... Para ejes de orientación y de máquina existen distintos tipos de interpolación. Independientemente de los polinomios de orientación PO[ángulo] y de eje PO[eje] activos en cada momento, pueden estar programados distintos tipos de polinomio, p. ej.: G1, G2, G3, CIP o POLY.

La modificación de la orientación de la herramienta también se puede programar con vectores de orientación. La orientación final de cada secuencia se puede realizar mediante programación directa del vector o mediante la programación de las posiciones de ejes giratorios.

Variantes de la programación de la orientación en transformadas de tres a cinco ejes

Las siguientes variantes de la programación de la orientación se excluyen mutuamente.

`A, B, C`	Indicación directa de las posiciones del eje giratorio
`A2, B2, C2`	Programación de ángulos de ejes virtuales mediante ángulo euleriano o ángulo RPY
`A3, B3, C3`	Indicación de los componentes vectoriales
`LEAD, TILT`	Indicación del ángulo de avance y del ángulo lateral relativos a la trayectoria y la superficie
`A4, B4, C4` `A5, B5, C5`	Vector normal a la superficie al inicio y al final de la secuencia
`A6, B6, C6` `A7, B7, C7`	Interpolación del vector de orientación en una superficie de cono
`A8, B8, C8`	Reorientación de la herramienta, dirección y longitud del trayecto en el movimiento de levantamiento

Desplazamiento al estado inicial de la orientación de herramienta (ORIRESET)

Mediante `ORIRESET(...)` los ejes de orientación de las respectivas cinemáticas de máquina se desplazan de forma lineal y síncrona desde su posición actual hasta la posición de estado inicial programada. Si no se programa ninguna posición de estado inicial para un eje, se utiliza la posición definida en el correspondiente dato de máquina $MC_TRAFO5_ROT_AX_OFFSET_1/2.

Los frames activos de los ejes giratorios no se tienen en cuenta.

Ejemplos de cinemática de máquina CA (nombres de eje de canal C, A)

Comandos	Descripción
`ORIRESET(90, 45)`	`Eje C: 90°` `Eje A: 45°`
`ORIRESET(, 30)`	`Eje C: $MC_TRAFO5_ROT_AX_OFFSET_1/2[0]` `Eje A: 30°`
`ORIRESET( )`	`Eje C: $MC_TRAFO5_ROT_AX_OFFSET_1/2[0]` `Eje A: $MC_TRAFO5_ROT_AX_OFFSET_1/2[1]`

Ejemplos de cinemática de máquina CAC (nombres de eje de canal C, A, B)

Comandos	Descripción

`ORIRESET(90, 45, 90)`	`Eje C: 90°`
	`Eje A: 45°`
	`Eje B: 90°`

`ORIRESET( )`	`Eje C: $MC_TRAFO5_ROT_AX_OFFSET_1/2[0]`
	`Eje A: $MC_TRAFO5_ROT_AX_OFFSET_1/2[1]`
	`Eje B: $MC_TRAFO5_ROT_AX_OFFSET_1/2[2]`

Nota

El desplazamiento al estado inicial de la orientación de herramienta mediante `ORIRESET(...)` sólo puede llevarse a cabo estando activada la transformada de orientación `TRAORI(...)`.

Programación de los giros LEAD, TILT y THETA

Ángulo de avance LEAD y ángulo lateral TILT

Los giros de la orientación de herramienta se programan en la transformada de tres a cinco ejes con el ángulo de avance LEAD y el ángulo lateral TILT.

Ángulo de giro THETA

En una transformada **con un tercer eje giratorio**, el giro de la herramienta sobre sí misma con el ángulo de giro THETA se puede programar tanto para la orientación con componentes vectoriales como para la programación de los ángulos LEAD, TILT.

3.9.2.4 Programación de la orientación de herramienta (A..., B..., C..., LEAD, TILT)

Existen las siguientes posibilidades para programar la orientación de la herramienta:

1. Programación directa de las trayectorias de los ejes giratorios. El cambio en la orientación siempre se tiene en cuenta en el sistema de coordenadas base o en el sistema de coordenadas asociado a la máquina. Los ejes de orientación se desplazan como ejes síncronos.
2. Programación mediante ángulos de Euler o ángulos RPY según definiciones de ángulos mediante `A2, B2, C2`.
3. Programación del vector de dirección a través de `A3, B3, C3`. El vector de dirección apunta desde la punta de la herramienta en dirección al alojamiento de la herramienta.
4. Programación del vector normal a la superficie al inicio de la secuencia con `A4, B4, C4` y al final de la secuencia con `A5, B5, C5` (fresado frontal).
5. Programación a través del ángulo de avance `LEAD` y Ángulo lateral `TILT`
6. Programación del eje giratorio del cono como vector normalizado a través de `A6, B6, C6` o de la orientación intermedia en la superficie del cono a través de `A7, B7, C7`, ver apartado "Programación de la orientación a lo largo de una superficie de cono (ORIPLANE, ORICONxx)".
7. Programación de la reorientación, dirección y longitud del trayecto de la herramienta durante el movimiento de levantamiento mediante `A8, B8, C8`, ver apartado "Alisado del desarrollo de la orientación (ORIPATHS A8=, B8=, C8=)"

Nota

En todos los casos, la programación de la orientación de la herramienta solamente es válida cuando está activada una transformada de orientación.

Ventaja: Estos programas son portables a cualquier cinemática de máquina.

Definición de la orientación de herramienta mediante comando G

Nota

Fabricante de la máquina

Se pueden utilizar datos de máquina para seleccionar el cambio entre ángulos de Euler o ángulos RPY. Con los correspondientes ajustes de datos de máquina, es posible realizar un cambio tanto dependiente como independientemente del comando G activo del grupo 50. Se puede escoger entre las siguientes posibilidades de ajuste:

1. Si ambos datos de máquina para la definición de los ejes y del ángulo de orientación mediante comando G están ajustados a cero:
   Los ángulos programados con `A2, B2, C2` se interpretan como ángulos de Euler o ángulos RPY **dependiendo del dato de máquina** Definición de ángulo de la programación de la orientación.
2. Cuando el dato de máquina para la definición de los ejes de orientación mediante comando G está ajustado a 1, se efectúa un cambio
   dependiendo del comando G activo del grupo 50:
   Los ángulos programados con `A2, B2, C2` se interpretan según uno de los comandos G activos `ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2, ORIAXPOS` y `ORIPY2`. Los valores programados con los ejes de orientación se interpretan también como ángulos de orientación según el comando G activo del grupo 50.
3. Cuando el dato de máquina para la definición del ángulo de orientación mediante comando G está ajustado a 1 y el dato de máquina para la definición de los ejes de orientación mediante comando G está ajustado a 0, se produce un cambio
   independiente del comando G activo del grupo 50:
   Los ángulos programados con `A2, B2, C2` se interpretan según uno de los comandos G activos `ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2 ORIAXPOS` y `ORIPY2`. Los valores programados con los ejes de orientación se interpretan siempre como posiciones de ejes giratorios independientemente del comando G activo del grupo 50.

Sintaxis

Posiciones de ejes giratorios

```
G1 X<Valor> Y<Valor> Z<Valor> A<Valor> B<Valor> C<Valor>
```

Ángulo euleriano

```
G1 X<Valor> Y<Valor> Z<Valor> A2=<Valor> B2=<Valor> C2=<Valor>
```

Vector de dirección

```
G1 X<Valor> Y<Valor> Z<Valor> A3=<Valor> B3=<Valor> C3=<Valor>
```

Vector normal a la superficie al inicio de la secuencia

```
G1 X<Valor> Y<Valor> Z<Valor> A4=<Valor> B4=<Valor> C4=<Valor>
```

Vector normal a la superficie al final de la secuencia

```
G1 X<Valor> Y<Valor> Z<Valor> A5=<Valor> B5=<Valor> C5=<Valor>
```

Ángulo de avance

`LEAD=<Valor>`

Ángulo lateral

`TILT=<Valor>`

Descripción

`G1`:	Interpolación lineal
`X, Y, Z`:	Posiciones de ejes lineales
`A, B, C`:	Posiciones de ejes giratorios
`A2=, B2=, C2=`:	Programación de ángulos (ángulos eulerianos o RPY)
`A3=, B3=, C3=`:	Vector de dirección en coordenadas X, Y, Z del WKS
`A4=, B4=, C4=`:	Vector normal a la superficie al inicio de la secuencia en coordenadas X, Y, Z del WKS
`A5=, B5=, C5=`:	Vector normal a la superficie al final de la secuencia en coordenadas X, Y, Z del WKS
`LEAD=` :	Ángulo de avance [1)]
`TILT=` :	Ángulo lateral [1)]
1) Los ángulos especificados se interpretan en función del ajuste en MD21094 $MC_ORIPATH_MODE	

Información adicional

Generalmente los programas de 5 ejes se generan en un sistema CAD/CAM (no se suelen introducir directamente desde el teclado del control numérico). Por lo tanto, la siguiente descripción está destinada sobre todo a los programadores de posprocesadores.

Para la programación de la orientación están disponibles los siguientes comandos:

Comando	Descripción
`ORIEULER`:	Ángulo euleriano con secuencia de giro ZX'Z''
`ORIRPY`:	Ángulo RPY con secuencia de giro XY'Z''
`ORIRPY2`:	Ángulo RPY con secuencia de giro ZY'X''
`ORIVIRT1`:	Ejes de orientación virtuales con secuencia de giro de libre definición mediante: MD21120 $MC_ORIAX_TURN_TAB_1
`ORIVIRT2`:	Ejes de orientación virtuales con secuencia de giro de libre definición mediante: MD21130 $MC_ORIAX_TURN_TAB_2
`ORIAXPOS`:	Ejes de orientación virtuales con posiciones de eje giratorio

Nota

El fabricante de la máquina puede definir distintas variantes mediante datos de máquina. Preste atención a las indicaciones del fabricante de la máquina.

Programación con ángulos eulerianos ORIEULER, secuencia de giro Z X' Z"

Los valores programados para la orientación ORIEULER mediante A2, B2, C2 se interpretan como ángulos eulerianos (en grados).

El nuevo vector de orientación resulta de las siguientes tres rotaciones del vector de orientación original

1. con el eje giratorio A2 alrededor del eje de coordenadas Z
2. con el eje giratorio B2 alrededor del nuevo eje de coordenadas X'
3. con el eje giratorio C2 alrededor del nuevo eje de coordenadas Z"

En este caso, el valor de C2 (giro alrededor del nuevo eje Z) es irrelevante y no necesita ser programado.

Programación en ángulos RPY ORIRPY, secuencia de giro X Y' Z"

Los valores programados para la orientación ORIRPY mediante A2, B2, C2 se interpretan como ángulos RPY en grados con la secuencia de giro X Y' Z".

Nota

A diferencia de la programación con ORIEULER, en el caso de ORIRPY influyen los tres valores en el vector de orientación.

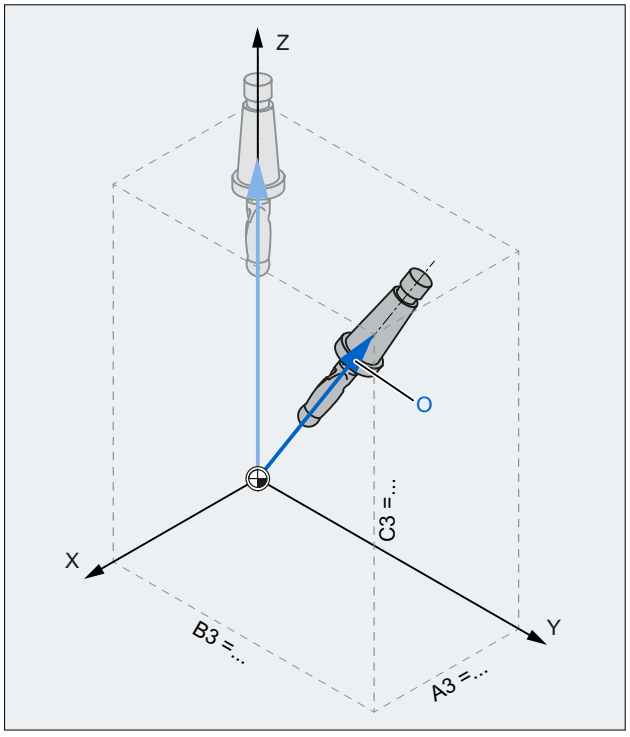
El nuevo vector de orientación resulta de las siguientes tres rotaciones del vector de orientación original

1. con el eje giratorio A2 alrededor del eje de coordenadas X
2. con el eje giratorio B2 alrededor del nuevo eje de coordenadas Y'
3. con el eje giratorio C2 alrededor del nuevo eje de coordenadas Z"

Programación del vector de dirección

Los componentes del vector de dirección se programan mediante A3, B3, C3. El vector indica la dirección de retirada de la herramienta, la longitud del vector carece por lo tanto de importancia.

Los componentes del vector no programados toman por defecto el valor 0.



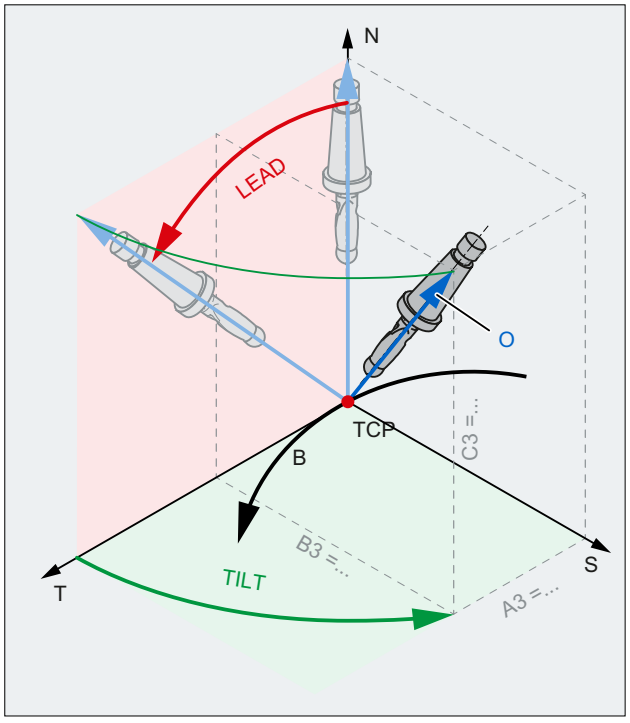
X, Y, Z	Ejes de coordenadas del WKS
A3, B3, C3	Componentes del vector de dirección
O	Vector de orientación

Figura 3-3 Programación del vector de dirección

Programación de la orientación de herramienta con LEAD y TILT

La orientación de herramienta resultante se averigua a partir de:

- Tangente a la trayectoria
- Vector normal a la superficie
  Al inicio de la secuencia `A4, B4, C4` y al final de la secuencia `A5, B5, C5`
- Ángulo de avance `LEAD`
  Ángulo en el plano definido por la tangente a la trayectoria y el vector normal a la superficie
- Ángulo lateral `TILT` al final de la secuencia
  Ángulo en el plano perpendicular a la tangente de trayectoria, relativo al vector normal a la superficie



T	Tangente a la trayectoria
S	Perpendicular a la tangente a la trayectoria
N	Normal a la superficie
B	Contorneado
TCP	Tool Center Point
O	Vector de orientación

Figura 3-4 Programación de LEAD TILT

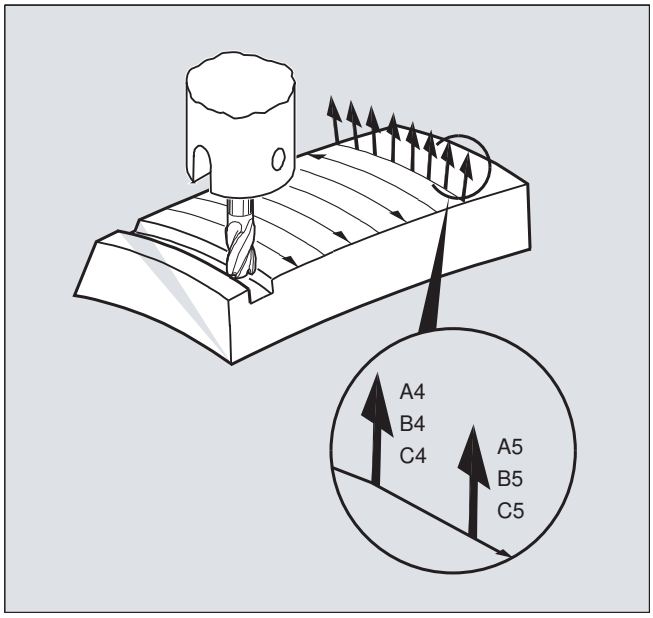
Nota

Comportamiento en esquinas interiores con corrección de herramienta en 3D

Si se acorta la secuencia en una esquina interior, la orientación de herramienta programada se adopta sin embargo al final de la secuencia.

3.9.2.5 Fresado frontal (A4, B4, C4, A5, B5, C5)

El fresado frontal sirve para mecanizar superficies con cualquier curvatura.



Para este tipo de fresado 3D se necesita una descripción línea a línea de las trayectorias 3D sobre la superficie de la pieza.

Los cálculos se realizan generalmente en sistemas CAM teniendo en cuenta la forma y las dimensiones de la herramienta. Las secuencias CN calculadas se cargan en el control numérico a través de posprocesadores.

Programación de la curvatura de la trayectoria

Descripción de las superficies

La descripción de la curvatura de la trayectoria se realiza mediante los vectores normales a la superficie con los siguientes componentes:

`A4, B4, C4` vector inicial al principio de la secuencia

`A5, B5, C5` vector final al final de la secuencia

Si en una secuencia solamente hay un vector inicial, el vector normal a la superficie permanece constante a lo largo de toda la secuencia. Si en una secuencia solamente se indica el vector final, se realiza una gran interpolación circular entre el valor final de la secuencia anterior y el valor final programado para la secuencia actual.

Si se programan los vectores inicial y final, entonces se realiza una interpolación circular mayor entre dichos vectores a lo largo de la secuencia. De esta manera es posible realizar desplazamientos suaves y uniformes a lo largo de la trayectoria generada en una secuencia.

En el ajuste básico los vectores normales a la superficie están orientados en la dirección Z, independientemente del plano activado (G17 hasta G19).

La longitud del vector carece de significado.

Los componentes no programados de los vectores tomarán el valor cero.

Cuando se encuentra activo ORIWKS (ver "Referencia de los ejes de orientación (ORIWKS, ORIMKS) (Página 694)"), los vectores normales a la superficie están referidos al frame activo y giran conjuntamente con las rotaciones asociadas al frame.

Fabricante de la máquina

El vector normal a la superficie deberá de encontrarse perpendicular a la tangente de trayectoria dentro de unos límites definidos mediante datos de máquina; en caso contrario se emite una alarma.

3.9.2.6 Referencia de los ejes de orientación (ORIWKS, ORIMKS)

En la programación de la orientación en el sistema de coordenadas de pieza a través de

- ángulos eulerianos o RPY o
- el vector de orientación

el movimiento de giro se puede ajustar mediante `ORIMKS`/`ORIWKS`.

Nota

Fabricante de la máquina

El tipo de interpolación para la orientación se define con el dato de máquina:

DM21104 $MC_ORI_IPO_WITH_G_CODE

= FALSE: La referencia son los comandos G ORIWKS y ORIMKS

= TRUE: La referencia son los comandos G del grupo 51 (ORIAXES, ORIVECT, ORIPLANE...)

Sintaxis

```
ORIMKS=...
ORIWKS=...
```

Descripción

`ORIMKS`:	Giro en el sistema de coordenadas de máquina
`ORIWKS`:	Giro en el sistema de coordenadas de pieza

Nota

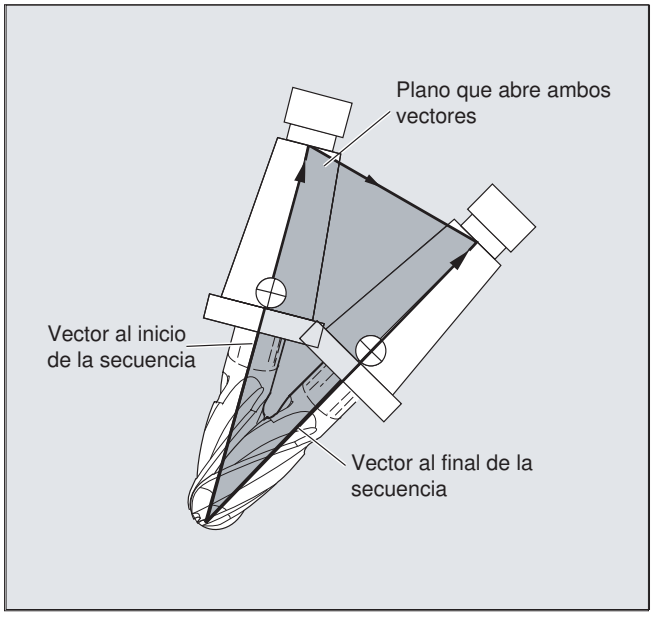
ORIWKS es el ajuste por defecto. Si a la hora de realizar una programación con cinco ejes no se ha definido para qué máquina se va a procesar el programa, se debe seleccionar ORIWKS. Qué movimientos sean realmente realizados por la máquina depende de la cinemática de ésta.

Con `ORIMKS` se pueden programar desplazamientos de máquina reales, por ejemplo, para evitar colisiones con dispositivos o similares.

Información adicional

Con ORIMKS, el movimiento ejecutado por la herramienta **depende** de la cinemática de la máquina. Los cambios de orientación de la herramienta sin desplazamiento de la punta de la misma se realizan mediante una interpolación lineal entre las posiciones de los ejes giratorios.

Con ORIWKS, el desplazamiento de la herramienta es **independiente** de la cinemática de la máquina. Un cambio en la orientación de la herramienta sin desplazamiento de la punta de la misma provoca un desplazamiento de la herramienta en el plano creado entre los vectores inicial y final.



Posiciones singulares

Nota

ORIWKS

Los desplazamientos para la orientación de la herramienta en la zona de posición singular en la máquina de cinco ejes requieren grandes desplazamientos de los ejes de máquina. (P. ej.: con un cabezal orientable con eje C como eje giratorio y eje A como eje de orientación, todas las posiciones con A=0 son singulares).

Fabricante de la máquina

Para no sobrecargar los ejes de máquina, se produce una disminución sustancial de la velocidad contorneo en la proximidad de los puntos singulares.

Con los datos de máquina

$MC_TRAFO5_NON_POLE_LIMIT

$MC_TRAFO5_POLE_LIMIT

es posible parametrizar la transformada de forma que los desplazamientos para la orientación

en las proximidades del polo se ejecuten atravesando el mismo, posibilitando así un mecanizado rápido.

Las posiciones singulares solo se tratan con el MD `$MC_TRAFO5_POLE_LIMIT`.

Información adicional: Manual de funciones Transformadas

3.9.2.7 Programación de los ejes de orientación (ORIAXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2)

La función "Ejes de orientación" describe la orientación de la herramienta en el espacio y se consigue mediante la programación de los offsets para los ejes giratorios. Un tercer grado de libertad adicional se consigue mediante el giro adicional de la herramienta alrededor de sí misma. Esta orientación de la herramienta se realiza libremente en el espacio a través de un tercer eje giratorio y exige la transformada de seis ejes. El giro propio de la herramienta en torno a su propio eje se define en función del tipo de interpolación de los vectores de giro con el ángulo de giro THETA (ver "Giros de la orientación de la herramienta (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Página 705)").

Los ejes de orientación se programan a través de los descriptores de ejes A2, B2, C2.

Sintaxis

```
N... ORIAXES/ORIVECT            ; Interpolación lineal o interpolación
                                circular de gran radio
N... G1 X Y Z A B C

N... ORIPLANE                   ; Interpolación de orientación del plano

N... ORIEULER/ORIRPY/ORIRPY2    ; Ángulo de orientación euleriano/RPY
N... G1 X Y Z A2= B2= C2=       ; Programación de ángulos de ejes virtuales

N... ORIVIRT1/ORIVIRT2          ; ejes de orientación virtuales def. 1/2
N... G1 X Y Z A3= B3= C3=       ; programación del vector de dirección
```

Nota

Para cambios de orientación a lo largo de una superficie de cono situada en el espacio, pueden programarse otros offsets de eje giratorio de los ejes de orientación (ver "Programación de la orientación a lo largo de una superficie de cono (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Página 698)").

Descripción

`ORIAXES`:	Interpolación lineal de los ejes de máquina o ejes de orientación
`ORIVECT`:	Interpolación circular de gran radio (idéntico a ORIPLANE)

ORIMKS: ORIWKS:	Giro en el sistema de coordenadas de máquina Giro en el sistema de coordenadas de pieza Descripción, ver "Referencia de los ejes de orientación (ORIWKS, ORIMKS) (Página 694)".
A= B= C=:	Programación de la posición de eje de la máquina
ORIEULER:	Programación de la orientación mediante ángulos eulerianos
ORIRPY:	Programación de la orientación de la herramienta con ángulos RPY La secuencia de giro es XYZ, donde: • A2 es el ángulo de giro en torno a X • B2 es el ángulo de giro en torno a Y • C2 es el ángulo de giro en torno a Z
ORIRPY2:	Programación de la orientación de la herramienta con ángulos RPY La secuencia de giro es ZYX, donde: • A2 es el ángulo de giro en torno a Z • B2 es el ángulo de giro en torno a Y • C2 es el ángulo de giro en torno a X
A2= B2= C2=:	Programación de ángulos de ejes virtuales
ORIVIRT1/ORIVIRT2:	Programación de la orientación de la herramienta con ejes de orientación virtuales Definición 1: definición según MD21120 $MC_ORIAX_TURN_TAB_1 Definición 2: definición según MD21130 $MC_ORIAX_TURN_TAB_2
A3= B3= C3=:	Programación del vector de dirección del eje de dirección

Información adicional

Fabricante de la máquina

Con el MD21102 $MC_ORI_DEF_WITH_G_CODE se establece el modo en que se definirán los ángulos programados A2, B2, C2:

La definición se efectúa de acuerdo con el MD21100 $MC_ORIENTATION_IS_EULER (estándar) o de acuerdo con el grupo G 50 (ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2).

Con el MD21104 $MC_ORI_IPO_WITH_G_CODE se define el tipo de interpolación activo: ORIWKS/ORIMKS u ORIAXES/ORIVECT.

Modo de operación JOG

En este modo de operación siempre se interpolan linealmente los ángulos de orientación. Durante el desplazamiento continuo e incremental mediante las teclas de desplazamiento sólo puede desplazarse un eje de orientación. Con los volantes electrónicos se pueden desplazar los ejes de orientación simultáneamente.

Para el desplazamiento manual de ejes de orientación actúa el selector de corrección del avance específico del canal, o bien el selector de corrección del rápido en caso de superposición del rápido.

Con los siguientes datos de máquina se puede predefinir una velocidad separada:

MD21160 $MC_JOG_VELO_RAPID_GEO

MD21165 $MC_JOG_VELO_GEO

MD21150 $MC_JOG_VELO_RAPID_ORI

MD21155 $MC_JOG_VELO_ORI

Nota

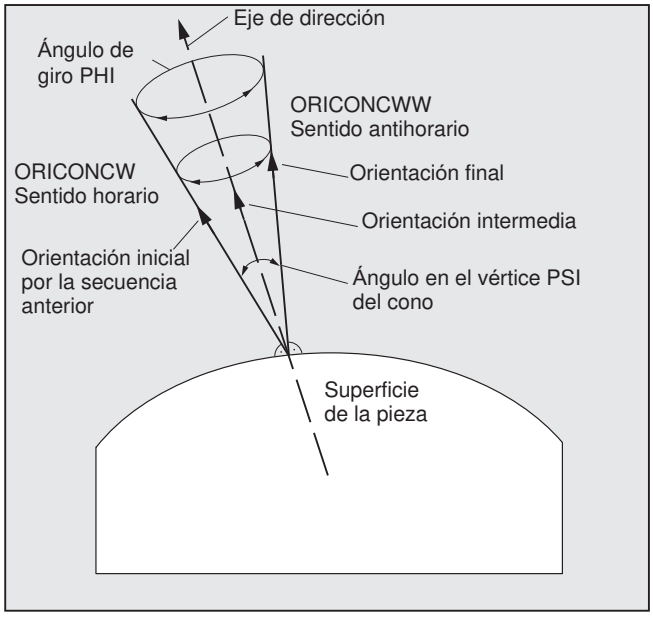
SINUMERIK 840D sl con "Paquete de transformación de manejo (Handling)"

La función "Desplazamiento manual cartesiano" permite regular por separado en el modo JOG la traslación de ejes geométricos en los sistemas de referencia MKS, WKS y TCS.

Información adicional: Manual de funciones Transformadas

3.9.2.8 Programación de la orientación a lo largo de una superficie de cono (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO)

Con la orientación ampliada es posible ejecutar cambios de orientación a lo largo de una superficie de cono situada en el espacio. La interpolación del vector de orientación en una superficie de cono se realiza con los comandos modales ORICONxx. Para la interpolación en un plano, la orientación final se puede programar con ORIPLANE. A nivel general, la orientación inicial es definida por las secuencias anteriores.



Programación

La orientación final se define indicando la programación de ángulos en ángulos Euler o RPY con A2, B2, C2 o mediante la programación de las posiciones de eje giratorio A, B, C. Para los ejes de orientación a lo largo de la superficie de cono se necesitan datos de programación adicionales:

- Eje giratorio del cono como vector con `A6, B6, C6`
- Ángulo en el vértice `PSI` con el identificador `NUT`
- Orientación intermedia en la superficie de cono con `A7, B7, C7`

Nota

Programación del vector de dirección A6, B6, C6 para el eje de giro del cono

No es absolutamente necesario programar una orientación final. Si no se indica ninguna orientación final, se interpola una superficie de cono completa con 360 grados.

Programación del ángulo en el vértice del cono con NUT=ángulo

Es absolutamente necesario indicar una orientación final.

De este modo no se puede interpolar una superficie de cono completa con 360 grados.

Programación de la orientación intermedia A7, B7, C7 en la superficie del cono

Es absolutamente necesario indicar una orientación final. El cambio de orientación y el sentido de giro se definen de forma unívoca por los tres sectores Orientación inicial, final e intermedia. Los tres vectores tienen que ser distintos. Si la orientación intermedia programada es paralela a la orientación inicial o final, se ejecuta una interpolación circular de gran radio lineal de la orientación en el plano abierto por los vectores inicial y final.

Interpolación de orientación ampliada en una superficie de cono

```
N... ORICONCW u ORICONCCW
N... A6= B6= C6= A3= B3= C3=
o bien
N... ORICONTO
N... G1 X Y Z A6= B6= C6=
o bien
N... ORICONIO
N... G1 X Y Z A7= B7= C7=
N... PO[PHI]=(a2, a3, a4, a5)
N... PO[PSI]=(b2, b3, b4, b5)
```

Interpolación en una superficie de cono con

vector de dirección en sentido horario/antihorario del cono y orientación final o

transición tangencial e indicación de la orientación final

o

indicación de la orientación final y una orientación intermedia en la superficie del cono con

polinomios para el ángulo de giro y polinomios para el ángulo en el vértice

Parámetros

`ORIPLANE`:	Interpolación en el plano (interpolación circular de gran radio)
`ORICONCW`:	Interpolación en una superficie de cono en sentido horario
`ORICONCCW`:	Interpolación en una superficie de cono en sentido antihorario

ORICONTO:	Interpolación en una superficie envolvente de cono con transición tangencial
A6= B6= C6=:	Programación del eje giratorio del cono (vector normalizado)
NUT=ángulo:	Ángulo en el vértice del cono en grados
NUT=+179:	Ángulo de desplazamiento menor o igual a 180 grados
NUT=-181:	Ángulo de desplazamiento mayor o igual a 180 grados
ORICONIO:	Interpolación en una superficie de cono
A7= B7= C7=:	Orientación intermedia (programación como vector normalizado)
PHI:	Ángulo de giro de la orientación alrededor del eje de dirección del cono
PSI:	Ángulo en el vértice del cono
Polinomios posibles PO[PHI]=(a2, a3, a4, a5) PO[PSI]=(b2, b3, b4, b5)	Además de los correspondientes ángulos también se pueden programar polinomios de máx. 5.º grado

Ejemplo: distintos cambios de orientación

Código de programa	**Comentarios**
…	
N10 G1 X0 Y0 F5000	
N20 TRAORI(1)	; Transformada de orientación activada.
N30 ORIVECT	; Interpolar orientación de herramienta como vector.
…	; Orientación de herramienta en el plano.
N40 ORIPLANE	; Seleccionar interpolación circular de gran radio.
N50 A3=0 B3=0 C3=1	
N60 A3=0 B3=1 C3=1	; Orientación en el plano Y/Z girada 45 grados; a final de secuencia se alcanza la orientación (0,1/√2,1/√2).
…	
N70 ORICONCW	; Programación de orientación en la superficie de cono:
N80 A6=0 B6=0 C6=1 A3=0 B3=0 C3=1	; El vector de orientación se interpola en una superficie de cono con la dirección (0,0,1) hasta la orientación (1/√2,0,1/√2) en sentido horario; el ángulo de giro es de 270 grados.
N90 A6=0 B6=0 C6=1	; La orientación de herramienta describe una vuelta completa en la misma superficie de cono.

Información adicional

Para describir cambios de orientación en una superficie de cono situada libremente en el espacio, se tiene que conocer el vector alrededor del cual se quiere girar la orientación de herramienta. Además se tiene que especificar la orientación inicial y final. La orientación estándar resulta de la secuencia anterior y la orientación final se tiene que programar o definir a través de otras condiciones.

Programación en el plano ORIPLANE equivale a ORIVECT

La programación de la interpolación circular de gran radio junto con polinomios angulares corresponde a la interpolación lineal y de polinomio de contornos. La orientación de herramienta se interpola un plano abierto por la orientación inicial y final. Si se programan adicionalmente polinomios, el vector de orientación también se puede volcar fuera del plano.

Programación de círculos en un plano G2/G3, CIP y CT

La orientación ampliada corresponde a la interpolación de círculos en un plano. Encontrará las posibilidades de programación de círculos correspondientes con indicación de centro o de radio como G2/G3, círculo con punto intermedio CIP y arcos tangenciales CT en el capítulo "Interpolación circular (Página 194)".

Programación de la orientación de la herramienta

Interpolación del vector de orientación en una superficie de cono ORICONxx

Para la interpolación de orientaciones en una superficie de cono se pueden seleccionar cuatro tipos de interpolación distintos del grupo G 51:

1. Interpolación en una superficie de cono en sentido horario ORICONCW con indicación de la orientación final y la dirección del cono o del ángulo en vértice. El vector de dirección se programa con los descriptores A6, B6, C6 y el ángulo en vértice del cono, con el descriptor NUT= rango de valores en un intervalo de 0 a 180 grados.
2. Interpolación en una superficie de cono en sentido antihorario ORICONCWW con indicación de la orientación final y la dirección del cono o del ángulo en vértice. El vector de dirección se programa con los descriptores A6, B6, C6 y el ángulo en vértice del cono, con el descriptor NUT= rango de valores en un intervalo de 0 a 180 grados.
3. Interpolación en una superficie de cono ORICONIO con indicación de la orientación final y una orientación intermedia que se programa con los descriptores A7, B7, C7.
4. Interpolación en una superficie de cono ORICONTO con transición tangencial e indicación de la orientación final. El vector de dirección se programa con los descriptores A6, B6, C6.

3.9.2.9 Especificación de orientación de dos puntos de contacto (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=)

Programación del cambio de orientación a través de la segunda curva espacial ORICURVE

Otra posibilidad de programar cambios de orientación consiste en programar, además de la punta de herramienta a lo largo de una curva espacial, también el desplazamiento de un segundo punto de contacto de la herramienta con ORICURVE. Esto permite definir de forma unívoca los cambios de orientación de la herramienta, como en la programación del mismo vector de herramienta.

Fabricante de la máquina

Observe las indicaciones del fabricante de la máquina acerca de descriptores de eje ajustables a través de datos de máquina para la programación de la 2.ª trayectoria de orientación de la herramienta.

Programación

En este tipo de interpolación se pueden programar, para las dos curvas espaciales, puntos con G1 o polinomios con POLY. No se admiten círculos ni evolventes. Adicionalmente se puede activar una interpolación spline con BSPLINE y la función "Resumen de secuencias spline breves".

Encontrará **más información** sobre la función "Resumen de secuencias spline breves" en el Manual de funciones Funciones básicas.

No se admiten los otros tipos de spline ASPLINE y CSPLINE, ni la activación de un compresor con COMP...

El desplazamiento de los dos puntos de contacto de la herramienta se puede especificar en la programación de los polinomios de orientación para coordenadas de máx. 5.º grado.

Interpolación de orientación ampliada con curva espacial adicional y polinomios para coordenadas

```
N... ORICURVE
N... PO[XH]=(xe, x2, x3, x4, x5)
N... PO[YH]=(ye, y2, y3, y4, y5)
N... PO[ZH]=(ze, z2, z3, z4, z5)
```

; Indicación del desplazamiento del segundo punto de contacto de la herramienta y polinomios adicionales de las coordenadas correspondientes

Parámetro

`ORICURVE`	Interpolación de la orientación con especificación del movimiento de dos puntos de contacto de la herramienta.
`XH YH ZH`	Descriptor de las coordenadas del segundo punto de contacto de la herramienta del contorno adicional como curva espacial

`PO[XH]=(xe, x2, x3, x4, x5)`
`PO[YH]=(ye, y2, y3, y4, y5)`
`PO[ZH]=(ze, z2, z3, z4, z5)`
	Polinomios posibles
	Además de los correspondientes puntos finales se pueden programar curvas espaciales adicionales con polinomios.

`xe, ye, ze`	Puntos finales de la curva espacial
`xi, yi, zi`	Coeficientes de los polinomios de máx. 5.º grado

Nota

Descriptor XH YH ZH para la programación de una 2.ª trayectoria de orientación

Los descriptores se tienen que elegir de modo que no se produzca ningún conflicto con otros descriptores de los ejes lineales

ejes X Y Z

y ejes giratorios como

A2 B2 C2 ángulo de Euler o RPY

A3 B3 C3 vectores de dirección

A4 B4 C4 ó A5 B5 C5 vectores normales a la superficie

A6 B6 C6 vectores de giro o A7 B7 C7 coordenadas de punto intermedio

u otros parámetros de interpolación.

3.9.3 Polinomios de orientación (PO[ángulo], PO[coordenada])

Independientemente de la interpolación polinómica del grupo 1 que se encuentre activa en este momento, se pueden programar dos tipos distintos de polinomios de orientación de máx. 5.º grado en una transformada de tres a cinco ejes.

1. Polinomios para **ángulo:** Ángulo de avance LEAD, ángulo lateral TILT relativo al plano tendido por la orientación inicial y final.
2. Polinomios para **coordenadas:** XH, YH, ZH de la segunda curva espacial para la orientación de un punto de referencia en la herramienta.

En una transformada de seis ejes el posible programar, para la orientación de herramienta, adicionalmente el giro del vector de giro THT con polinomios de máx. 5º grado para giros de la misma herramienta.

Sintaxis

Polinomios de orientación del tipo 1 para **ángulo**

```
N… PO[PHI]=(a2, a3, a4, a5)
N… PO[PSI]=(b2, b3, b4, b5)
```

Transformadas de tres a cinco ejes

Polinomios de orientación del tipo 2 para **coordenadas**

```
N… PO[XH]=(xe, x2, x3, x4, x5)
N… PO[YH]=(ye, y2, y3, y4, y5)
N… PO[ZH]=(ze, z2, z3, z4, z5)
```

Descriptor para las coordenadas de la segunda trayectoria de orientación para la orientación de herramienta

Adicionalmente, se puede programar en ambos casos un polinomio para el **giro** en la transformada de seis ejes con

`N… PO[THT]=(c2, c3, c4, c5)`	Interpolación relativa a la trayectoria del giro
o bien	
`N… PO[THT]=(d2, d3, d4, d5)`	Interpolación absoluta, relativa y tangencial para el cambio de orientación

del vector de orientación. Esto es posible si la transformada soporta un vector de giro con un decalaje programable e interpolable con el ángulo de giro THETA.

Descripción

`PO[PHI]`	Ángulo en el plano entre la orientación inicial y final
`PO[PSI]`	Ángulo del plano entre orientación inicial y final
`PO[THT]`	Ángulo de giro del vector en comando G del grupo 54 programado con THETA
`PHI`	Ángulo de avance LEAD
`PSI`	Ángulo lateral TILT
`THETA`	Giro alrededor de la dirección de herramienta en Z
`PO[XH]`	Coordenada X del punto de referencia en la herramienta
`PO[YH]`	Coordenada Y del punto de referencia en la herramienta
`PO[ZH]`	Coordenada Z del punto de referencia en la herramienta

Información adicional

No se pueden programar polinomios de orientación

- si las interpolaciones spline ASPLINE, BSPLINE, CSPLINE están activas.
  Polinomios del tipo 1 para ángulos de orientación son posibles para cada tipo de interpolación, excepto spline, es decir, en la interpolación lineal con marcha rápida G00 o con avance G01
  en la interpolación de polinomios con POLY y
  en la interpolación circular y de evoluta con G02, G03, CIP, CT, INVCW e INCCCW.
  En cambio, los polinomios del tipo 2 para coordenadas de orientación solo son posibles si está activa la
  interpolación lineal con marcha rápida G00 o con avance G01 o
  la interpolación de polinomios con POLY.
- si la orientación se interpola mediante interpolación de ejes ORIAXES. En este caso se pueden programar directamente polinomios con PO[A] y PO[B] para los ejes de orientación A y B.

Polinomios de orientación del tipo 1 con ORIVECT, ORIPLANE y ORICONxx

En la interpolación circular de gran radio y la interpolación de superficie de cono con ORIVECT, ORIPLANE y ORICONxx solo son posibles polinomios de orientación del tipo 1.

Polinomios de orientación del tipo 2 con ORICURVE

Si la interpolación con curva espacial adicional ORICURVE está activa, se interpolan los componentes cartesianos del vector de orientación y sólo son posibles polinomios de interpolación del tipo 2.

3.9.4 Giros de la orientación de la herramienta (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA)

Si, en tipos de máquina con herramienta móvil, se tiene que poder cambiar también la orientación de la herramienta, se programa cada secuencia con una orientación final. En función de la cinemática de la máquina se pueden programar el sentido de orientación de los ejes de orientación o el sentido de giro del vector de orientación THETA. Para estos vectores de giro se pueden programar distintos tipos de interpolación:

- ORIROTA: Ángulo de rotación frente a un sentido de giro absoluto especificado.
- ORIROTR: Ángulo de rotación relativo al plano entre la orientación inicial y final.
- ORIROTT: Ángulo de rotación relativa a la modificación del vector de orientación.
- ORIROTC: Ángulo de giro tangencial a la trayectoria tangente

Sintaxis

Solo si está activo el tipo de interpolación ORIROTA, el ángulo de giro o el vector de giro se puede programar como sigue de las cuatro formas posibles:

1. Directamente las posiciones de eje giratorio `A, B, C`
2. Ángulo de Euler (en grados) a través de `A2, B2, C2`
3. Ángulo RPY (en grados) a través de `A2, B2, C2`
4. Vector de dirección mediante `A3, B3, C3` (ángulo de giro mediante `THETA=<valor>`).

Si ORIOTR u ORIOTT están activos, el ángulo de giro ya solo se puede programar directamente con `THETA`.

Un giro también se puede programar por sí solo en una secuencia sin que se produzca un cambio de orientación. Entonces, ORIROTR y ORIROTT no tienen ningún significado. En este caso, el ángulo de giro se interpreta siempre con relación a la dirección absoluta (ORIROTA).

`N... ORIROTA`	Definir interpolación del vector de giro
`N... ORIROTR`	
`N... ORIROTT`	
`N... ORIROTC`	
`N... A3= B3= C3= THETA=<valor>`	Definir el giro del vector de orientación
`N... PO[THT]=(`d_2`, `d_3`, `d_4`, `d_5`)`	Interpolar ángulo de giro con polinomio de 5.º grado

Descripción

ORIROTA:	Ángulo de rotación frente a un sentido de giro absoluto especificado
ORIROTR:	Ángulo de rotación relativo al plano entre la orientación inicial y final
ORIROTT:	Ángulo de giro como vector de giro tangencial relativo al cambio de orientación
ORIROTC:	Ángulo de giro como vector de giro tangencial relativo a la tangente de trayectoria
THETA:	Giro del vector de orientación
THETA=<valor>:	Ángulo de rotación en grados que se alcanza al final de la secuencia
THETA=Θe:	Ángulo de giro con ángulo final Θ_e del vector de giro
THETA=AC(…):	Conmutación por secuencias al acotado absoluto
THETA=AC(…):	Conmutación por secuencias al acotado incremental
Θe:	Ángulo final del vector de giro, tanto absoluto con G90 como relativo con G91 (acotado incremental), está activo
PO[THT]=(....):	Polinomio para el ángulo de giro

Ejemplo: Giros de las orientaciones

Código de programa	**Comentarios**
N10 TRAORI	; Activar transformada de orientación
N20 G1 X0 Y0 Z0 F5000	; Orientación de la herramienta
N30 A3=0 B3=0 C3=1 THETA=0	; en dirección Z con el ángulo de giro 0
N40 A3=1 B3=0 C3=0 THETA=90	; en dirección X con giro de 90 grados
N50 A3=0 B3=1 C3=0 PO[THT]=(180,90)	; Orientación
N60 A3=0 B3=1 C3=0 THETA=IC(-90)	; en dirección Y con giro a 180 grados
N70 ORIROTT	; se mantiene constante y giro a 90 grados
N80 A3=1 B3=0 C3=0 THETA=30	; Ángulo de giro relativo al cambio de orientación
	; Vector de giro en un ángulo de 30 grados al plano X-Y

En la interpolación de la secuencia N40, el ángulo de giro se interpola del valor inicial de 0 grados al valor final de 90 grados de forma lineal. En la secuencia N50, el ángulo de giro cambia de 90 grados a 180 grados según la parábola θ(u) = +90u2. En N60 también se puede ejecutar un giro sin que se produzca un cambio de orientación.

En N80, la orientación de herramienta gira de la dirección Y a la dirección X. El cambio de orientación se sitúa en el plano X-Y y el vector de giro forma un ángulo de 30 grados frente a dicho plano.

Información adicional

ORIROTA

El ángulo de rotación THETA se interpola con relación a una dirección absoluta establecida en el espacio. El sentido de giro básico se establece a través de datos de máquina.

ORIROTR

El ángulo de rotación `THETA` se interpreta con relación al plano abierto por la orientación inicial y final.

ORIROTT

El ángulo de rotación `THETA` se interpreta con relación al cambio de orientación. Para `THETA=0`, el vector de giro se interpola tangencialmente al cambio de orientación y se distingue únicamente de `ORIROTR` si, para la orientación, se ha programado al menos un polinomio para el "ángulo de inclinación PSI“. De este modo se obtiene un cambio de orientación que no se desarrolla en el plano. Mediante un ángulo de giro `THETA` programado adicionalmente se puede, por ejemplo, interpolar el vector de giro de tal modo que forma siempre un determinado valor para el cambio de orientación.

ORIROTC

El vector de giro se interpola relativamente a la tangente de trayectoria con una corrección programable a través del ángulo `THETA`. Para el ángulo de decalaje también se puede programar un polinomio `PO[THT]=(c2, c3, c4, c5)` de máx. 5º grado.

3.9.5 Orientaciones relativas a la trayectoria

3.9.5.1 Tipos de orientación relativos a la trayectoria

Con esta función ampliada, la orientación relativa no se alcanza sólo al final de la secuencia, sino a lo largo de toda la trayectoria. La orientación alcanzada en la secuencia anterior se traslada mediante interpolación circular de gran radio a la orientación final programada. Básicamente existen dos posibilidades de programar la orientación deseada relativa a la trayectoria:

1. La orientación de herramienta, así como el giro de la herramienta se interpolan con ORIPATH, ORPATHTS con relación a la trayectoria.
2. El vector de orientación se programa e interpola como de costumbre. Con ORIROTC, el giro del vector de orientación se aplica relativo a la tangente de trayectoria.

Sintaxis

El tipo de interpolación de la orientación y del giro de la herramienta se programa con:

`N... ORIPATH`	Orientación relativa a la trayectoria
`N... ORIPATHS`	Orientación relativa a la trayectoria con alisado del desarrollo de la orientación
`N... ORIROTC`	Interpolación del vector de giro relativa a la trayectoria

Un acodamiento de la orientación producida por un ángulo en el desarrollo de la trayectoria se puede alisar con ORIPATHS. La dirección y la longitud del trayecto del movimiento de levantamiento se programan a través del vector con los componentes A8 = X, B8 = Y y C8 = Z.

Con ORIPATH/ORIPATHS se pueden programar distintas referencias a la tangente de trayectoria a través de los tres ángulos

- LEAD = indicación del ángulo de avance relativo a la trayectoria y la superficie
- TILT = indicación del ángulo lateral relativo a la trayectoria y la superficie
- THETA = ángulo de giro

para toda la trayectoria. Para el ángulo de giro THETA se pueden programar con PO[THT] = (...) polinomios adicionales de máx. 5.º grado.

Nota

Fabricante de la máquina

Preste atención a las indicaciones del fabricante de la máquina. A través de datos de máquina y de operador configurables se pueden realizar otros ajustes para el tipo de orientación relativo a la trayectoria.

Información adicional: Manual de funciones Transformadas

Descripción

La interpolación de los ángulos LEAD y TILT se puede ajustar de distintas maneras a través de datos de máquina:

- La referencia de la orientación de herramienta programada con LEAD y TILT se mantiene a lo largo de toda la secuencia.
- Ángulo de avance LEAD: giro en la dirección perpendicular a la tangente y al vector normal TILT: Giro de la orientación alrededor del vector normal.
- Ángulo de avance LEAD: giro en la dirección perpendicular a la tangente y al vector normal ángulo lateral TILT: Giro de la orientación alrededor de la dirección de la tangente de trayectoria.
- Ángulo de giro THETA: Giro de la herramienta alrededor de sí misma con un tercer eje giratorio adicional como eje de orientación en la transformada de seis ejes.

Nota

Orientación relativa a la trayectoria no permitida con OSC, OSS, OSSE, OSD, OST

La interpolación de orientación relativa a la trayectoria ORIPATH u ORIPATHS y ORIOTC no se pueden programar junto con el alisado del desarrollo de la orientación con uno de los comandos G del grupo 34. Para este fin, OSOF tiene que estar activo.

3.9.5.2 Giro relativo a la trayectoria de la orientación de herramienta (ORIPATH, ORIPATHS, ángulo de giro)

En una transformada de seis ejes, para una orientación libre de la herramienta en el espacio, la herramienta también se puede girar alrededor de sí misma con un tercer eje giratorio. En caso de giro relativo a la trayectoria de la orientación de herramienta con ORIPATH u ORIPATHS, el giro adicional se puede programar a través del ángulo de giro THETA. Como alternativa se pueden programar los ángulos LEAD y TILT a través de un sector en el plano perpendicular a la dirección de herramienta.

Fabricante de la máquina

Preste atención a las indicaciones del fabricante de la máquina. A través de un dato de máquina se puede ajustar la interpolación de los ángulos LEAD y TILT de distintas maneras.

Sintaxis

Giro de la orientación de la herramienta y de la herramienta

El tipo de orientación de herramienta relativo a la trayectoria se activa con ORIPATH o ORIPATHS.

`N... ORIPATH`	Activar tipo de orientación relativa a la trayectoria
`N... ORIPATHS`	Activar tipo de orientación relativa a la trayectoria con alisamiento del desarrollo de la orientación

Activación de los tres ángulos posibles con efecto de giro:

`N... LEAD=`	Ángulo para la orientación programada con relación al vector normal a la superficie
`N... TILT=`	Ángulo para la orientación programada en el plano, vertical respecto a la tangente de trayectoria, relativa respecto al vector normal a la superficie
`N... THETA=`	Ángulo de giro relativo al cambio de orientación alrededor de la dirección de herramienta del tercer eje giratorio

Los valores de los ángulos al final de la secuencia se programan con `LEAD=valor`, `TILT=valor` o `THETA=valor`. Además de los ángulos constantes se pueden programar para los tres ángulos unos polinomios de máx.5º grado.

`N... PO[PHI]=(a2, a3, a4, a5)`	Polinomio para el ángulo de avance LEAD
`N... PO[PSI]=(b2, b3, b4, b5)`	Polinomio para el ángulo lateral TILT
`N... PO[THT]=(d2, d3, d4, d5)`	Polinomio para el ángulo de giro THETA

En la programación se pueden prescindir de los coeficientes de polinomio mayores iguales a cero. Ejemplo: `PO[PHI]=a2` produce para el ángulo de avance `LEAD` una parábola.

Descripción

Orientación de herramienta relativa a la trayectoria

`ORIPATH`:	Orientación de la herramienta con relación a la trayectoria
`ORIPATHS`:	Orientación de la herramienta con relación a la trayectoria; se suaviza un acodamiento en el desarrollo de la orientación
`LEAD`:	Ángulo relativo al vector normal a la superficie en el plano tendido por la tangente de trayectoria y el vector normal a la superficie
`TILT`:	Giro de la orientación en dirección Z o giro alrededor de la tangente de trayectoria
`THETA`:	Giro alrededor de la dirección de herramienta hacia Z
`PO[PHI]`:	Polinomio de orientación para el ángulo de avance LEAD
`PO[PSI]`:	Polinomio de orientación para el ángulo lateral TILT
`PO[THT]`:	Polinomio de orientación para el ángulo de giro THETA

Nota

Ángulo de giro THETA

Para el giro de la herramienta alrededor de sí misma con el tercer eje giratorio como eje de orientación se precisa una transformada de seis ejes.

3.9.5.3 Interpolación relativa a la trayectoria del giro de herramienta (ORIROTC, THETA)

Interpolación con vectores de giro

Para el giro programado con ORIROTC de la herramienta con relación a la tangente de trayectoria, el vector de giro también se puede interpolar con un decalaje programable a través del ángulo de giro THETA. Para el ángulo de decalaje se puede programar con PO[THT] un polinomio de máx. 5.º grado.

Sintaxis

`N... ORIROTC`	Aplicación de un giro de la herramienta con relación a la tangente de trayectoria
`N... A3= B3= C3= THETA=valor`	Definir el giro del vector de orientación
`N... A3= B3= C3= PO[THT]=(c2, c3, c4, c5)`	Interpolar ángulo de decalaje con polinomio de máx. 5º grado

Un giro también se puede programar por sí solo en una secuencia sin que se produzca un cambio de orientación.

Descripción

Interpolación relativa a la trayectoria del giro de la herramienta en la transformada de seis ejes

`ORIROTC`:	Aplicar vector de giro tangencial a la trayectoria tangente
`THETA=valor`:	Ángulo de rotación en grados que se alcanza al final de la secuencia
`THETA=θe`:	Ángulo de giro con ángulo final Θ_e del vector de giro
`THETA=AC(…)`:	Conmutación por secuencias al acotado absoluto
`THETA=IC(…)`:	Conmutación por secuencias al acotado incremental
`PO[THT]=(c2, c3, c4, c5)`:	Interpolar ángulo de decalaje con polinomio de 5º grado

Nota

Interpolación del vector de giro ORIROTC

Si, en contra de la dirección de orientación de la herramienta, también se tiene que aplicar el giro de la herramienta con relación a la tangente de trayectoria, esto sólo es posible con una transformada de seis ejes.

Con ORIROTC activo

No se puede programar el vector de giro ORIROTA. En caso de una programación se emite la ALARMA 14128 "Programación absoluta del giro de herramienta con ORIROTC" activo.

Dirección de orientación de la herramienta en transformada de tres a cinco ejes

Según lo acostumbrado en la transformada de tres a cinco ejes, la dirección de orientación de la herramienta se puede programar a través de ángulos de Euler o RPY o del vector de dirección. Además, son posibles cambios de orientación de la herramienta en el espacio mediante la programación de la interpolación circular de gran radio ORIVECT, la interpolación lineal de los ejes de orientación ORIAXES, todas las interpolaciones en una superficie de cono ORICONxx, así como la interpolación adicionalmente a la curva espacial con dos puntos de contacto de la herramienta ORICURVE.

`G....`:	Indicación del tipo de movimiento de los ejes giratorios
`X, Y, Z`:	Indicación de los ejes lineales
`ORIAXES`:	Interpolación lineal de los ejes de máquina o ejes de orientación
`ORIVECT`:	Interpolación circular de gran radio (idéntico a ORIPLANE)
`ORIMKS`: `ORIWKS`:	Giro en el sistema de coordenadas de máquina Giro en el sistema de coordenadas de pieza Descripción, ver aptdo. Giros de la orientación de la herramienta
`A= B= C=`:	Programación de la posición de eje de la máquina
`ORIEULER`:	Programación de la orientación mediante ángulos eulerianos
`ORIRPY`:	Programación de la orientación de la herramienta con ángulos RPY
`A2= B2= C2=`:	Programación de ángulos de ejes virtuales
`ORIVIRT1`: `ORIVIRT2`:	Programación de la orientación de la herramienta con ejes de orientación virtuales (definición 1), definición según DM $MC_ORIAX_TURN_TAB_1 (definición 2), definición según DM $MC_ORIAX_TURN_TAB_2

A3= B3= C3=:	Programación del vector de dirección del eje de dirección
ORIPLANE:	Interpolación en el plano (interpolación circular de gran radio)
ORICONCW:	Interpolación en una superficie de cono en sentido horario
ORICONCCW:	Interpolación en una superficie de cono en sentido antihorario
ORICONTO:	Interpolación en una superficie envolvente de cono con transición tangencial
A6= B6= C6=:	Programación del eje giratorio del cono (vector normalizado)
NUT=ángulo	Ángulo en el vértice del cono en grados
NUT=+179	Ángulo de desplazamiento menor o igual a 180 grados
NUT=-181	Ángulo de desplazamiento mayor o igual a 180 grados
ORICONIO:	Interpolación en una superficie de cono
A7= B7= C7=:	Orientación intermedia (programación como vector normalizado)
ORICURVE XH YH ZH, p. ej.: con polinomio PO[XH]=(xe, x2, x3, x4, x5)	Interpolación de la orientación con especificación del movimiento de dos puntos de contacto de la herramienta. Además del correspondiente punto final se pueden programar polinomios de curvas de espacio adicionales.

Nota

Si la orientación de herramienta con ORIAXES activos se interpola a través de los ejes de orientación, la aplicación relativa a la trayectoria del ángulo de giro sólo se cumple al final de la secuencia.

3.9.5.4 Alisamiento del desarrollo de la orientación (ORIPATHS A8=, B8=, C8=)

En cambios de orientación con aceleración continua en el contorno no se desean interrupciones de los desplazamiento de trayectoria que se pueden producir especialmente en una esquina del contorno. El acodamiento resultante en el desarrollo de la orientación se puede alisar insertando una secuencia intermedia propia. Entonces, el cambio de orientación se realiza con aceleración continua si, durante la reorientación, también está activo ORIPATHS. En esta fase se puede ejecutar un movimiento de levantamiento de la herramienta.

Fabricante de la máquina

Observe las indicaciones del fabricante de la máquina con respecto a eventuales datos de máquina y de operador predefinidos con los cuales se activa esta función.

A través de un dato de máquina se puede ajustar cómo se interpreta el vector de levantamiento:

1. En el sistema de coordenadas de herramienta se define la coordenada Z por la dirección de la herramienta.
2. En el sistema de coordenadas de pieza se define la coordenada Z por el plano activo.

Encontrará **más información** sobre la función "Orientación relativa a la trayectoria" en el Manual de funciones Transformadas.

Sintaxis

Para orientaciones de herramienta continuas con relación a la trayectoria completa se precisan datos de programación adicionales en una esquina del contorno. La dirección y la longitud del trayecto de este movimiento se programan a través del vector con los componentes A8 = X, B8 = Y y C8 = Z:

```
N... ORIPATHS A8=X B8=Y C8=Z
```

Descripción

`ORIPATHS`:	Orientación de la herramienta con relación a la trayectoria; se suaviza un acodamiento en el desarrollo de la orientación.
`A8= B8= C8=`:	Componentes del vector para dirección y longitud del trayecto
`X, Y, Z`:	Movimiento de levantamiento en la dirección de la herramienta

Nota

Programación del vector de dirección A8, B8, C8

Si la longitud de este vector es igual a cero, no se produce ningún movimiento de levantamiento.

ORIPATHS

La orientación de herramienta relativa a la trayectoria se activa con ORIPATHS. De lo contrario, la orientación se traslada mediante interpolación circular de gran radio lineal de la orientación inicial a la orientación final.

3.9.6 Compresión de la orientación (COMPON, COMPCURV, COMPCAD, COMPSURF)

Los programas CN en los que está activa una transformada de orientación (TRAORI) y están programadas orientaciones de la herramienta (sean del tipo que sean) se pueden comprimir, a condición de que se cumplan las tolerancias especificadas.

Programación

Orientación de la herramienta

Si está activa una transformada de orientación (TRAORI), en las máquinas de 5 ejes se puede programar la orientación de la herramienta de la siguiente manera (independientemente de la cinemática):

- Programación del **vector** de dirección mediante:
  `A3=<...> B3=<...> C3=<...>`
- Programación de los **ángulos** de Euler o los **ángulos** RPY mediante:
  `A2=<...> B2=<...> C2=<...>`

Giro de la herramienta

Para las máquinas de **6 ejes**, se puede programar, además de la orientación de la herramienta, el giro de ésta.

Para programar el ángulo de giro se utiliza:

```
THETA=<...>
```

Ver "Giros de la orientación de la herramienta (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Página 705)".

Nota

Las secuencias CN que contienen adicionalmente un giro solo se pueden comprimir si el ángulo de giro se modifica de modo **lineal**. Por lo tanto, no debe haber programado ningún polinomio con PO[THT] = (...) para el ángulo de giro.

Forma general de una secuencia CN comprimible

En consecuencia, la forma general de una secuencia CN comprimible puede ser como la siguiente:

```
N... X=<...> Y=<...> Z=<...> A3=<...> B3=<...> C3=<...> THETA=<...> F=<...>
```

o bien

```
N... X=<...> Y=<...> Z=<...> A2=<...> B2=<...> C2=<...> THETA=<...> F=<...>
```

Nota

Los valores de posición se pueden indicar de forma directa (p. ej. X90) o de forma indirecta a través de parametrizaciones (p. ej. X=R1*(R2+R3)).

Programación de la orientación de la herramienta por medio de posiciones de ejes giratorios

La orientación de la herramienta se puede indicar también mediante posiciones de eje giratorio, p. ej. en la forma:

```
N... X=<...> Y=<...> Z=<...> A=<...> B=<...> C=<...> THETA=<...> F=<...>
```

En tal caso, la compresión se realiza de dos maneras diferentes, según si se lleva a cabo o no una interpolación circular de gran radio. Si no hay interpolación circular de gran radio, el cambio de orientación comprimido se expresa de la manera habitual, mediante polinomios axiales correspondientes a los ejes giratorios.

Precisión del contorno

Según cuál sea el modo de comprensión ajustado (MD20482 $MC_COMPRESSOR_MODE), para la compresión se aplicarán a los ejes geométricos y a los ejes de orientación las tolerancias específicas de eje configuradas (MD33100 $MA_COMPRESS_POS_TOL) o las siguientes tolerancias específicas de canal, ajustables mediante datos de operador:

SD42475 $SC_COMPRESS_CONTUR_TOL (desviación máxima del contorno)

SD42476 $SC_COMPRESS_ORI_TOL (desviación angular máxima para la orientación de la herramienta)

SD42477 $SC_COMPRESS_ORI_ROT_TOL desviación angular máxima para el ángulo de giro de la herramienta (sólo disponible en máquinas de 6 ejes)

Activación/desactivación

Las funciones del compresor se activan mediante los comandos G modales COMPON, COMPCURV, COMPCAD o COMPSURF.

La función de compresor se finaliza con el comando COMPOF.

Ver "Activar/desactivar compresión de secuencias CN (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF) (Página 621)".

Nota

El movimiento de orientación sólo se comprime si está activa la interpolación circular de gran radio (es decir, el cambio de la orientación de la herramienta tiene lugar en el plano definido por la orientación inicial y final).

También se lleva a cabo una interpolación circular de gran radio en las siguientes circunstancias:

- MD21104 $MC_ORI_IPO_WITH_G_CODE = 0,
  ORIWKS está activo y
  la orientación está programada mediante vectores (con A3, B3, C3 o A2, B2, C2).
- MD21104 $MC_ORI_IPO_WITH_G_CODE = 1 y
  ORIVECT u ORIPLANE activos.
  La orientación de la herramienta puede estar programada como vector de dirección o con posiciones de eje giratorio. Si está activo uno de los comandos G ORICONxx u ORICURVE o hay polinomios programados para los ángulos de orientación (PO[PHI] y PO[PSI]), no se lleva a cabo la interpolación circular de gran radio.

Ejemplo

En el siguiente ejemplo de programa se comprime un círculo aproximado con un trazo polinomial. La orientación de herramienta se mueve de forma síncrona en una superficie de cono. Aunque los cambios de orientación sucesivos programados se producen de forma discontinua, la función del compresor genera un desarrollo liso de la orientación.

```
Programación                          Comentarios
DEF INT CANTIDAD=60
DEF REAL RADIO=20
DEF INT COUNTER
DEF REAL ÁNGULO
N10 G1 X0 Y0 F5000 G64

$SC_COMPRESS_CONTUR_TOL=0.05          ; Desviación máxima del contorno = 0.05 mm
$SC_COMPRESS_ORI_TOL=5                ; Desviación máxima de la orientación
                                      = 5 grados


TRAORI
```

Programación	Comentarios
COMPCURV	; Se ejecuta un arco formado por polígonos. La orientación se mueve en un cono alrededor del eje Z con un ángulo en el vértice de 45 grados.
N100 X0 Y0 A3=0 B3=-1 C3=1	
N110 FOR COUNTER=0 TO CANTIDAD	
N120 ÁNGULO=360*COUNTER/CANTIDAD	
N130 X=RADIO*cos(ÁNGULO) Y=RADIO*sin(ÁNGULO) A3=sin(ÁNGULO) B3=-cos(ÁNGULO) C3=1	
N140 ENDFOR	

3.9.7 Compresión de la orientación (COMPCAD, COMPSURF)

Los programas CN en los que está activa una transformada de orientación (TRAORI) y están programadas orientaciones de la herramienta (sean del tipo que sean) se pueden comprimir, a condición de que se cumplan las tolerancias especificadas.

Programación

Orientación de la herramienta

Si está activa una transformada de orientación (TRAORI), en las máquinas de 5 ejes se puede programar la orientación de la herramienta de la siguiente manera (independientemente de la cinemática):

- Programación del **vector** de dirección mediante:
  `A3=<...> B3=<...> C3=<...>`
- Programación de los **ángulos** de Euler o los **ángulos** RPY mediante:
  `A2=<...> B2=<...> C2=<...>`

Giro de la herramienta

Para las máquinas de **6 ejes**, se puede programar, además de la orientación de la herramienta, el giro de ésta.

Para programar el ángulo de giro se utiliza:

`THETA=<...>`

Ver "Giros de la orientación de la herramienta (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Página 705)".

Nota

Las secuencias CN que contienen adicionalmente un giro solo se pueden comprimir si el ángulo de giro se modifica de modo **lineal**. Por lo tanto, no debe haber programado ningún polinomio con PO[THT] = (...) para el ángulo de giro.

Forma general de una secuencia CN comprimible

En consecuencia, la forma general de una secuencia CN comprimible puede ser como la siguiente:

```
N... X=<...> Y=<...> Z=<...> A3=<...> B3=<...> C3=<...> THETA=<...> F=<...>
```

o bien

```
N... X=<...> Y=<...> Z=<...> A2=<...> B2=<...> C2=<...> THETA=<...> F=<...>
```

Nota

Los valores de posición se pueden indicar de forma directa (p. ej. X90) o de forma indirecta a través de parametrizaciones (p. ej. X=R1*(R2+R3)).

Programación de la orientación de la herramienta por medio de posiciones de ejes giratorios

La orientación de la herramienta se puede indicar también mediante posiciones de eje giratorio, p. ej. en la forma:

```
N... X=<...> Y=<...> Z=<...> A=<...> B=<...> C=<...> THETA=<...> F=<...>
```

En tal caso, la compresión se realiza de dos maneras diferentes, según si se lleva a cabo o no una interpolación circular de gran radio. Si no hay interpolación circular de gran radio, el cambio de orientación comprimido se expresa de la manera habitual, mediante polinomios axiales correspondientes a los ejes giratorios.

Precisión del contorno

Según cuál sea el modo de comprensión ajustado (MD20482 $MC_COMPRESSOR_MODE), para la compresión se aplicarán a los ejes geométricos y a los ejes de orientación las tolerancias específicas de eje configuradas (MD33100 $MA_COMPRESS_POS_TOL) o las siguientes tolerancias específicas de canal, ajustables mediante datos de operador:

SD42475 $SC_COMPRESS_CONTUR_TOL (desviación máxima del contorno)

SD42476 $SC_COMPRESS_ORI_TOL (desviación angular máxima para la orientación de la herramienta)

SD42477 $SC_COMPRESS_ORI_ROT_TOL desviación angular máxima para el ángulo de giro de la herramienta (solo disponible en máquinas de 6 ejes)

Activación/desactivación

Las funciones del compresor se activan mediante los comandos G modales COMPCAD y COMPSURF.

La función de compresor se finaliza con el comando COMPOF.

Ver "AUTOHOTSPOT ".

Nota

El movimiento de orientación solo se comprime si está activa la interpolación circular de gran radio (es decir, el cambio de la orientación de la herramienta tiene lugar en el plano definido por la orientación inicial y final).

También se lleva a cabo una interpolación circular de gran radio en las siguientes circunstancias:

- MD21104 $MC_ORI_IPO_WITH_G_CODE = 0,
  ORIWKS está activo y
  la orientación está programada mediante vectores (con A3, B3, C3 o A2, B2, C2).
- MD21104 $MC_ORI_IPO_WITH_G_CODE = 1 y
  ORIVECT u ORIPLANE activos.
  La orientación de la herramienta puede estar programada como vector de dirección o con posiciones de eje giratorio. Si está activo uno de los comandos G ORICONxx u ORICURVE o hay polinomios programados para los ángulos de orientación (PO[PHI] y PO[PSI]), no se lleva a cabo la interpolación circular de gran radio.

Ejemplo

En el siguiente ejemplo de programa se comprime un círculo aproximado con un trazo polinomial. La orientación de herramienta se mueve de forma síncrona en una superficie de cono. Aunque los cambios de orientación sucesivos programados se producen de forma discontinua, la función del compresor genera un desarrollo liso de la orientación.

```
Programación                              Comentarios
DEF INT CANTIDAD=60
DEF REAL RADIO=20
DEF INT COUNTER
DEF REAL ÁNGULO
N10 G1 X0 Y0 F5000 G64

$SC_COMPRESS_CONTUR_TOL=0.05              ; Desviación máxima del contorno = 0,05 mm
$SC_COMPRESS_ORI_TOL=5                    ; Desviación máxima de la orientación
                                          = 5 grados


TRAORI
COMPCAD                                   ; Se ejecuta un arco formado por polígonos.
                                          La orientación se mueve en un cono alrededor
                                          del eje Z con un ángulo en el vértice de
                                          45 grados.
N100 X0 Y0 A3=0 B3=-1 C3=1
N110 FOR COUNTER=0 TO CANTIDAD
N120 ÁNGULO=360*COUNTER/CANTIDAD
N130 X=RADIO*cos(ÁNGULO) Y=RADIO*sin(ÁNGULO)
A3=sin(ÁNGULO) B3=-cos(ÁNGULO) C3=1
N140 ENDFOR
```

3.9.8 Activar/desactivar alisado del desarrollo de la orientación (ORISON, ORISOF)

El "Alisado de la trayectoria de orientación" se activa/desactiva en el programa de pieza mediante los comandos del grupo G 61. Los comandos son de tipo modal.

Requisitos

- Sistema con transformada de 5/6 ejes.
- La función de compresor COMPCAD está activa.

Sintaxis

```
ORISON
...
ORISOF
```

Descripción

ORISON:	Activar alisado de la trayectoria de orientación
ORISOF:	Desactivar alisado de la trayectoria de orientación

Ejemplo

```
Código de programa                  Comentarios
...
TRAORI()                            ; Activación de la transformada de orientación.
COMPCAD                             ; Activación de la función de compresor COMPCAD.
ORISON                              ; Activación del alisado de la orientación.
$SC_ORISON_TOL=1.0                  ; Desviación angular máxima de la orientación de la he-
                                    rramienta = 1,0 grados.
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                              ; Desactivación del alisado de la orientación.
...
```

La orientación gira 90 grados en el plano XZ de -45 a +45 grados. Con el alisado del desarrollo de la orientación, la orientación ya no alcanza los valores angulares máximos de -45 o +45 grados.

3.9.9 Transformación cinemática

3.9.9.1 Conexión de la transformada de lados frontales (TRANSMIT)

La transformada de lados frontales (TRANSMIT) se activa en el programa de pieza o en la acción síncrona a través de la instrucción `TRANSMIT`.

Sintaxis

```
TRANSMIT
TRANSMIT(<n>)
```

Descripción

`TRANSMIT`:	Activar TRANSMIT con el primer registro TRANSMIT
`TRANSMIT(n)`:	Activar TRANSMIT con el <n>-simo registro TRANSMIT

Nota

Una transformada TRANSMIT activa en el canal se desactiva mediante:

- Desactivar transformada: `TRAFOOF`
- Activación de otra transformada: p. ej., `TRACYL`, `TRAANG`, `TRAORI`

3.9.9.2 Conexión de la transformada de envolvente de cilindro (TRACYL)

La transformada de envolvente de cilindro (TRACYL) se activa en el programa de pieza o en la acción síncrona a través de la instrucción `TRACYL`.

Sintaxis

```
TRACYL(<d>)
TRACYL(<d>,<n>)
TRACYL(<d>,<n>,<k>)
```

Descripción

`TRACYL(<d>):`	Activar TRACYL con el primer registro TRACYL y el diámetro de trabajo <d>	
`TRACYL (<d>,<n>):`	Activar TRACYL con el <n>-simo registro TRACYL y el diámetro de trabajo <d>	
`<d>`:	Diámetro de referencia y de trabajo El valor debe ser mayor que 1.	
`<n>`:	Número de registro TRACYL (opcional)	
	Rango de valores:	1, 2
`<k>`:	El parámetro <k> solo es relevante para el tipo de transformada 514	
	k = 0:	sin corrección de pared de ranura
	k = 1:	con corrección de pared de ranura
	Si no se indica el parámetro, tiene efecto el ajuste básico parametrizado: $MC_TRACYL_DEFAULT_MODE_<n> donde <n> = número de registro de TRACYL	

Nota

Una transformada TRACYL activa en el canal se desactiva mediante:

- Desactivar transformada: `TRAFOOF`
- Activación de otra transformada: p. ej., `TRAANG`, `TRANSMIT`, `TRAORI`

Ejemplo

Código de programa	**Comentarios**
`...`	
`N40 TRACYL(40.)`	`; Activar TRACYL con el primer registro TRACYL y el diámetro de trabajo 40 mm.`
`...`	

Información adicional

Estructura de programa

Un programa de pieza para el fresado de una ranura con la transformada TRACYL 513 (TRACYL con corrección de pared de ranura) consta en general de los siguientes pasos:

1. Seleccionar herramienta.
2. Seleccionar TRACYL.
3. Seleccionar el decalaje de coordenadas adecuado (FRAME).
4. Posicionar.
5. Programar OFFN.
6. Seleccionar corrección de radio de herramienta.

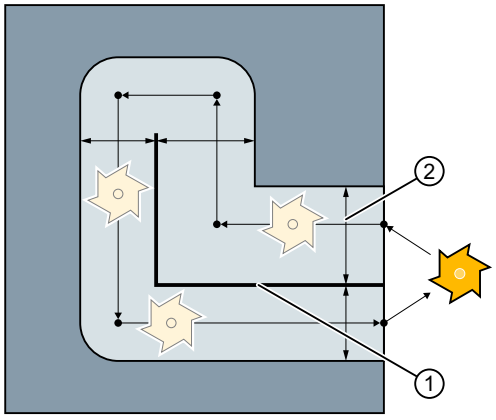
7. Secuencia de desplazamiento (entrada de la corrección de radio de herramienta y aproximación a la pared de ranura).
8. Contorno de la línea central de la ranura.
9. Cancelar corrección de radio de herramienta.
10. Secuencia de retirada (salida de la corrección de radio de herramienta y retirada de la pared de ranura).
11. Posicionar.
12. TRAFOOF.
13. Volver a seleccionar el decalaje de coordenadas (FRAME) original.

Corrección de contorno (OFFN)

Para fresar ranuras con la transformada TRACYL 513, en el programa de pieza se programa la línea central de la ranura y, a través de la dirección OFFN, **la mitad del ancho de la ranura**.

Para evitar daños en la pared de la ranura, OFFN solo se activa con la corrección del radio de herramienta seleccionada.

Se puede modificar OFFN dentro del programa de pieza. Esto permite desplazar del centro la línea central de la ranura:



① OFFN
② Trayectoria programada

Nota

OFFN debe ser al menos igual al tamaño del radio de la herramienta para excluir daños en la pared opuesta de la ranura.

Nota

OFFN con TRACYL actúa de forma diferente que sin TRACYL. Dado que OFFN se incluye también sin TRACYL cuando está activa la corrección de radio de herramienta, OFFN se debería volver a poner a cero después de TRAFOOF.

ATENCIÓN
El efecto de OFFN depende del tipo de transformada
En la transformada TRACYL 513 (TRACYL con corrección de pared de ranura), se programa para OFFN la mitad del ancho de la ranura.
Por el contrario, en la transformada TRACYL 512 (TRACYL sin corrección de pared de ranura) el valor de OFFN actúa como creces para la corrección del radio de herramienta.

Corrección del radio de herramienta (WRK)

En la transformada TRACYL 513, la corrección del radio de herramienta no se calcula en relación con la pared de ranura, sino con respecto al centro programado de la ranura. Para que la herramienta se desplace a la izquierda de la pared de la ranura, debe programarse la instrucción G41 en lugar de G42, o bien indicarse el valor de OFFN con signo negativo.

Diámetro de la herramienta

Con TRACYL y una herramienta cuyo diámetro sea menor que el ancho de la ranura, no se genera la misma geometría de pared de ranura que con una herramienta cuyo diámetro sea igual al ancho de la ranura. Para una mejor precisión se recomienda optar por una herramienta que tenga un diámetro ligeramente menor que el ancho de la ranura.

Uso de los ejes

Nota

Los siguientes ejes no se pueden utilizar como eje de posicionado o eje de vaivén:

- El eje geométrico en el sentido circunferencial de la envolvente del cilindro (eje Y)
- El eje lineal adicional en la corrección de la pared de ranura (eje Z)

3.9.9.3 Activar transformada de ángulo inclinado con ángulo programable (TRAANG)

La transformada de ángulo inclinado con ángulo programable se activa en el programa de pieza o en la acción síncrona a través de la instrucción `TRAANG`.

Sintaxis

```
TRAANG
TRAANG()
TRAANG(, <n>)
TRAANG(<α>)
TRAANG(<α>,<n>)
```

Descripción

`TRAANG`: `TRAANG()`:	Activar TRAANG con el primer registro TRAANG y el último ángulo válido <α>	
`TRAANG(,<n>)`:	Activar TRAANG con el <n>-simo registro TRAANG y el último ángulo válido <α>	
`TRAANG(<α>)`:	Activar TRAANG con el primer registro TRAANG y el ángulo <α>	
`TRAANG(<α>,<n>)`:	Activar TRAANG con el <n>-simo registro TRAANG y el ángulo <α>	
<α>:	Ángulo del eje inclinado (opcional)	
	Rango de valores:	-90° < α < + 90°
	Si no se especifica ningún ángulo, se aplica la posición básica parametrizada en el dato de máquina: MD2xxxx $MC_TRAANG_ANGLE_<n>	
`<n>`:	Número del registro TRAANG (opcional)	
	Rango de valores:	1, 2

Nota

Una transformada de ángulo inclinado `TRAANG` activa en el canal se desactiva mediante:

- Desactivar transformada: `TRAFOOF`
- Activación de otra transformada: p. ej., `TRACYL`, `TRANSMIT`, `TRAORI`

Ejemplo

Código de programa	**Comentarios**
`N20 TRAANG(45)`	`; Activar TRAANG con el primer registro TRAANG y el ángulo 45°.`

3.9.9.4 Ranurado oblicuo en rectificadoras (G5, G7)

Los comandos G7 y G5 simplifican la programación del ranurado oblicuo en rectificadoras con la transformada "Eje oblicuo (TRAANG)", de manera que al ranurar únicamente se desplaza el eje oblicuo.

A la hora de programar únicamente se dispone de la posición final deseada del movimiento de ranurado en X y Z. La posición inicial correspondiente es calculada por el CN con la función G7, que también sirve para el desplazamiento a dicha posición, partiendo de la posición actual del eje X, de la posición final programada y del ángulo α del eje oblicuo.

La posición inicial resulta del punto de intersección de las dos rectas:

- Recta paralela al eje Z a la distancia de la posición actual del eje X
- Recta paralela al eje oblicuo a través de la posición final programada

El eje oblicuo se desplaza a la posición final programada con la función G5.

Sintaxis

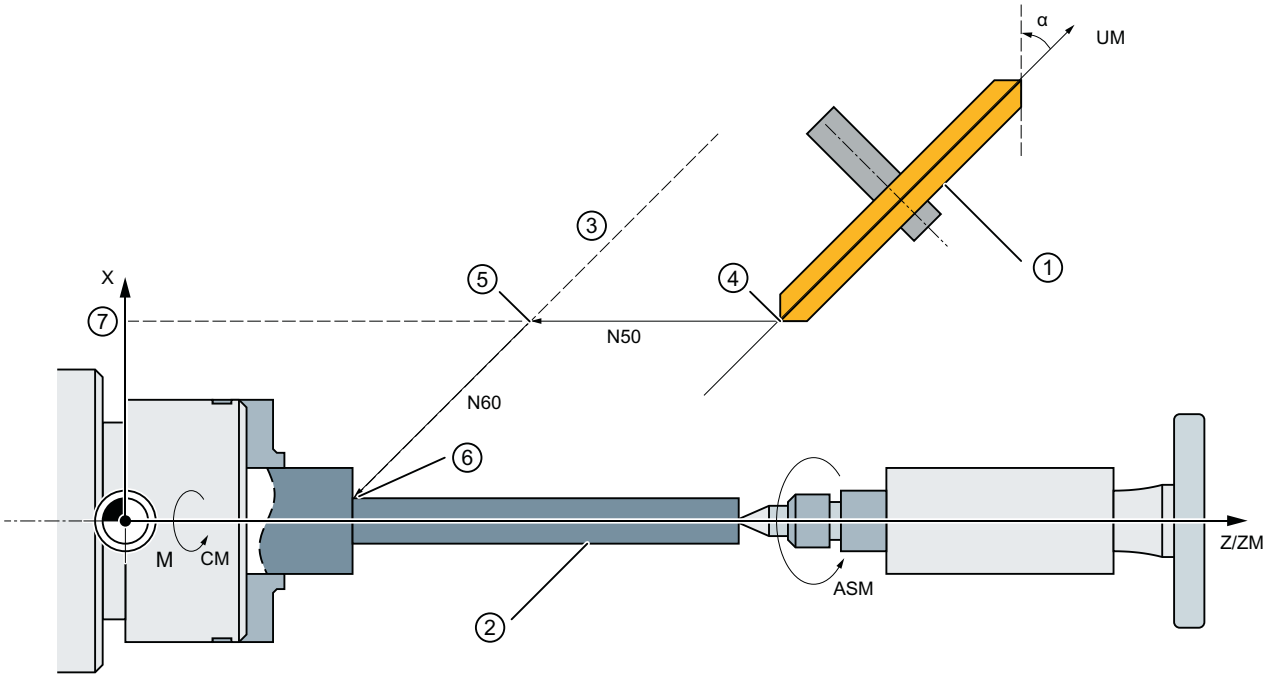
```
G7 <Endpos_X> <Endpos_Z>
```

```
G5 <Endpos_X>
```

Descripción

G7:	Calcular la posición inicial para el ranurado oblicuo y desplazar a dicha posición
G5:	Desplazamiento del eje oblicuo a la posición final programada
<Endpos_X>:	Posición final del eje X
<Endpos_Z>:	Posición final del eje Z

Ejemplo



① Muela
② Pieza
③ Paralela al eje oblicuo a través de la posición final programada
④ Posición de salida
⑤ Ranurado: Posición inicial
⑥ Ranurado: posición final
⑦ Paralela al eje Z a la distancia de la posición actual del eje X
X Eje geométrico
Z Eje geométrico
ZM (MC) Eje de máquina
UM Eje de máquina

Figura 3-5 Programar eje oblicuo

Código de programa	**Comentarios**
N... G18	; Seleccionar plano XZ.
N40 TRAANG(45.0)	; Activar transformada TRAANG, ángulo = 45°.
N50 G7 X40 Z70 F4000	; Calcular posición inicial y desplazar a dicha posición.
N60 G5 X40 F100	; Desplazar eje oblicuo a posición final.
N70 ...	

3.9.10 Conexión de la transformada concatenada (TRACON)

Una transformada concatenada configurada se activa en el programa de pieza o en la acción síncrona a través de la instrucción TRACON.

Sintaxis

```
TRACON(<N.°_transf>,<Par_1>,...,<Par_n>,<Par_n+1>)
...
TRAFOOF
```

Descripción

TRACON:	Activar transformada concatenada Otra transformada activada previamente es desactivada implícitamente por TRACON().		
<N°_transf>:	Número de la transformada concatenada		
	Tipo:	INT	
	Rango de valores:	0 ... 2	
	Valor:	0, 1	Primera/única transformada concatenada
		2	Segunda transformada concatenada
		Sin especificar	Mismo significado con 0 o 1
		Nota: Valores distintos a 0, 1, 2 generan una alarma de error.	

<Par_1>, ..., <Par_n>, <Par_n+1>:	Parámetros para las transformadas concatenadas entre sí	
	<Par_1>, ..., <Par_n>	Parámetros de la primera transformada de la cadena El número efectivo depende del tipo de transformada: • TRAORI: 2 parámetros • TRACYL: 1 o 2 parámetros
	<Par_n+1>	Parámetros de la segunda transformada de la cadena (TRAANG) ≙ Ángulo del eje oblicuo
	Si no se definen parámetros, se aplican los ajustes previos o los últimos parámetros utilizados. Colocando comas, se tiene que asegurar que los parámetros indicados se evalúan en el orden en el cual se esperan si tienen que actuar ajustes previos para parámetros anteriores. En particular, en caso de indicación de al menos un parámetro, este tiene que ir precedido de una coma aunque no sea necesario indicar <N.º_transf>, por ejemplo, TRACON(, 3.7).	
TRAFOOF:	Desactivar la última transformada activada (concatenada)	

Ejemplo

Código de programa	Comentarios
...	
N230 TRACON(1,45.)	; Activar primera transformada concatenada.
	; La transformada activa previamente se cancela automáticamente.
	; El ángulo para el eje oblicuo es de 45°.
...	
N330 TRACON(2,40.)	; Activar segunda transformada concatenada.
	; El ángulo para el eje oblicuo es de 40°.
...	
N380 TRAFOOF	; Desactivar segunda transformada concatenada.
...	

3.9.11 Desplazamiento PTP cartesiano

3.9.11.1 Conexión/desconexión del desplazamiento cartesiano PTP (PTP, PTPG0, PTPWOC, CP)

El desplazamiento cartesiano punto a punto o PTP se conecta/desconecta en el programa CN con los comandos del grupo G 49.

Los comandos son de tipo modal. El ajuste predeterminado es el desplazamiento de trayectoria cartesiano (CP).

A diferencia de CP, con el desplazamiento PTP activo solo se ejecuta una transformada del punto final cartesiano, y los ejes de máquina se desplazan de forma síncrona.

Para convertir el punto final cartesiano de forma unívoca en valores de ejes de máquina, además de indicaciones de posición y de ángulo se requiere información que identifique las posiciones de eje. Estos datos se producen mediante las direcciones ajustables STAT (Página 729) y TU (Página 733).

Requisitos

La transformada TRAORI, TRANSMIT, RCTRA o ROBX está activa.

Sintaxis

```
PTP/PTPG0/PTPWOC
...
CP
```

Descripción

PTP:	Activar el desplazamiento punto a punto PTP La posición cartesiana programada en secuencias G0 y G1 se alcanza con un movimiento de eje síncrono.
PTPG0:	Activar el desplazamiento punto a punto PTPG0 La posición cartesiana programada solo se alcanza en secuencias G0 con un movimiento de eje síncrono. En secuencias G1 se cambia al desplazamiento de trayectoria CP.
PTPWOC:	Activar el desplazamiento punto a punto PTPWOC (posible solo si la transformada de orientación está activa) Igual que PTP, pero sin desplazamientos de compensación causados por los desplazamientos de ejes giratorios y ejes de orientación.
CP:	Desactivar el desplazamiento punto a punto y activar el desplazamiento de trayectoria CP Con CP se ejecuta un desplazamiento de trayectoria cartesiano.

Nota

PTPWOC

La aplicación de PTPWOC en combinación con una transformada RCTRA o ROBX no es conveniente.

Ejemplos

Ver:

- Ejemplo 1: desplazamiento PTP de un robot de 6 ejes con transformada ROBX (Página 736)
- Ejemplo 2: Desplazamiento PTP en transformada genérica de 5 ejes (Página 737)
- Ejemplo 3: PTPG0 y TRANSMIT (Página 737)

3.9.11.2 Indicación de la posición de las articulaciones (STAT)

La indicación de la posición mediante coordenadas cartesianas y la indicación de la orientación de la herramienta no bastan para determinar la posición de la máquina inequívocamente, ya que son posibles varias posiciones de las articulaciones con la misma orientación de la herramienta. Dependiendo de la cinemática existen hasta 8 posiciones de las articulaciones diferentes. Estas posiciones de las articulaciones diferentes son específicas de transformada.

Para evitar ambigüedades, la posición de las articulaciones se indica bajo la dirección STAT.

Nota

El control solo tiene en cuenta valores STAT programados en desplazamientos PTP. En desplazamientos CP se ignoran, ya que en el desplazamiento con transformada activa no suele ser posible cambiar la posición. En el desplazamiento con CP activo se transfiere la posición para el punto final desde el punto inicial.

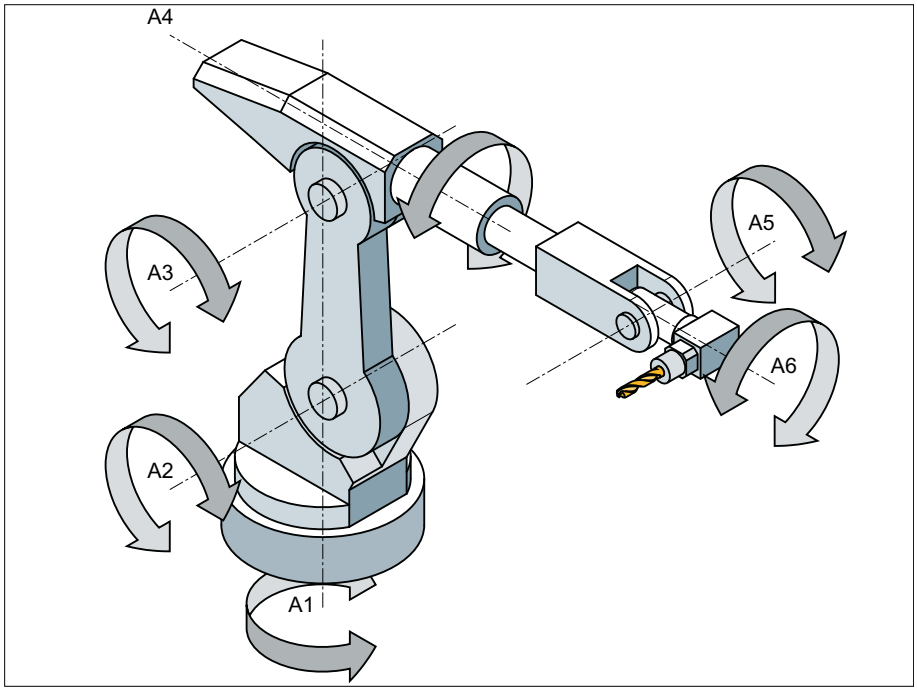
Sintaxis

```
STAT=<Valor>
```

Descripción

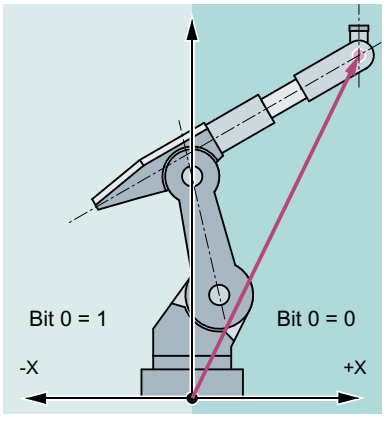
`STAT`:	Dirección ajustable para indicar las posiciones de las articulaciones
`<Valor>`:	Valor binario o decimal Contiene un bit para cada una de las posiciones posibles. El significado de los bits está determinado por la transformada correspondiente.

La aplicación de STAT se ilustra tomando como ejemplo un robot de brazo articulado de 6 ejes con cabezal de fresado. La transformada cinemática se realiza mediante la transformada de robot ROBX (requisito: el ciclo compilado "RMCC/ROBX Transformada robótica ampliada" está cargado y activo).

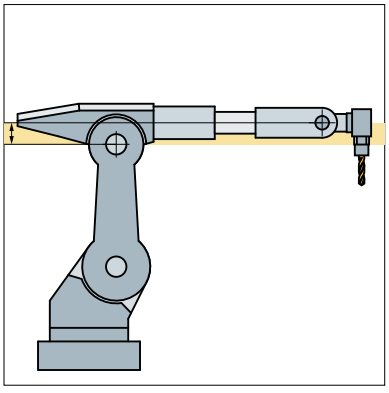


Los ejes A1, A2 y A3 son los ejes principales del robot de brazo articulado. Los ejes A4, A5 y A6, denominados ejes de cabezal o de mano, se posicionan con los ejes principales en la zona de trabajo. Las posibilidades de movimiento adicionales de los ejes de mano permiten orientar el cabezal de fresado en el espacio como lo requiere la tarea de mecanizado. Una misma orientación de la herramienta admite diferentes posiciones de las articulaciones.

La selección de las posiciones de las articulaciones necesarias para el mecanizado se realiza mediante la programación del bit 0 ... 2 de la dirección ajustable STAT:

<table>
<tr><td rowspan="3">Bit 0</td><td colspan="2">Posición del punto de intersección de los ejes de mano (A4, A5, A6)</td><td rowspan="3"> 

Ejemplo: el punto de intersección de los ejes de mano está situado en el área inferior</td></tr>
<tr><td>= 0</td><td>Área inferior (Shoulder Right)
El robot se encuentra en el área inferior cuando el valor X del punto de intersección de los ejes de mano es positivo en relación con el sistema de coordenadas A1.</td></tr>
<tr><td>= 1</td><td>Área superior (Shoulder Left)
El robot se encuentra en el área superior cuando el valor X del punto de intersección de los ejes de mano es negativo en relación con el sistema de coordenadas A1.</td></tr>
</table>

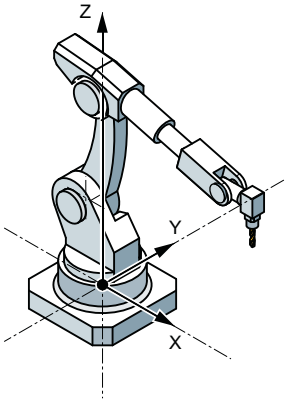
Bit 1	Posición del eje 3 El ángulo en el que se modifica el valor de bit 1 depende del tipo de robot. Para robots cuyos ejes 3 y 4 se cortan, se aplica lo siguiente:	
	= 0	A3<0° (**Elbow Down**)
	= 1	A3≥0° (**Elbow Up**)
	Nota: En robots con un decalaje entre el eje 3 y el 4, el ángulo con que se modifica el valor de bit 1 depende del tamaño de este decalaje.	 Decalaje entre A3 y A4
Bit 2	Posición del eje 5	
	= 0	A5≥0° (**No Handflip**)
	= 1	A5<0° (**Handflip**)

Ejemplo de programación:

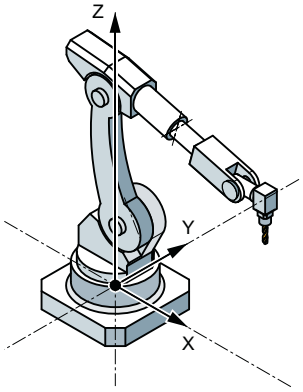
Código de programa	**Comentarios**
...	
N14 T="T8MILLD20" D1	; $TC_DP3[1,1]=132.95
N16 ORIMKS	
N17 G1 PTP X1665.67 Y0 Z1377.405 A=0 B=0 C=0 **STAT=...** F2000	; El valor STAT determina las posiciones de las articulaciones (ver abajo).
...	

`STAT=1 ('B001')`

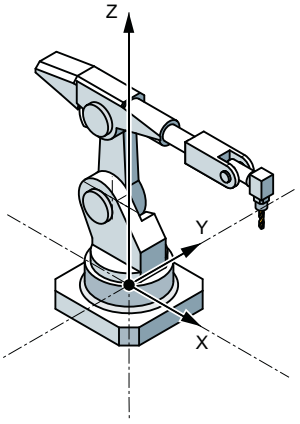
→ Shoulder Left
→ Elbow Down
→ No Handflip



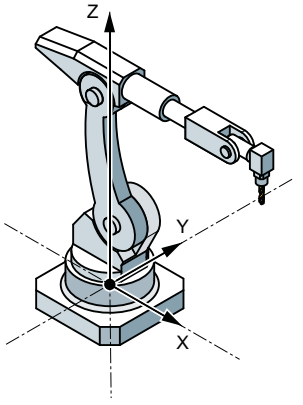
`STAT=2 ('B010')` → Shoulder Right
→ Elbow Up
→ No Handflip



`STAT=5 ('B101')` → Shoulder Left
→ Elbow Down
→ Handflip



`STAT=6 ('B110')` → Shoulder Right
→ Elbow Up
→ Handflip



TRANSMIT

En TRANSMIT se utiliza la dirección STAT para deshacer la ambigüedad en relación con el polo.

Si el eje giratorio debe girar 180º o el contorno en CP pasase a través del polo, se aplica:

Bit 0	Solo relevante con $MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 1 o 2:	
	= 0	El eje giratorio se desplaza +180º o gira en sentido horario.
	= 1	El eje giratorio se desplaza -180º o gira en sentido antihorario.
Bit 1	Solo relevante con $MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 0:	
	= 0	Se desplaza a través del polo. El eje giratorio no gira.
	= 1	Gira en torno al polo. El bit 0 de STAT es relevante.

3.9.11.3 Indicación de los signos de los ángulos de eje (TU)

Para poder posicionarse en ángulos de eje de más de +180° o menos de -180° con ejes giratorios sin una estrategia de desplazamiento especial (p. ej., puntos intermedios), deben indicarse los signos de los ángulos de eje debajo de la dirección TU ajustable.

Nota

El control solo tiene en cuenta valores TU programados en desplazamientos PTP. En desplazamientos CP se ignoran.

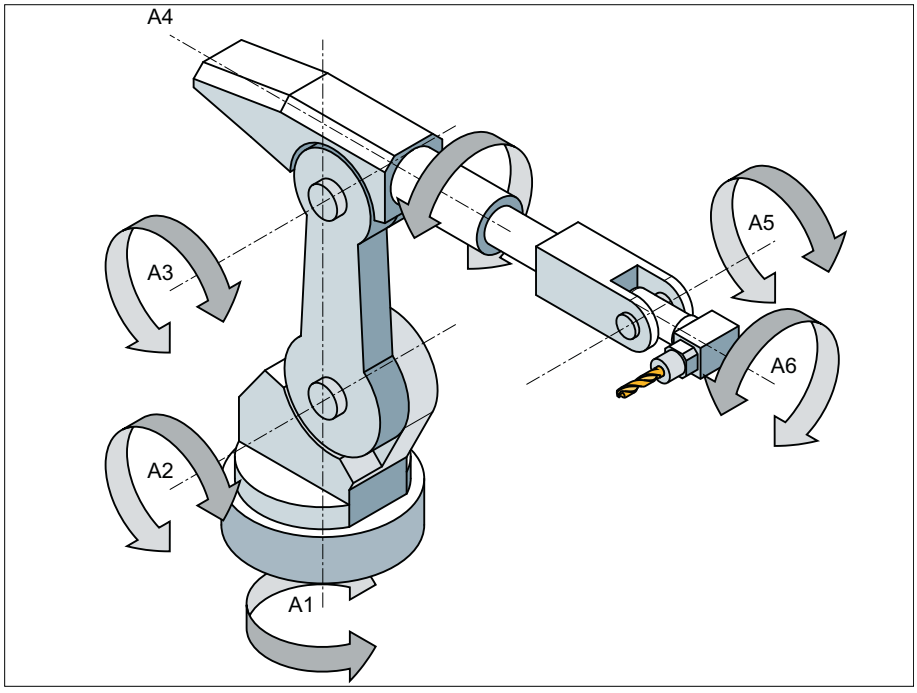
Sintaxis

```
TU=<Valor>
```

Descripción

`TU`:	Dirección ajustable para indicar los signos de los ángulos de eje.			
`<Valor>`:	Valor binario o decimal Para cada eje que se ve afectado por la transformada existe un bit que muestra el signo del ángulo de eje (θ) y, por lo tanto, el sentido de desplazamiento.			
	Bit	= 0	Signo de los ángulos de eje: +	Rango de ángulos de eje: 0° ≤ θ < 360°
		= 1	Signo de los ángulos de eje: -	Rango de ángulos de eje: –360° < θ < 0°

Ejemplo: robot de brazo articulado de 6 ejes



Bit	Descripción	Valor	Signo del ángulo de eje	Ángulo del eje
Bit 0 [1]	Signo del ángulo del eje A1	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 1 [1]	Signo del ángulo del eje A2	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 2 [1]	Signo del ángulo del eje A3	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 3 [1]	Signo del ángulo del eje A4	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 4 [1]	Signo del ángulo del eje A5	= 0	+	≥ 0°
		= 1	-	< 0°
Bit 5 [1]	Signo del ángulo del eje A6	= 0	+	≥ 0°
		= 1	-	< 0°

[1] Los números de bit TU reales se obtienen a partir de los números de ejes de canal de los ejes robóticos. En el ejemplo, los ejes robóticos (A1 a A6) son los seis primeros ejes del canal, de modo que se utilizan los bits TU 0 ... 5. En una asignación de ejes de canal diferente para los ejes robóticos se modificarían oportunamente los números de bit TU de los ejes robóticos (p. ej.: los ejes robóticos son el 3.er al 8.º eje de canal, es decir, se utilizan los bits TU 2 ... 7 para los ejes robóticos).

TU=19 (corresponde a TU='B010011) entonces significaría:

Bit	Valor		Ángulo de eje
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	⇒	$\theta_{A6} \geq 0°$

Nota

En ejes con una zona de desplazamiento > ±360°, el desplazamiento se realiza siempre por el trayecto más corto, ya que la posición del eje no es unívoca a través de la información de TU.

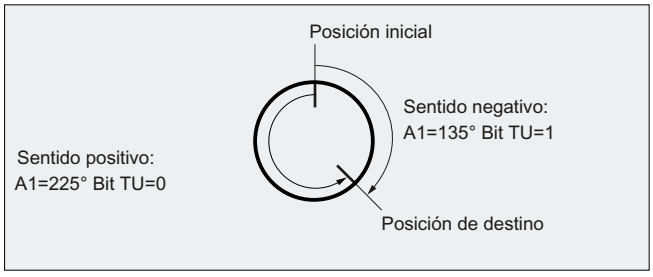
Si no se programa ningún TU en una posición, en función de MD30455 $MA_MISC_FUNCTION_MASK se recorre el trayecto corto o largo.

TRANSMIT

En el desplazamiento PTP con TRANSMIT activo, la dirección TU no es relevante.

Ejemplo

La posición de eje giratorio indicada en la siguiente figura puede alcanzarse en sentido negativo o positivo. En la dirección A1 se programa la posición angular. El sentido de desplazamiento no es unívoco hasta que no se indica TU.



3.9.11.4 Ejemplo 1: desplazamiento PTP de un robot de 6 ejes con transformada ROBX

En el siguiente ejemplo de aplicación se muestra a modo de ejemplo el desplazamiento cartesiano PTP y los comandos CN relacionados.

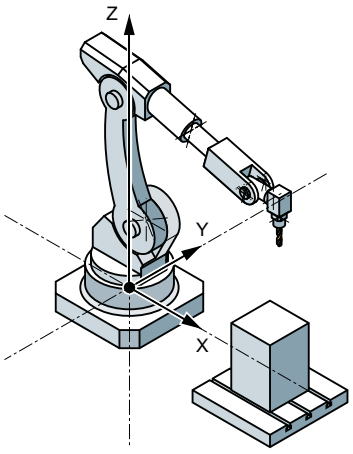


Figura 3-6 Robot de brazo articulado de 6 ejes con cabezal de fresado

```
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
```

3.9.11.5 Ejemplo 2: Desplazamiento PTP en transformada genérica de 5 ejes

Supuesto: Se tiene una cinemática CA rectangular.

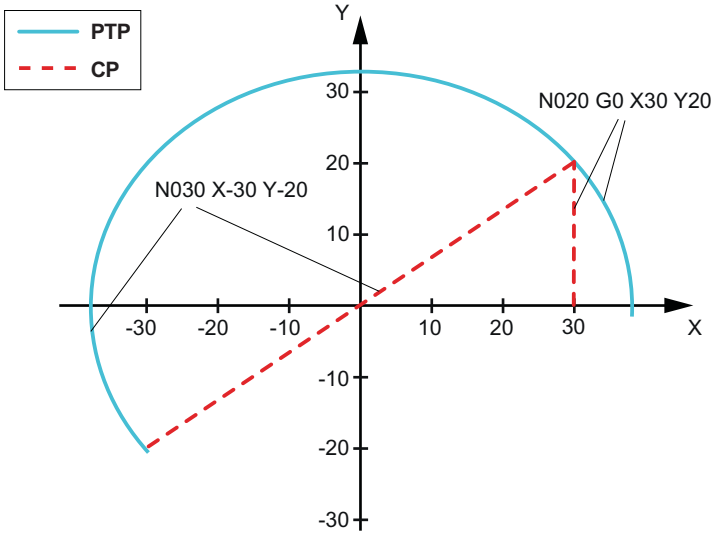
Código de programa	Comentarios
TRAORI	; Transformada de cinemática CA activada
PTP	; Conectar desplazamiento PTP
N10 A3=0 B3=0 C3=1	; Posiciones de ejes giratorios C=0 A=0
N20 A3=1 B3=0 C3=1	; Posiciones de ejes giratorios C=90 A=45
N30 A3=1 B3=0 C3=0	; Posiciones de ejes giratorios C=90 A=90
N40 A3=1 B3=0 C3=1 **STAT=1**	; Posiciones de ejes giratorios C=270 A=–45

Seleccionar posición de desplazamiento unívoca para la posición de los ejes giratorios:

De esta forma, en la secuencia N40 los ejes giratorios se desplazan mediante la programación de **STAT=1** por el trayecto más largo desde su punto de inicio (C=90, A=90) hasta el punto final (C=270, A=–45). En cambio, con **STAT=0**, los ejes giratorios se desplazarían por el trayecto más corto hasta el punto final (C=90, A=45).

3.9.11.6 Ejemplo 3: PTPG0 y TRANSMIT

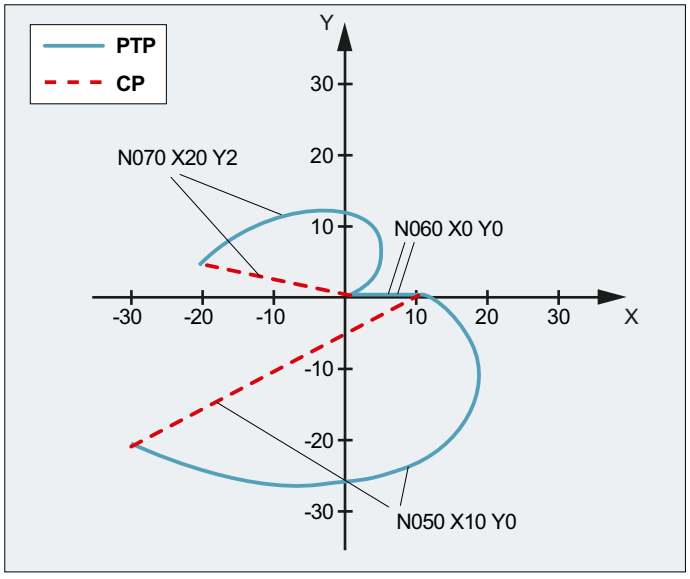
Desplazamiento alrededor del polo con PTPG0 y TRANSMIT



Código de programa	Comentarios
N001 G0 X30 Z0 F10000 T1 D1 G90	; Posición inicial acotado absoluto
N002 SPOS=0	
N003 **TRANSMIT**	; Transformada TRANSMIT
N010 **PTPG0**	; Para cada secuencia G0 automáticamente PTP y luego de nuevo CP.
N020 G0 X30 Y20	
N030 X-30 Y-20	

Código de programa	Comentarios
N120 G1 X30 Y20	
N110 X30 Y0	
M30	

Retirada del polo con PTPG0 y TRANSMIT



Programación	Comentarios
N001 G0 X90 Z0 F10000 T1 D1 G90	; Posición inicial
N002 SPOS=0	
N003 **TRANSMIT**	; Transformada TRANSMIT
N010 **PTPG0**	; Para cada secuencia G0 automáticamente PTP y luego de nuevo 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	

3.9.12 Limitaciones en la selección de una transformada

Función

La selección de transformadas es posible a través del programa de pieza o MDA. Se deberá de tener en cuenta:

- Una secuencia intermedia de desplazamiento no se inserta (chaflanes/radios).
- Una sucesión de secuencias spline tiene que estar terminada; de lo contrario, aparece un mensaje.
- La corrección de herramienta fina tiene que estar cancelada (FTOCOF); de lo contrario, aparece un mensaje.
- La corrección de radio de herramienta tiene que estar cancelada (G40); de lo contrario, aparece un mensaje.
- Una corrección de longitud de herramienta activada es adoptada por el control a la transformada.
- El frame actual activo antes de la transformada es cancelado por el control.
- Una limitación del campo de trabajo activa es cancelada por el control para los ejes afectadas por la transformada (equivale a WALIMOF).
- La vigilancia de la zona protegida se cancela.
- El modo de contorneado y el matado de esquinas se interrumpen.
- Todos los ejes indicados en el dato de máquina tienen que estar sincronizados con relación a la secuencia.
- En los ejes invertidos se deshace la inversión; de lo contrario, aparece un mensaje.
- En ejes dependientes se emite un mensaje.

Cambio de herramienta

Un cambio de herramienta sólo se admite si la corrección del radio de herramienta está cancelada.

Un cambio de la corrección de longitud de herramienta y la selección/cancelación de la corrección del radio de herramienta no se deben programar en la misma secuencia.

Cambio de frame

Se admiten todas las instrucciones que se refieren únicamente en el sistema de coordenadas básico (FRAME, corrección del radio de herramienta). Sin embargo, a diferencia de la transformada activa, un cambio de frame con G91 (medida incremental) no se trata por separado. El incremento a ejecutar se evalúa en el sistema de coordenadas de pieza del nuevo frame, independientemente del frame que actuaba en la secuencia anterior.

Exclusiones

Los ejes afectados por la transformada no se pueden utilizar:

- como eje Preset (alarma),
- para el desplazamiento a punto fijo (alarma),
- para el referenciado (alarma).

3.9.13 Cancelar transformada (TRAFOOF)

Con el procedimiento predefinido TRAFOOF se desactivan todas las transformadas y todos los frames activos.

Nota

Para la cancelación de la transformada se aplican las mismas condiciones (Página 739) que para la selección.

Los frames que se necesitan a continuación se tienen que activar mediante una nueva programación.

Sintaxis

```
...
TRAFOOF
```

Descripción

`TRAFOOF`:	Desactiva todas las transformadas/frames activos

3.10 Cadenas cinemáticas

3.10.1 Eliminación de componentes (DELOBJ)

La función `DELOBJ()` "borra" componentes restableciendo las variables de sistema asignadas a su valor por defecto:

- Elementos de cadenas cinemáticas
- Zonas protegidas, elementos de zonas protegidas y pares de colisión
- Datos de transformadas

Sintaxis

```
[<RetVal>=] DELOBJ(<CompType>[,,,<NoAlarm>)])
[<RetVal>=] DELOBJ(<CompType>,<Index1>[,,<NoAlarm>])
[<RetVal>=] DELOBJ(<CompType>[,<Index1>][,<Index2>][,<NoAlarm>])
```

Descripción

DELOBJ:	Eliminación de elementos de cadenas cinemáticas, zonas protegidas, elementos de zonas protegidas, pares de colisión y datos de transformadas
<CompType>:	Tipo de componente que se desea eliminar
	Tipo de dato: STRING
	Valor: "KIN_CHAIN_ELEM" **Descripción**: Variables de sistema de todos los elementos cinemáticos: $NK_...
	Valor: "KIN_CHAIN_SWITCH" **Descripción**: Variable de sistema $NK_SWITCH[<i>]
	Valor: "KIN_CHAIN_ALL" **Descripción**: Todos los elementos cinemáticos e interruptores. Mismo significado con la llamada sucesiva de DELOBJ con "KIN_CHAIN_ELEM" y "KIN_CHAIN_SWITCH"
	Valor: "PROT_AREA" **Descripción**: Variables de sistema de las zonas protegidas: • $NP_PROT_NAME • $NP_CHAIN_NAME • $NP_CHAIN_ELEM • $NP_1ST_PROT
	Valor: "PROT_AREA_ELEM" **Descripción**: Variables de sistema de los elementos de zonas protegidas de máquina o zonas protegidas de herramienta automáticas: • $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
	Valor: "PROT_AREA_COLL_PAIRS" **Descripción**: Variables de sistema de los pares de colisión: • $NP_COLL_PAIR • $NP_SAFETY_DIST
	Valor: "PROT_AREA_ALL" **Descripción**: Todas las zonas protegidas, elementos de zonas protegidas y pares de colisión (variables de sistema $NP_...) Mismo significado que la llamada sucesiva de DELOBJ con "PROT_AREA", "PROT_AREA_ELEM" y "PROT_AREA_COLL_PAIRS"
	Valor: "TRAFO_DATA" **Descripción**: Variables de sistema de todas las transformadas $NT_...

<Index1>:	Índice del primer componente que se desea eliminar (**opcional**)	
	Tipo de dato:	INT
	Valor por defecto:	-1
	Rango de valores:	-1 ≤ x ≤ (número máximo de componentes configurados - 1)
	Valor	**Descripción**
	0, 1, 2...	Índice de los componentes que se desea eliminar.
	-1	Se eliminan todos los componentes del tipo especificado. <Index2> no se evalúa.
<Index2>:	Índice del último componente que se desea eliminar (**opcional**) Si <Index2> no está programado, solo se eliminarán las variables de sistema del componente referenciado en <Index1>.	
	Tipo de dato:	INT
	Valor por defecto:	Solo se eliminarán las variables de sistema del componente referenciado en <Index1>.
	Rango de valores:	<Index1> < x ≤ (número máx. de componentes configurados - 1)
<NoAlarm>:	Supresión de alarmas (**opcional**)	
	Tipo de dato:	BOOL
	Valor por defecto:	FALSE
	Valor	**Descripción**
	FALSE	En caso de error (<RetVal> < 0) se detiene la ejecución del programa y se emite una alarma.
	TRUE	En caso de error no se detiene la ejecución del programa ni se emite alarma alguna. Caso de aplicación: reacción específica de usuario, conforme al valor de retorno
<RetVal>:	Valor de retorno de la función	
	Tipo de dato:	INT
	Rango de valores:	0, -1, -2 ... -7
	Valor	**Descripción**
	0	No se ha producido ningún fallo.
	-1	Llamada de la función sin parámetros. Debe especificarse por lo menos el parámetro <CompType>.
	-2	<CompType> indica componente desconocido
	-3	<Index1> es menor que -1
	-4	<Index1> es mayor que el número configurado de componentes
	-5	<Index1> tiene un valor distinto a -1 en el momento de la eliminación de un **grupo** de componentes
	-6	<Index2> es menor que <Index1>
	-7	<Index2> es mayor que el número configurado de componentes

3.10.2 Determinación del índice a partir del nombre (NAMETOINT)

En los elementos de matriz de sistema del tipo STRING se guardan nombres específicos del usuario. Debido al identificador de las variables de sistema y del nombre, la función `NAMETOINT()` determina el valor de índice que pertenece al nombre y bajo el cual este se ha guardado en el elemento de matriz de sistema.

Sintaxis

```
<RetVal> = NAMETOINT(<SysVar>,<Name>[,<NoAlarm>])
```

Descripción

`NAMETOINT`:	Determinación del índice de una variable de sistema	
`<SysVar>`:	Nombre del elemento de matriz de sistema del tipo STRING	
	Tipo de dato:	STRING
	Rango de valores:	Nombres de todos los elementos de matriz de sistema del CN del tipo STRING
`<Name>`:	Cadena de caracteres o nombre a partir del cual se determinará el índice de la variable de sistema.	
	Tipo de dato:	STRING
`<NoAlarm>`:	Supresión de alarmas (**opcional**)	
	Tipo de dato:	BOOL
	Valor por defecto:	FALSE
	Valor	**Descripción**
	TRUE	En caso de error no se detiene la ejecución del programa ni se emite alarma alguna. Caso de aplicación: reacción específica de usuario, conforme al valor de retorno
	FALSE	En caso de error (<RetVal> < 0) se detiene la ejecución del programa y se emite una alarma.
`<RetVal>`:	Índice de variable de sistema o aviso de error	
	Tipo de dato:	INT
	Rango de valores:	-1 ≤ x ≤ (número máx. de componentes configurados - 1)
	Valor	**Descripción**
	≥ 0	El nombre buscado se ha encontrado bajo el índice de variable de sistema indicado.
	-1	No se ha encontrado el nombre buscado o se ha producido un error.

Ejemplo

Código de programa	**Comentarios**
`DEF INT INDEX`	

Código de programa	**Comentarios**
`$NP_PROT_NAME[27] = "Cubierta"`	
`...`	
`INDEX = NAMETOINT("$NP_PROT_NAME","Cubierta")`	`; INDEX == 27`

3.11 Prevención de colisiones con cadenas cinemáticas

Nota

Zonas protegidas

Las zonas protegidas que se mencionan en los siguientes capítulos se refieren a la función "Modelado geométrico de máquina".

Encontrará **información** sobre esta función en el Manual de funciones Vigilancia y compensación.

3.11.1 Comprobación de par de colisión (COLLPAIR)

La función `COLLPAIR(...)` determina si dos zonas protegidas forman un par de colisión.

Sintaxis

```
[<RetVal> =] COLLPAIR(<Name_1>,<Name_2>[,<NoAlarm>)])
```

Descripción

`COLLPAIR`:	Comprobar la pertenencia a un par de colisión		
`<RetVal>`:	Valor de retorno de la función		
	Tipo de dato:	INT	
	Valor:	≥ 0	Ambas zonas protegidas forman un par de colisión. Valor de retorno == índice de par de colisión m (ver $NP_COLL_PAIR)
		-1	Se han indicado menos de dos cadenas, o al menos una de ellas es la cadena cero.
		-2	No se encuentra la zona protegida indicada en el primer parámetro.
		-3	No se encuentra la zona protegida indicada en el segundo parámetro.
		-4	No se encuentra ninguna de las dos zonas protegidas indicadas.
		-5	Se han encontrado las dos zonas protegidas indicadas, pero no forman parte de un mismo par de colisión.
`<Name_1>`:	Nombre de la primera zona protegida		
	Tipo de dato:	STRING	
	Rango de valores:	Nombres parametrizados de las zonas protegidas	
`<Name_2>`:	Nombre de la segunda zona protegida		
	Tipo de dato:	STRING	
	Rango de valores:	Nombres parametrizados de las zonas protegidas	

<NoAlarm>:	Supresión de alarmas **(opcional)**		
	Tipo de dato:	BOOL	
	Valor:	FALSE (Predeterminado)	En caso de error (<RetVal> < 0) se detiene la ejecución del programa y se emite una alarma.
		TRUE	En caso de error no se detiene la ejecución del programa ni se emite alarma alguna. Caso de aplicación: reacción específica de usuario, conforme al valor de retorno

3.11.2 Solicitar nuevo cálculo del modelo de máquina de la prevención de colisiones (PROTA)

Si en el programa de pieza se escriben variables de sistema de la secuencia cinemática $NK_..., del modelado geométrico de máquina o de la prevención de colisiones $NP_..., a continuación se debe llamar el procedimiento PROTA para que el cambio sea efectivo en el modelo de máquina interno del CN de la prevención de colisiones.

Sintaxis

```
PROTA[(<Par>)]
```

Descripción

PROTA:	Solicitar nuevo cálculo del modelo de máquina de la prevención de colisiones		
	• Desencadena una parada de decodificación previa. • Debe encontrarse sola en la secuencia.		
<Par>:	Parámetro (**opcional**)		
	Tipo de dato:	STRING	
	Valor:	---	Sin parámetro. Se recalcula el modelo de máquina. Los estados de las zonas protegidas se conservan.
		"R"	Se recalcula el modelo de máquina. Se asigna a las zonas protegidas su respectivo estado de inicialización conforme a $NP_INIT_STAT.

Condiciones

Simulación

El procedimiento PROTA no debe utilizarse en programas de pieza en combinación con la simulación (simNC).

Ejemplo: Prevención de la llamada PROTA mientras está activa la simulación.

Código de programa	Comentarios
...	

Código de programa	**Comentarios**
`IF $P_SIM == FALSE`	`; IF simulación no activa`
`   PROTA`	`; THEN recalcular modelo de colisión`
`ENDIF`	`; ENDIF`
`...`	

Consulte también

Asignar estado a la zona protegida (PROTS) (Página 748)

3.11.3 Asignar estado a la zona protegida (PROTS)

El procedimiento `PROTS(...)` asigna el estado de zonas protegidas conforme al valor indicado.

Sintaxis

```
PROTS(<State>[,<Name_1>,...,<Name_n>])
```

Descripción

`PROTS`:	Asignar el estado de zonas protegidas		
	• Debe encontrarse sola en la secuencia.		
`<State>`:	Estado que se asignará a las zonas protegidas indicadas		
	Tipo de dato:	CHAR	
	Valor:	"A" o "a"	Estado: Activo
		"I" o "i"	Estado: Inactivo
		"P" o "p"	Estado: preactivado o controlado por PLC [1]
		"R" o "r"	Estado: Valor interno del CN del estado de inicialización [2]
`<Name_1> ... <Name_n>`:	Nombre de una o varias zonas protegidas a las que se desea asignar el estado indicado (**opcional**) Si no se indica ningún nombre, se asignará el estado indicado a todas las zonas protegidas definidas.		
	Tipo de dato:	STRING	
	Rango de valores:	Nombres parametrizados de las zonas protegidas	
	Nota: El número máximo de zonas protegidas que pueden indicarse como parámetro depende únicamente del máximo posible de caracteres por línea de código.		
[1] La activación / desactivación se realiza a través de: DB10.DBX234.0 - DBX241.7 [2] El estado adopta el valor interno del CN del estado de inicialización, es decir, el valor que tenía la variable de sistema $NP_INIT_STAT en el instante de la última llamada de PROTA(...) (Página 747).			

3.11.4 Determinación de la distancia de dos zonas protegidas (PROTD)

La función `PROTD(...)` calcula la distancia de dos zonas protegidas.

Propiedades de la función:

- El cálculo de distancia se realiza independientemente del estado de las zonas protegidas (activadas, desactivadas o preactivadas).
- Para el cálculo de distancia de dos zonas protegidas se consideran solo los elementos de zona protegida identificados mediante $NP_USAGE = "C" o "A". Los elementos de la zona protegida identificados con $NP_USAGE = "V" no se tienen en cuenta.
- Las zonas protegidas en las que todos los elementos de la zona protegida están identificados con $NP_USAGE = "V" no pueden utilizarse para el cálculo de distancia.
- El cálculo de distancia se realiza con las posiciones válidas al final de la secuencia anterior.
- Las superposiciones que se tienen en cuenta para el cálculo en el proceso principal, p. ej. decalaje DRF o decalajes de origen externos, se incluyen en el cálculo de distancia con los valores vigentes en el **momento de interpretación** de la función.

Nota

Sincronización

Si se utiliza la función `PROTD(...)`, es responsabilidad exclusiva del usuario sincronizar, en caso necesario, el proceso principal y la decodificación previa mediante una parada de decodificación previa `STOPRE`.

Colisión

Si hay una colisión entre las zonas protegidas indicadas, la función devuelve una distancia de 0,0. Se produce una colisión cuando las dos zonas protegidas se tocan o penetran.

La distancia de seguridad para la comprobación de colisión (MD10622 $MN_COLLISION_SAFETY_DIST) no se tiene en cuenta para el cálculo de la distancia.

Sintaxis

```
[<RetVal> =] PROTD([<Name_1>],[<Name_2>],VAR <Vector>[,<System>])
```

Descripción

`PROTD`:	Calcular la distancia de las dos zonas protegidas indicadas.	
	• Debe encontrarse sola en la secuencia.	
`<RetVal>`:	Valor de retorno de la función: Valor absoluto de la distancia de las dos zonas protegidas o 0,0 en caso de existir colisión (ver arriba apartado "Colisión")	
	Tipo de dato:	REAL
	Rango de valores:	0,0 ≤ x ≤ +máx. valor REAL

<Name_1>, <Name_2>:	Nombres de las dos zonas protegidas cuya separación se desea calcular (**opcional**)		
	Tipo de dato:	STRING	
	Rango de valores:	Nombres parametrizados de zonas protegidas	
	Valor por defecto:	"" (Cadena vacía) Si no se indica ninguna zona protegida, la función calcula la menor distancia actual a partir de todas las zonas protegidas activadas y preactivadas contenidas en el modelo de colisión.	
<Vector>:	Valor de retorno: Vector tridimensional de la distancia entre la zona protegida <Name_2> y la zona protegida <Name_1> con: • <Vector>[0]: coordenada X en el sistema de coordenadas mundial • <Vector>[1]: coordenada Y en el sistema de coordenadas mundial • <Vector>[2]: coordenada Z en el sistema de coordenadas mundial En caso de colisión: <Vector> == vector nulo		
	Tipo de dato:	VAR REAL [3]	
	Rango de valores:	<Vector> [n]: 0,0 ≤ x ≤ ±valor REAL máx.	
<System>:	Sistema de medida (pulgada/métrico) para la distancia y el vector de distancia (**opcional**)		
	Tipo de dato:	BOOL	
	Valor:	FALSE (Predeterminado)	Sistema de medida conforme al comando G actualmente activo del grupo G 13 (G70, G71, G700, G710).
		TRUE	Sistema de medida de acuerdo con el sistema base ajustado: MD52806 $MN_ISO_SCALING_SYSTEM

3.12 Transformadas con cadenas cinemáticas

3.12.1 Activar transformada (TRAFOOF)

Se activa una transformada definida con cadenas cinemáticas con el procedimiento predefinido TRAFOON. La llamada debe encontrarse por sí sola en una secuencia.

Nota

Una transformada definida con cadenas cinemáticas también puede activarse con comandos de programación convencionales como, por ejemplo, TRAORI o TRANSMIT. Para ello, en la variable de sistema $NT_TRAFO_INDEX debe estar introducido un valor distinto de cero.

Encontrará más información sobre $NT_TRAFO_INDEX en el "Manual de listas Variables del sistema".

Sintaxis

```
TRAFOON(<Trafoname>,<Diameter>,<k>)
```

Descripción

TRAFOON:	Procedimiento activar una transformada definida con cadenas cinemáticas		
<Trafoname>:	Nombre del juego de datos de la transformada		
	Tipo de dato:	STRING	
	Rango de valores:	Todos los nombres de juegos de datos de transformada definidos con $NT_NAME	
	Nota: El nombre del juego de datos de transformada debe ser inequívoco. Solo puede aparecer una vez en $NT_NAME.		
<Diameter>:	Diámetro de referencia y de trabajo (solo TRACYL)		
	Tipo de dato:	REAL	
	El valor debe ser >1.		
<k>:	Define la utilización de la corrección de pared de ranura (solo TRACYL).		
	Tipo de dato:	BOOL	
	Valor:	FALSE	sin corrección de pared de ranura
		TRUE	con corrección de pared de ranura
	Corresponde al tipo de transformada TRACYL 514 (corrección de pared de ranura programable). Si no se indica <k>, se aplica el ajuste parametrizado del bit 10 en $NT_CNTRL[<n>].		

Ejemplo

Código de programa	Comentarios
`TRAFOON["Trans_1"]`	`Activa la transformada con el nombre Trans_1.`

3.12.2 Modificar la transformada de orientación después de la medición de la máquina (CORRTRAFO)

En las máquinas con transformadas de orientación definidas mediante cadenas cinemáticas, el usuario puede utilizar la función predefinida CORRTRAFO para modificar los vectores de offset o los vectores de dirección de los ejes de orientación en el modelo cinemático de la máquina después de una medición de la máquina.

Nota

Los valores de corrección escritos con la función CORRTRAFO no son efectivos de inmediato en la transformada. Los valores de corrección solo se aplican tras cancelación de la transformada, NEWCONF y selección de la transformada.

Sintaxis

```
<Corr_Status> = CORRTRAFO(<Corr_Vect>, <Corr_Index>, <Corr_Mode>,
[ <No_Alarm>])
```

Descripción

CORRTRAFO:	Llamada de función

<Corr_Status>:	Valor de retorno de la función		
	Tipo de dato:	INT	
	Valores:	0	La función se ha ejecutado sin errores.
		1	No hay ninguna transformada activa.
		2	La transformada activa actualmente no es una transformada de orientación.
		3	La transformada de orientación activa no se ha definido con cadenas cinemáticas.
		10	El parámetro de llamada <Corr_Index> es negativo.
		11	El parámetro de llamada <Corr_Mode> es negativo.
		12	Referencia no válida a una sección de una subcadena (dígito de unidades de <Corr_Index>). El valor no puede ser mayor que el número de ejes de orientación de la subcadena.
		13	Referencia no válida al eje de orientación de una subcadena (dígito de unidades de <Corr_Index>). El valor debe ser menor que el número de ejes de orientación de la subcadena.
		14	Referencia no válida a una subcadena (dígito de decenas de <Corr_Index>). Solo se permiten los valores 0 y 1 (referencia a cadena de pieza o de herramienta). Este número de error aparece también cuando no existe la subcadena a la que hace referencia <Corr_Index>.
		15	En la sección a la que se hace referencia con el parámetro <Corr_Index> no se ha definido ningún elemento de corrección ($NT_CORR_ELEM_P o $NT_CORR_ELEM_T).
		20	Modo de corrección no válido (dígito de unidades de <Corr_Mode>). Solo se permiten los valores 0 y 1.
		21	Modo de corrección no válido (dígito de decenas o de centenas de <Corr_Mode>). Al escribir una dirección de eje, solo puede ser distinto de cero el dígito de unidades.
		30	El dígito de centenas de <Corr_Mode> no es válido. Solo se permiten los valores 0 y 1.
		31	El dígito de millares de <Corr_Mode> no es válido. Solo se permiten los valores 0 y 1.
		40	El vector de dirección que se debe adoptar como dirección de eje es el vector cero. Este error solo puede aparecer cuando el dígito de millares de <Corr_Mode> es igual a 0. Si el dígito de millares de este parámetro es igual a 1 (vigilancia de la corrección máxima desactivada), también puede escribirse el vector nulo.
		41	En la corrección de un vector de offset, la desviación respecto a los valores actuales es mayor que el valor máximo especificado por el dato de operador SD41610 $SN_CORR_TRAFO_LIN_MAX en al menos una coordenada. El parámetro <Corr_Vect> se sobrescribe por un vector de error. Esto también sucede cuando se interrumpe el mecanizado con una alarma (ver parámetro <No_Alarm>). En los componentes cuyo valor de corrección ha sobrepasado el límite permitido, el vector de error contiene la diferencia con

			signo correcto entre valor de corrección determinado y el límite. El contenido de los componentes que no han sobrepasado su límite es cero.
		42	En la corrección de un vector de dirección, la desviación angular respecto a la dirección actual es mayor que el valor máximo especificado por el dato de operador SD41611 $SN_CORR_TRAFO_DIR_MAX.
		43	Se ha rechazado el intento de escribir una variable de sistema debido a la falta de derechos de escritura.
<Corr_Vect>:	Vector de corrección El contenido del vector de corrección se define mediante los parámetros siguientes **<**Corr_Index**>** y <Corr_Mode>. Si <Corr_Status> = 41, se sobrescribe el contenido del vector (ver arriba).		
	Tipo de dato:	REAL	
<Corr_Index**>**:	Sección cuyo elemento de corrección debe modificarse/ Índice del eje de orientación cuyo vector de dirección debe modificarse		
	Tipo de dato:	INT	
	El parámetro <Corr_Index> está codificado en decimal (dígito de unidades a decenas):		
	Dígito de unidades:	Contiene el índice de la sección o del eje de orientación en la subcadena.	
	Dígito de decenas:	Hace referencia a la subcadena.	
		0x	Cadena de piezas
		1x	Cadena de herramientas

<Corr_Mode>:	Modo de corrección		
	Tipo de dato:	INT	
	El parámetro <Corr_Mode> está codificado en decimal (dígito de unidades a millares):		
	Dígito de unidades:	Determina el elemento que debe ser corregido.	
		xxx**0**	Corrección de un vector de offset lineal
		xxx**1**	Corrección del vector de dirección de un eje de orientación
	Dígito de decenas:	Determina cómo debe modificarse el elemento de corrección al que hace referencia el contenido de <Corr_Index>.	
		xx**0**x	El vector de corrección se escribe inmediatamente en el elemento de corrección. Esta variante puede utilizarse para escribir inmediatamente el elemento de corrección sin que tenga que conocerse el índice <n> del dato del sistema en cuestión ($NK_OFF_DIR[<n>, ...]).
		xx**1**x	Como 0, pero con la diferencia de que el valor de corrección transferido se interpreta en coordenadas mundiales. Siempre puede surgir una diferencia entre las variantes 0 y 1 si la cadena cinemática incluye giros adicionales en la posición básica (posiciones de todos los ejes de orientación iguales a 0).
		xx**2**x	Como 1, pero con la diferencia de que el valor de corrección hace referencia a la sección completa, es decir: en el elemento de corrección se ha introducido un valor con el que la sección completa alcanza la longitud definida a través del valor de corrección.
		Nota: No se permiten los valores 1 y 2 para escribir el vector de dirección de un eje de orientación.	
	Dígito de centenas:	Determina cómo hay que interpretar el contenido del parámetro <Corr_Vect>.	
		x**0**xx	El vector de corrección transferido <Corr_Vect> contiene la nueva longitud completa del elemento de corrección o de la sección a los que hace referencia <Corr_Index> en combinación con el dígito de decenas de <Corr_Mode> (corrección absoluta).
		x**1**xx	El vector de corrección transferido <Corr_Vect> contiene solo la diferencia respecto a la longitud actual del elemento de corrección o de la sección a los que hace referencia <Corr_Index> en combinación con el dígito de decenas de <Corr_Mode> (corrección incremental).
		Nota: En la corrección del vector de dirección de un eje de orientación el contenido del dígito de centenas debe ser 0.	
	Dígito de millares:	Determina si la corrección debe estar limitada por el siguiente valor máximo: • SD41610 $SN_CORR_TRAFO_LIN_MAX o bien • SD41611 $SN_CORR_TRAFO_DIR_MAX	
		0xxx	La vigilancia de la corrección máxima está activa.

		1xxx	La vigilancia de la corrección máxima no está activa.
<No_Alarm>:	Respuesta en caso de error (valor de retorno >0) (**opcional**)		
	Tipo de dato:	BOOL	
	Valor:	FALSE (predeterminado)	En caso de error, se detiene la ejecución del programa y se muestra la alarma 14103.
		TRUE	En caso de error no se detiene la ejecución del programa ni se emite alarma alguna. Caso de aplicación: reacción específica de usuario, conforme al valor de retorno

Nota

Si aparece un error al llamar la función, se genera una alarma o se devuelve un número de error (ver parámetro <No_Alarm>), de forma que el propio usuario pueda reaccionar de manera adecuada al estado de error. La causa del error se identifica de forma más precisa con un parámetro de alarma. Un número de error devuelto en lugar de la alarma es idéntico al parámetro de alarma.

Más información sobre CORRTRAFO

La estructura cinemática de una máquina con transformada de orientación se describe a través de una o dos cadenas cinemáticas (subcadenas) que parten del origen del sistema de coordenadas mundial. Una de las dos cadenas, la **cadena de herramientas**, finaliza en el punto de referencia de la herramienta; la otra, la **cadena de piezas**, en el origen del sistema de coordenadas básico.

La función CORRTRAFO escribe longitudes de brazo de palanca y direcciones de eje en elementos de corrección especiales en máquinas con transformada de orientación. Una cadena cinemática se describe, entre otros, con elementos del tipo OFFSET, que se definen mediante $NK_TYPE.

CORRTRAFO trabaja con secciones

Las dos subcadenas se pueden dividir cada una en cuatro secciones como máximo:

- La sección 1 empieza en el punto inicial de la cadena y termina en el primer eje de orientación.
- La sección 2 es la que hay entre el eje de orientación 1 y el eje de orientación 2.
- La sección 3 es la que hay entre el eje de orientación 2 y el eje de orientación 3.
- La sección 4 es la que hay entre el eje de orientación 3 y el final de la cadena de herramientas o de piezas.

Cada una de las secciones puede contener elementos de cadena constantes del tipo OFFSET o ROT_CONST.

El gráfico siguiente muestra una transformada de orientación con 2 ejes de orientación.

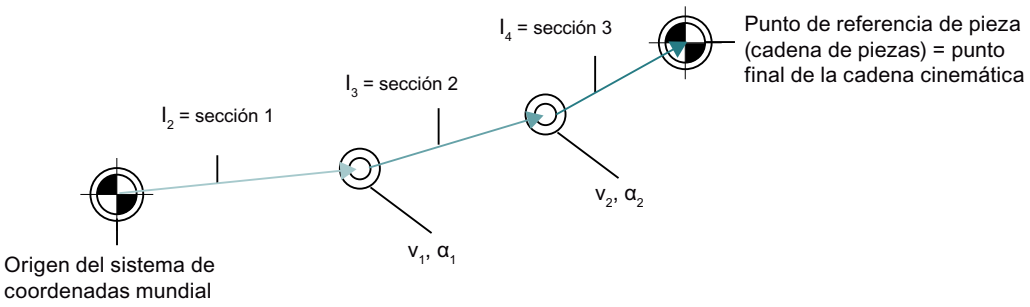


Figura 3-7 Ejemplo de CORRTRAFO

Las secciones se definen de forma unívoca: Cuando se recorre la subcadena cinemática desde el punto inicial hasta el punto final, la primera sección tiene el índice 0, la siguiente el índice 1 y así sucesivamente. El índice de la última sección es siempre igual al número de ejes de orientación.

Elementos de corrección

Se puede hacer referencia a cada elemento de cadena cinemática constante (elemento de cadena del tipo $NK_TYPE[<n>] = "OFFSET") de estas secciones con las variables de sistema $NT_CORR_ELEM_T[<n>, 0 ... 3] o $NT_CORR_ELEM_P[<n>, 0 ... 3]. En estos elementos denominados de este modo se introducen por medio de la función CORRTRAFO los valores de corrección determinados en la medición de máquina.

Ejemplo con índice de transformada = 1:

- $NT_CORR_ELEM_T[1,0] = "C_AXIS_OFFSET"; el offset del eje C (eje de orientación 1) en la sección 1 está definido como elemento de corrección.
- $NT_CORR_ELEM_T[1,1] = "B_AXIS_OFFSET"; el offset del eje B (eje de orientación 2) en la sección 2 está definido como elemento de corrección.
- $NT_CORR_ELEM_T[1,2] = "BASE_TOOL_OFFSET"; el offset del eje B al punto de referencia de la herramienta en la sección 3 está definido como elemento de corrección.

El orden de las referencias en $NT_CORR_ELEM_T/P[<n>, 0 ... 3] debe corresponderse con las secciones descritas más arriba, es decir, en $NT_CORR_ELEM_T/P [<n>, 0] solo puede haber un elemento de cadena que se encuentre antes del primer eje de orientación, y así sucesivamente.

La función CORRTRAFO escribe en los elementos de corrección definidos de este modo los valores determinados mediante la medición de la máquina. La modificación de los valores de corrección se define en CORRTRAFO mediante el parámetro <Corr_Mode>.

Cerrar cadena

Si se han seteado el bit 7 o el bit 8 en la variable de sistema $NT_CNTRL[<n>], al final de la cadena de piezas (bit 7) o antes del punto de inicio de la cadena de herramientas (bit 8), se insertan internamente de forma automática elementos de cadena constantes adicionales que conectan el punto final de la cadena y el origen de la máquina ("Cerrar cadena").

Estos elementos insertados de forma automática no pueden escribirse desde el exterior, sino solo ser leídos (ver al respecto las variables de sistema $AC_TRAFO_CORR_ELEM_P/T).

Punto para el cierre de la cadena de herramientas

Si la variable de sistema $NT_CLOSE_CHAIN_T no está vacía, la cadena de herramientas no se cierra en el punto final de la cadena, sino en el punto final del elemento de cadena designado. Los otros elementos de cadena que se encuentran detrás de este punto llevan a un correspondiente decalaje de origen cuando se activa la transformada.

Índice de un eje de orientación

Además del offset constante entre los ejes de orientación, con la función CORRTRAFO también se pueden describir los vectores de dirección de los ejes de orientación. Por índice de un eje de orientación se entiende el índice que resulta cuando la subcadena cinemática se recorre de principio a fin, empezando el cómputo por cero. Por tanto, el índice de un eje de orientación siempre es igual al índice de la sección anterior.

El índice de un eje de orientación también puede determinarse con la variable de sistema $AC_TRAFO_ORIAX_LOC.

Máxima modificación admisible de un elemento de cadena

La modificación máxima admisible de un elemento de cadena puede limitarse con los dos datos de operador SD41610 $SN_CORR_TRAFO_LIN_MAX para vectores de offset y SD41611 $SN_CORR_TRAFO_DIR_MAX para vectores de dirección de ejes de orientación. SD41610 $SN_CORR_TRAFO_LIN_MAX indica el valor máximo con el que se puede modificar cada componente de vector con respecto a su valor de referencia. SD41611 $SN_CORR_TRAFO_DIR_MAX indica el ángulo máximo con el que se puede modificar la dirección del vector de eje con respecto a su valor de referencia. El valor de referencia es siempre el valor que actúa en la transformada activa cuando se llama CORRTRAFO. Es decir, en caso de que se modificasen los contenidos de los datos cinemáticos en la gestión de datos tras activar la transformada, la función CORRTRAFO no se vería afectada.

3.13 Correcciones de herramientas

3.13.1 Memoria de corrección

Estructura de la memoria de corrección

Cada campo de datos se puede activar con un número T y D y contiene, además de la información geométrica de la herramienta, otros campos como, p. ej., el tipo de herramienta.

Datos de filo del usuario

Mediante los datos de máquina se pueden configurar los datos de filo de usuario. Preste atención a las indicaciones del fabricante de la máquina.

Parámetros de herramienta

Nota

Valores individuales en la memoria de corrección

Los valores individuales de la memoria de corrección (P1 a P25) se pueden leer y escribir por programa a través de variables del sistema. Todos los demás parámetros están reservados.

Los parámetros de herramienta $TC_DP6 a $TC_DP8, $TC_DP10 y $TC_DP11, así como $TC_DP15 a $TC_DP17, $TC_DP19 y $TC_DP20 cambian de significado en función del tipo de herramienta.

Número del parámetro de herramientas (DP)	Significado de las variables de sistema	Observación
$TC_DP1	Tipo de herramienta	Vista de conjunto: ver lista
$TC_DP2	Posición del filo	Sólo para herramientas de torneado
Geometría	**Corrección longitudinal**	
$TC_DP3	Longitud 1	Cálculo según
$TC_DP4	Longitud 2	Tipo y plano
$TC_DP5	Longitud 3	
Geometría	**Radio**	
$TC_DP6 [1)] $TC_DP6 [2)]	Radio 1/longitud 1 diámetro d	Herramienta de fresado/torneado/ rectificado Sierra para ranurar
$TC_DP7 [1)] $TC_DP7 [2)]	Longitud 2/radio de redondeo de fresas cónicas Anchura de la ranura b radio de redondeo	Fresas Sierra para ranurar
$TC_DP8 [1)] $TC_DP8 [2)]	Radio de redondeo 1 para fresas Alero k	Fresas Sierra para ranurar
$TC_DP9 [1) 3)]	Radio de redondeo 2	reservado
$TC_DP10 [1)]	Ángulo 1 superficie frontal de la herramienta	Fresas cónicas
$TC_DP11 [1)]	Ángulo 2 eje longitudinal de la herramienta	Fresas cónicas
Desgaste	**Corrección longitudinal y radial**	

Número del parámetro de herramientas (DP)	Significado de las variables de sistema	Observación
$TC_DP12	Longitud 1	
$TC_DP13	Longitud 2	
$TC_DP14	Longitud 3	
$TC_DP15 [1)] $TC_DP15 [2)]	Radio 1/longitud 1 diámetro d	Herramienta de fresado/torneado/ rectificado Sierra para ranurar
$TC_DP16 [1)] $TC_DP16 [3)]	Longitud 2/radio de redondeo de fresas cónicas Anchura de la ranura b radio de redondeo	Fresas Sierra para ranurar
$TC_DP17 [1)] $TC_DP17 [2)]	Radio de redondeo 1 para fresas Alero k	Fresado/fresado frontal en 3D Sierra para ranurar
$TC_DP18 [1) 3)]	Radio de redondeo 2	reservado
$TC_DP19[1)]	Ángulo 1 superficie frontal de la herramienta	Fresas cónicas
$TC_DP20[1)]	Ángulo 2 eje longitudinal de la herramienta	Fresas cónicas
Acotado básico/Adaptador	**Correcciones longitudinales**	
$TC_DP21	Longitud 1	
$TC_DP22	Longitud 2	
$TC_DP23	Longitud 3	
Tecnología		
$TC_DP24	Ángulo de despulla	sólo para herramientas de torneado
$TC_DP25		reservado

1) Es válido también como fresa para el fresado frontal en 3D

2) En tipo de herramienta Sierra para ranurar

3) Reservado

Observaciones

Para los datos geométricos (p. ej.: longitud 1 o radio) existen varios campos. El tamaño de la herramienta es el resultado aditivo de varios de estos campos (p. ej.: longitud total 1, radio total).

A los valores de corrección que no sean necesarios se les debe dar el valor 0.

Parámetros de herramienta $TC-DP1 a $TC-DP23 con herramientas de contorno

Nota

Los parámetros de herramientas que no están representados en la tabla como, p. ej., $TC_DP7, no se evalúan, es decir, su contenido es irrelevante.

Número del parámetro de herramientas (DP)	Descripción	Filos Dn		Observación
$TC_DP1	Tipo de herramienta			400 a 599
$TC_DP2	Posición del filo			
Geometría	**Corrección longitudinal**			

Número del parámetro de herramientas (DP)	Descripción	Filos Dn		Observación
$TC_DP3	Longitud 1			
$TC_DP4	Longitud 2			
$TC_DP5	Longitud 3			
Geometría	**Radio**			
$TC_DP6	Radio			
Geometría	**Ángulo límite**			
$TC_DP10	Ángulo límite mínimo			
$TC_DP11	Ángulo límite máximo			
Desgaste	**Corrección longitudinal y radial**			
$TC_DP12	Desgaste longitud 1			
$TC_DP13	Desgaste longitud 2			
$TC_DP14	Desgaste longitud 3			
$TC_DP15	Desgaste radio			
Desgaste	**Ángulo límite**			
$TC_DP19	Desgaste ángulo límite mínimo			
$TC_DP20	Desgaste ángulo límite máximo			
Acotado básico/Adaptador	**Correcciones longitudinales**			
$TC_DP21	Longitud 1			
$TC_DP22	Longitud 2			
$TC_DP23	Longitud 3			

Valor básico y valor de desgaste

Las magnitudes resultantes proceden de la correspondiente suma del valor básico más el valor de desgaste (p. ej. $TC_DP6 + $TC_DP15 para el radio). Para la longitud de herramienta de los primeros filos se añade además el acotado básico ($TC_DP21 – $TC_DP23). Adicionalmente, a estas longitudes de herramienta les afectan todas las demás magnitudes que también pueden influir en la longitud efectiva de la herramienta en el caso de una herramienta convencional (adaptador, portaherramientas orientable, datos de operador).

Ángulos límite 1 y 2

Los ángulos límite 1 o 2 se refieren al vector que va desde el centro hasta el punto de referencia del filo y se miden en sentido antihorario.

3.13.2 Correcciones aditivas

3.13.2.1 Seleccionar correcciones aditivas (DL)

Las correcciones aditivas pueden considerarse como correcciones de proceso programables durante la ejecución. Se refieren a los datos geométricos de un filo y forman parte, por lo tanto, de los datos del filo de la herramienta.

Los datos de una corrección aditiva se activan a través de un número DL (DL: location dependent; correcciones relativas al lugar de uso) y se introducen a través de la interfaz de usuario.

Aplicación

Las correcciones aditivas sirven para compensar errores dimensionales debidos a las peculiaridades del lugar de uso.

Sintaxis

```
DL=<Número>
```

Descripción

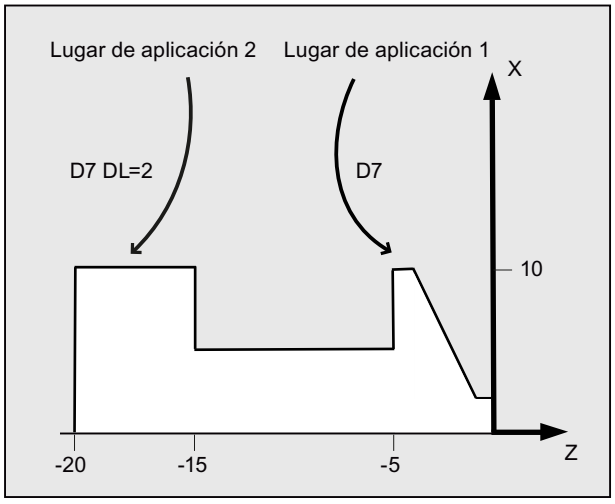
DL:	Comando para activar una corrección aditiva
<Número>:	El parámetro <número> especifica el juego de datos de corrección de herramienta aditivo que se desea activar

Nota

La cantidad y la activación de las correcciones aditivas se definen mediante datos de máquina (→ ¡Observar las indicaciones del constructor de la máquina!).

Ejemplo

Se utiliza el mismo filo para 2 asientos de cojinete:



Código de programa	Comentarios
N110 T7 D7	; La torreta revólver se posiciona en el puesto 7. D7 y DL=1 se activan y se aplican en la secuencia siguiente.
N120 G0 X10 Z1	

Código de programa	Comentarios
N130 G1 Z-6	
N140 G0 DL=2 Z-14	; Junto a D7 se activa DL = 2 y se aplica en la secuencia siguiente.
N150 G1 Z-21	
N160 G0 X200 Z200	; Posicionamiento en el punto de cambio de herramienta.
...	

3.13.2.2 Definir valores de desgaste y de ajuste ($TC_SCPxy[t,d], $TC_ECPxy[t,d])

Los valores de desgaste y de ajuste se pueden leer y escribir mediante variables de sistema. Para ello se orienta la lógica a la de las correspondientes variables de sistema para herramientas y filos.

Variables del sistema

$TC_SCPxy[<t>,<d>]:	Valores de desgaste asignados a los parámetros geométricos correspondientes mediante xy, donde "x" es el número del valor del desgaste e "y" la referencia a los parámetros geométricos.
$TC_ECPxy[<t>,<d>]:	Valores de ajuste que estén asignados a los parámetros geométricos mediante xy, donde "x" es el número del valor del ajuste e "y" la referencia a los parámetros geométricos.
<t>: Número T de la herramienta <d>: Número D del filo de la herramienta	

Nota

Los valores definidos para el desgaste y el ajuste se suman a los parámetros geométricos y al resto de los parámetros de corrección (números D).

Ejemplo

El valor de desgaste de la longitud 1 se fija a 1.0 para el filo <d> de la herramienta <t>.

Parámetros: $TC_DP**3** (Longitud 1, en herramientas de tornear)

Valores de desgaste: $TC_SCP1**3** hasta $TC_SCP6**3**

Valores de ajuste: $TC_ECP1**3** hasta $TC_ECP6**3**

$TC_SCP4**3** [<t>,<d>] = 1.0

3.13.2.3 Borrar correcciones aditivas (DELDL)

Con el comando DELDL se borran las correcciones aditivas del filo de una herramienta (libera memoria). Para ello se borran tanto los valores definidos para el desgaste como los de ajuste.

Sintaxis

```
DELDL[<t>,<d>]
DELDL[<t>]
DELDL
<Estado>=DELDL[<t>,<d>]
```

Descripción

DELDL:	Comando para borrar correcciones aditivas	
<t>:	Número T de la herramienta	
<d>:	Número D del filo de la herramienta	
DELDL[<t>,<d>]:	Se borran todas las correcciones aditivas del filo <d> de la herramienta <t>.	
DELDL[<t>]:	Se borran todas las correcciones aditivas de todos los filos de la herramienta <t>.	
DELDL:	Se borran todas las correcciones aditivas de los filos de todas las herramientas de la unidad TO (para el canal en el que se ha programado el comando).	
<Estado>:	Estado de borrado	
	Valor:	Significado:
	0	El borrado se ha ejecutado con éxito.
	-	No se ha ejecutado el borrado (si la parametrización designa exactamente un filo), o el borrado no se ha ejecutado por completo (si la parametrización designa varios filos).

Nota

Los valores de ajuste y de desgaste de las herramientas activas no se pueden borrar (análogamente al comportamiento de borrado de correcciones D o de datos de herramienta).

3.13.3 Acciones especiales de corrección de herramienta

Los datos de operador DO42900 a DO42960 permiten controlar la evaluación del signo de la longitud de herramienta y el desgaste.

Ello también es válido para el comportamiento de las componentes del desgaste al simetrizar ejes geométricos o al cambiar el plano de trabajo y para la compensación de la temperatura en la dirección de la herramienta.

Valores de desgaste

Cuando a partir de ahora se haga referencia a valores de desgaste, se entenderá respectivamente la suma de los valores de desgaste propiamente ($TC_DP12 a $TC_DP20) y las correcciones de suma con los valores de desgaste ($SCPX3 a $SCPX11) y de ajuste ($ECPX3 a $ECPX11).

Encontrará **información** sobre las correcciones de suma en el Manual de funciones Gestión de herramientas.

Datos del operador

DO42900 $SC_MIRROR_TOOL_LENGTH	Simetrizar componentes longitudinales de herramientas y componentes del acotado básico.
SD42910 $SC_MIRROR_TOOL_WEAR	Simetrizar los valores de desgaste de las componentes longitudinales de herramienta.
SD42920 $SC_WEAR_SIGN_CUTPOS	Valoración del signo de las componentes de desgaste dependiendo de la posición del filo.
DO42930 $SC_WEAR_SIGN	Invierte el signo de las dimensiones de desgaste.
DO42935 $SC_WEAR_TRANSFORM	Transformación de los valores de desgaste.
DO42940 $SC_TOOL_LENGTH_CONST	Asignación de las componentes longitudinales de herramienta a los ejes geométricos.
SD42950 $SC_TOOL_LENGTH_TYPE	Asignación de las componentes longitudinales de herramienta independientemente del tipo de herramienta.
DO42960 $SC_TOOL_TEMP_COMP	Valor de compensación de temperatura en la dirección de la herramienta. También está activo si existe una orientación de herramienta

Información adicional

Validez de los datos de operador modificados

La nueva valoración de los datos de operador tras una modificación de los componentes de la herramienta se realiza la próxima vez que se seleccione el filo de la herramienta. Si la herramienta para la que se desean modificar los datos esta siendo activada y se pretende validar la modificación, entonces se debe volver a seleccionar la herramienta.

Ello también es válido para el caso en que se modifique la longitud resultante de la herramienta porque se haya cambiado el estado de simetría de un eje. La herramienta se debe volver a seleccionar tras el comando de simetrización; de esta manera, serán válidos las componentes longitudinales modificadas de la herramienta.

Portaherramientas orientable y nuevos datos de operador

Los datos de operador DO42900 a DO42940 no afectan a las componentes de un hipotético portaherramientas orientable activo. Para el cálculo con portaherramientas orientable siempre se tiene en cuenta la herramienta con su longitud total (longitud de herramienta + desgaste + dimensión base). En el cálculo de la longitud total resultante se tienen en cuenta todas las modificaciones generadas mediante datos de operador; es decir, los vectores del portaherramientas orientable son independientes del plano de trabajo.

Nota

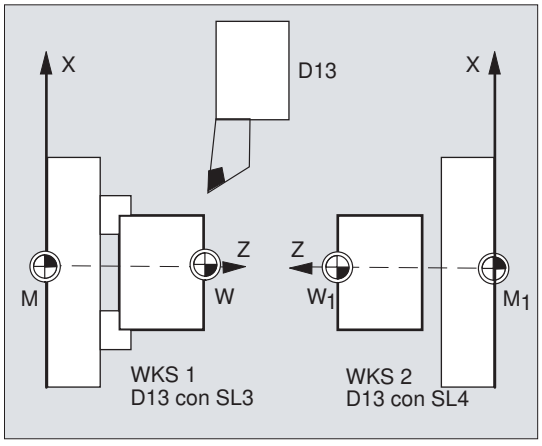
Al utilizar un portaherramientas orientable es frecuente definir todas las herramientas para un sistema base no simetrizado, incluso aquellas que sólo se vayan a utilizar para el mecanizado con simetría especular. Para el mecanizado con ejes simetrizados se gira el portaherramientas de tal manera que se describa correctamente la posición de la herramienta. Todos los componentes longitudinales de herramienta actúan automáticamente en la dirección correcta, de modo que no es necesario controlar la valoración de los componentes individuales mediante datos de operador dependiendo del estado de la simetría especular de cada eje.

Otras posibilidades de aplicación

Puede tener sentido utilizar la funcionalidad del portaherramientas orientable cuando no sea posible girar la herramienta físicamente en la máquina, aunque se pueda fijar la herramienta con diferentes orientaciones fijas. El acotado de la herramienta se puede realizar de forma unitaria tomando una orientación base y obtener las dimensiones relevantes para el mecanizado mediante el giro de un portaherramientas virtual.

3.13.3.1 Simetrizar longitudes de herramienta

Con datos de operador DO42900 $SC_MIRROR_TOOL_LENGTH y DO42910 $SC_MIRROR_TOOL_WEAR distintos a cero, las componentes de longitud de herramientas y las componentes del acotado básico se pueden simetrizar con valores de desgaste de los correspondientes ejes.



DO42900 $SC_MIRROR_TOOL_LENGTH

Dato de operador **distinto** de cero:

Invirtiendo el signo se simetrizan los componentes longitudinales de la herramienta ($TC_DP3, $TC_DP4 y $TC_DP5) y los componentes de las dimensiones base ($TC_DP21, $TC_DP22 y $TC_DP23) cuyos ejes han sido simetrizados.

Los valores de desgaste **no** se simetrizan. Si se tienen que simetrizar también estos valores, se debe activar el SD42910 $SC_MIRROR_TOOL_WEAR.

DO42910 $SC_MIRROR_TOOL_WEAR

Dato de operador **distinto** de cero:

Invirtiendo el signo se simetrizan los valores de desgaste de las componentes de longitud de la herramienta cuyos ejes han sido simetrizados.

3.13.3.2 Evaluación de signos de desgaste

Con los datos de operador DO42920 $SC_WEAR_SIGN_CUTPOS y DO42930 $SC_WEAR_SIGN distintos de cero, se puede invertir la evaluación de signos de las componentes de desgaste.

DO42920 $SC_WEAR_SIGN_CUTPOS

Dato de operador **distinto** de cero:

En herramientas con posición orientada del filo (herramientas de torneado y rectificado, tipos de herramientas 400), la valoración del signo de las componentes del desgaste en el plano de mecanizado depende de la posición del filo. Para herramientas sin posición fija para el corte, este DO carece de sentido.

En la siguiente tabla se marcan con una X las dimensiones cuyos signos se invierten mediante el DO42920 (distinto de 0):

Posición del filo	Longitud 1	Longitud 2
1		
2		X
3	X	X
4	X	
5		
6		
7		X
8	X	
9		

Nota

Las valoraciones del signo con los DO42920 y DO42910 son independientes entre sí. Si el signo de una medida se modifica mediante los dos datos de operador, el resultado es que no se altera el signo de la medida.

DO42930 $SC_WEAR_SIGN

Dato de operador **distinto** de cero:

Se invierte el signo de todas las medidas de desgaste. Actúa tanto para la longitud de la herramienta como para el resto de dimensiones como el radio de la herramienta, radio de redondeo, etc.

Si se introduce un valor de desgaste positivo, entonces la herramienta se hace mas "corta" y más "fina", ver el apartado "Corrección de herramienta -Tratamiento especial", Validez de los datos de operador modificados".

3.13.3.3 Sistema de coordenadas del mecanizado activo (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)

En función de la cinemática de la máquina o de la existencia de un portaherramientas orientable, los valores de desgaste medidos en uno de estos sistemas de coordenadas se trasladan a un sistema de coordenadas apropiado o se transforman.

Sistemas de coordenadas del mecanizado activo

De los siguientes sistemas de coordenadas pueden derivarse offsets de la longitud de herramienta que incluyan el componente longitudinal de herramienta "Desgaste" en una herramienta activa por medio del comando G correspondiente del grupo 56.

- Sistema de coordenadas de máquina (MKS)
- Sistema de coordenadas básico (BKS)
- Sistema de coordenadas de pieza (WKS)
- Sistema de coordenadas de herramienta (TCS)
- Sistema de coordenadas de herramienta de la transformación cinemática (KCS)

Sintaxis

```
TOWSTD
TOWMCS
TOWWCS
TOWBCS
TOWTCS
TOWKCS
```

Descripción

`TOWSTD`:	Valor de posición preferencial para correcciones en la longitud de la herramienta Valor de desgaste
`TOWMCS`:	Correcciones en la longitud de la herramienta en el MKS
`TOWWCS`:	Correcciones en la longitud de la herramienta en el WKS
`TOWBCS`:	Correcciones en la longitud de la herramienta en el BKS
`TOWTCS`:	Correcciones de la longitud de la herramienta en el punto de referencia del portaherramientas (portaherramientas orientable)
`TOWKCS`:	Correcciones de la longitud de herramienta del cabezal de herramienta (transformada cinemática)

Información adicional

Características distintivas

En la siguiente tabla se representan las principales características de distinción:

Comando G	Valor de desgaste	Portaherramientas orientable activo
TOWSTD	Valor preferencial, longitud de la herramienta	Los valores de desgaste están sujetos al giro.
TOWMCS	Valor de desgaste en el MKS. TOWMCS es idéntico a TOWSTD si no está activo ningún portaherramientas orientable.	Solo gira el vector de la longitud de herramienta resultante, sin tener en cuenta el desgaste.
TOWWCS	El valor de desgaste se convierte en el WKS al MKS.	El vector de herramienta se calcula como en TOWMCS, sin tener en cuenta el desgaste.

TOWBCS	El valor de desgaste se convierte en el BKS al MKS.	El vector de herramienta se calcula como en TOWMCS, sin tener en cuenta el desgaste.
TOWTCS	El valor de desgaste se convierte en el sistema de coordenadas de herramienta al MKS.	El vector de herramienta se calcula como en TOWMCS, sin tener en cuenta el desgaste.

TOWWCS, TOWBCS, TOWTCS: El vector de desgaste se suma al vector de herramienta.

Transformación lineal

La longitud de herramienta en el MKS sólo se puede definir de forma racional si el MKS resulta del BKS mediante una transformación lineal.

Transformación no lineal

Si, p. ej., con TRANSMIT, está activa una transformada no lineal, se utiliza automáticamente el BKS si se indica el MKS como sistema de coordenadas deseado.

Sin transformación cinemática ni portaherramientas orientable

Si no está activa una transformación cinemática ni un portaherramientas orientable, coinciden los cuatro sistemas de coordenadas, con excepción de WKS. De este modo, sólo el sistema de coordenadas de pieza se distingue de los demás. Dado que sólo se necesitan evaluar longitudes de herramienta, las traslaciones entre los sistemas de coordenadas carecen de importancia.

Inclusión de los valores de desgaste

El dato de operador **DO42935 $SC_WEAR_TRANSFORM** establece cuál de los tres componentes de desgaste

- Desgaste
- Correcciones de sumas finas
- Correcciones de suma gruesas

será sometido a giro por una transformada de adaptador o un portaherramientas orientable cuando esté activo uno de los siguientes comandos G:

- TOWSTD
  Ajuste inicial. Para correcciones de la longitud de herramienta.
- TOWMCS
  Valores de desgaste en el sistema de coordenadas de máquina (MKS).
- TOWWCS
  Valores de desgaste en el sistema de coordenadas de pieza (WKS).
- TOWBCS
  Valores de desgaste en el sistema de coordenadas básico (BKS).

- TOWTCS
  Valores de desgaste en el sistema de coordenadas de herramienta en el alojamiento del portaherramientas (referencia de portaherramientas T).
- TOWKCS
  Valores de desgaste en el sistema de coordenadas del cabezal de herramienta con transformada cinética.

Nota

En la evaluación de los distintos componentes de desgaste (asignación a los ejes geométricos, evaluación de signos) influyen los factores siguientes:

- Nivel activo
- Transformada de adaptador
- Datos de operador:
  - SD42910 $SC_MIRROR_TOOL_WEAR
  - SD42920 $SC_WEAR_SIGN_CUTPOS
  - DO42930 $SC_WEAR_SIGN
  - DO42940 $SC_TOOL_LENGTH_CONST
  - SD42950 $SC_TOOL_LENGTH_TYPE

3.13.3.4 Longitud de herramienta y cambio de plano

Con los datos de operador DO42940 $SC_TOOL_LENGTH_CONST distintos de cero, las componentes de longitud de herramienta como longitud, desgaste y acotado básico, se pueden asignar a los ejes geométricos para herramientas de tornear y rectificar en un cambio de plano.

DO42940 $SC_TOOL_LENGTH_CONST

Dato de operador **distinto** de cero:

No se modifica la asignación de las componentes de longitud de la herramienta (longitud, desgaste y acotado básico) a los ejes geométricos (`G17` - `G19`) al cambiar el plano de trabajo.

La siguiente tabla muestra la asignación de los componentes longitudinales a los ejes geométricos para herramientas de torneado y rectificado (herramientas del tipo 400 a 599):

Contenido	Longitud 1	Longitud 2	Longitud 3
17	Y	X	Z
*)	X	Z	Y
19	Z	Y	X
-17	X	Y	Z
-18	Z	X	Y
-19	Y	Z	X

*) Todos los valores distintos de 0 que no sean iguales a alguno de los seis valores presentados se evalúan como el valor 18.

La siguiente tabla muestra la asignación de los componentes longitudinales a los ejes geométricos para el resto de herramientas (hta. del tipo < 400 ó> 599):

Plano de mecanizado	Longitud 1	Longitud 2	Longitud 3
*)	Z	Y	X
18	Y	X	Z
19	X	Z	Y
-17	Z	X	Y
-18	Y	Z	X
-19	X	Y	Z

*) Todos los valores distintos de 0 que no sean iguales a alguno de los seis valores presentados se evalúan como el valor 17.

Nota

En la presentación de las tablas, se parte de la premisa de que los ejes geométricos 1 a 3 están identificados con los identificadores X, Y, Z. La asignación de una corrección a un eje viene especificada por el número secuencial del eje y no por el identificador del eje.

3.13.4 Corrección de herramienta online

3.13.4.1 Definición de la función polinómica (FCTDEF)

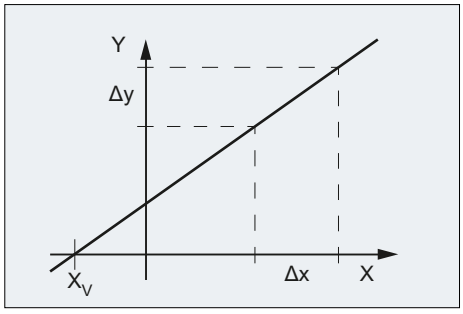
Existen determinadas estrategias de diamantado (p. ej., cilindro de diamantado) que se caracterizan por la reducción continua (lineal) de la muela con la penetración del cilindro de diamantado en el radio. Para ello se necesita una función lineal entre la penetración del cilindro de diamantado y la escritura del valor de desgaste de la longitud correspondiente. La definición de la función lineal se realiza mediante el procedimiento predefinido FCTDEF(...) para funciones polinómicas de hasta tercer grado.

Ecuación lineal

$y = f(x) = a_0 + a_1 * x_1$

a_1: Pendiente de la recta, con $a_1 = \Delta x / \Delta y$

a_0: Decalaje de la recta a lo largo del eje X, con $a_0 = -a1 * X_V$



Sintaxis

```
FCTDEF(<Func>,<LLimit>,<ULimit>,<a0>,<a1>,<a2>,<a3>)
```

Descripción

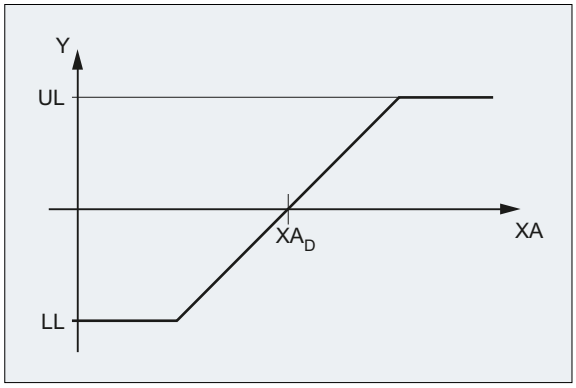
`FCTDEF(...)`:	Definición de una función polinómica para PUTFTOCF(...): $y = f(x) = a_0 + a_1*x + a_2*x^2 + a_3*x^3$	
<`Func`**>**:	Número de función	
	Tipo de dato:	INT
	Rango de valores:	1, 2, 3
`<LLimit>`:	Valor de limitación inferior	
	Tipo de dato:	REAL
`<ULimit>`:	Valor de limitación superior	
	Tipo de dato:	REAL
`<a0>,<a1>,<a2>,<a3>`:	Coeficientes de la función polinómica	
	Tipo de dato:	REAL

Ejemplo

Definiciones

- Número de función: 1
- Valores de limitación superior e inferior: -100, 100
- Pendiente de la curva característica: $a_1 = 1$
- El punto de operación debe encontrarse en el centro de la curva característica. Para ello, la curva característica debe desplazarse en dirección Y negativa mediante la posición de consigna del eje XA en WKS en el momento de la definición de funciones en el programa CN: $a_0 = -a_1 * XA_D = -1 * \$AA_IW$
- $a_2 = a_3 = 0$

Curva característica



UL Valor de limitación superior

LL Valor de limitación inferior

XA_D Consigna del eje XA en el momento de la definición de funciones en el programa CN

Programación

Código de programa	**Comentarios**
`FCTDEF(1,-100,100,-$AA_IW[XA],1)`	`; Definición de funciones`

3.13.4.2 Escritura continua de corrección de herramienta online (PUTFTOCF)

Con el procedimiento predefinido PUTFTOCF(...) se realiza una corrección de herramienta online mediante una función polinómica definida con FCTDEF(...) (Página 772).

Nota

La corrección de herramienta online puede realizarse también a través de una acción síncrona.

Para más información, ver el manual de funciones Acciones síncronas.

Sintaxis

```
PUTFTOCF(<Func>,<RefVal>,<ToolPar>,<Chan>,<Sp>)
```

Descripción

`PUTFTOCF(...)`:	Escritura de la corrección de herramienta online de forma continua por secuencias mediante la función polinómica definida con FCTDEF(...)	
`<Func>`:	Número de función determinado en la definición de funciones con FCTDEF(...)	
	Tipo de dato:	INT
	Rango de valores:	1, 2, 3
`<RefVal>`:	Valor de referencia del que debe deducirse la corrección (p. ej., consigna de un eje).	
	Tipo de dato:	VAR REAL

<ToolPar>:	Número del parámetro de desgaste (longitud 1, 2 ó 3) en el que debe calcularse el valor de corrección.	
	Tipo de dato:	INT
<Chan>:	Número del canal en el que debe activarse la corrección de herramienta online. **Nota**: La indicación solo es necesaria si la corrección no debe activarse en el canal activo.	
	Tipo de dato:	INT
<Sp>:	Número del cabezal para el que debe activarse la corrección de herramienta online. **Nota:** La indicación solo es necesaria si, en lugar de la herramienta que se encuentra activa y en uso, se debe corregir una muela rectificadora no activa.	
	Tipo de dato:	INT

3.13.4.3 Escritura discreta de corrección de herramienta online (PUTFTOC)

Función

Con el procedimiento predefinido PUTFTOC(...) se realiza una corrección de herramienta online con un valor de corrección fijo.

Sintaxis

```
PUTFTOC(<CorrVal>,<ToolPar>,<Chan>,<Sp>)
```

Descripción

PUTFTOC(...):	Escritura de la corrección de herramienta online	
<CorrVal>:	Valor de corrección que se suma al parámetro de desgaste	
	Tipo de dato:	REAL
<ToolPar>:	Número del parámetro de desgaste (longitud 1, 2 ó 3) en el que debe calcularse el valor de corrección.	
	Tipo de dato:	INT
<Chan>:	Número del canal en el que debe activarse la corrección de herramienta online. **Nota:** La indicación solo es necesaria si la corrección no debe activarse en el canal activo.	
	Tipo de dato:	INT
<Sp>:	Número del cabezal para el que debe activarse la corrección de herramienta online. **Nota:** La indicación solo es necesaria si, en lugar de la herramienta que se encuentra activa y en uso, se debe corregir una muela rectificadora no activa.	
	Tipo de dato:	INT

3.13.4.4 Activación/desactivación de la corrección de herramienta online (FTOCON/FTOCOF)

Con los comandos G FTOCON y FTOCOF se activa/desactiva la corrección de herramienta online.

Sintaxis

```
FTOCON
...
FTOCOF
```

Descripción

FTOCON:	Activación de la corrección de herramienta online El comando debe programarse en el canal en el que se desea activar la corrección de herramienta online.
FTOCOF:	Desactivación de la corrección de herramienta online El comando debe programarse en el canal en el que se desea desactivar la corrección de herramienta online. **Nota:** Con FTOCOF, la corrección de herramienta no sigue retirándose. Sin embargo, en los datos de corrección específicos del filo se conserva el valor calculado con PUTFTOC/PUTFTOCF. Para desactivar definitivamente la corrección de herramienta online, tras FTOCOF debe realizarse otra selección/cancelación de la herramienta (T...).

3.13.5 Corrección de radio de herramienta 3D

3.13.5.1 Seleccionar la corrección de radio de herramienta 3D para fresado periférico 3D (CUT3DC, CUT3DCD, ISD)

La corrección de radio de herramienta 3D (3D-WRK) para el fresado periférico 3D sin tener en cuenta las superficies de limitación se selecciona con el comando de efecto modal G CUT3DC o CUT3DCD.

La activación propiamente dicha se efectúa con G41 o G42 La corrección de radio de herramienta se desactiva con G40.

Sintaxis

```
G41/G42 ORIC/ORID ISD=... CUT3DC/CUT3DCD CDOF2 X... Y... Z...
...
G40 X... Y... Z...
```

Descripción

CUT3DC:	3D-WRK para el fresado periférico (solo si está activa la transformada de 5 ejes)	
CUT3DCD:	3D-WRK relativa a una herramienta diferencial para el fresado periférico (solo si está activa la transformada de 5 ejes) La diferencia de radio se determina a través del parámetro de herramienta $TC_DP15.	
G41/G42 X... Y... Z...:	Activar la corrección de radio de herramienta	
	G41:	Corrección del radio de la herramienta a la izquierda del contorno
	G42:	Corrección del radio de la herramienta a la derecha del contorno
	Nota: La activación debe efectuarse en una secuencia lineal (G0/G1).	
CDOF2:	Desactivar vigilancia de colisión para fresado periférico 3D	
ORIC/ORID:	El comportamiento en caso de cambio de la orientación en esquinas exteriores se determina con los comandos G ORIC y ORID.	
	ORIC:	Los cambios de orientación en esquinas exteriores se superponen a la secuencia circular a insertar.
	ORID:	Los cambios de orientación en esquinas exteriores se ejecutan antes de la secuencia circular que se desea insertar.
ISD=<Value>:	La dirección ISD permite modificar la profundidad de penetración de la herramienta en el fresado periférico con la corrección de radio de herramienta 3D **activa**.	
	<Value>:	Longitud de la profundidad de penetración
G40 X... Y... Z...:	Desactivación de la corrección del radio de corte/herramienta **Nota:** La desactivación debe efectuarse en una secuencia lineal (G0/G1) con desplazamientos de ejes geométricos.	

Nota

Los comandos G para la selección de la corrección de radio de herramienta 3D se evalúan en la secuencia de aproximación, es decir, normalmente en la secuencia que contiene G41 o G42.

G41 o G42 también pueden estar programados en secuencias sin desplazamiento en los ejes geométricos relevantes para la corrección. En este caso, la secuencia de aproximación es la primera secuencia de desplazamiento que sigue a una secuencia de este tipo.

Cuando la corrección de radio de herramienta está activa, se ignora sin generar alarma cualquier cambio de la variante de corrección de radio de herramienta 3D.

Ejemplo

Código de programa	Comentarios
	; Definición de la herramienta D1:
$TC_DP1[1,1]=120	; Tipo (fresa de mango)

Código de programa	Comentarios
`$TC_DP3[1,1]=20`	`; Vector de corrección longitudinal`
`$TC_DP6[1,1]=8`	`; Radio`
`N10 X0 Y0 Z0 T1 D1 F12000`	`; Selección de la herramienta.`
`N20 TRAORI(1)`	`; Activación de la transformada.`
`N30 G42 ORIC ISD=10 CUT3DC G64 X30`	`; Activación del fresado periférico 3D,`
	`; Cambios de orientación continuos en esquinas exteriores,`
	`; Profundidad de penetración: 10 mm`
`N40 ORIWKS A30 B15`	`; Cambio de orientación en una esquina indicando las posiciones de eje.`
`N50 Y20 A3=1 C3=1`	`; Secuencia de desplazamiento con cambio de orientación,`
	`; Indicación de la orientación con vector de dirección.`
`N60 X50 Y30`	`; Secuencia de desplazamiento con orientación constante.`
`N70 Y50 A3=0.5 B3=1 C3=5`	`; Secuencia de desplazamiento con cambio de orientación.`
`N80 M63`	`; Secuencia sin información de desplazamiento.`
`N90 X0 ISD=20`	`; Secuencia de desplazamiento con cambio de profundidad de penetración.`
`N100 G40 Y0`	`; Desactivación de la corrección de radio de herramienta.`
`N110 M30`	

Información adicional

Trayectoria y orientación

La variante del fresado periférico aquí utilizada se realiza definiendo una trayectoria y la orientación de la misma. En este tipo de mecanizado la forma de la herramienta carece de significado para la trayectoria. Sólo cuenta el radio en el punto de contacto de la herramienta.

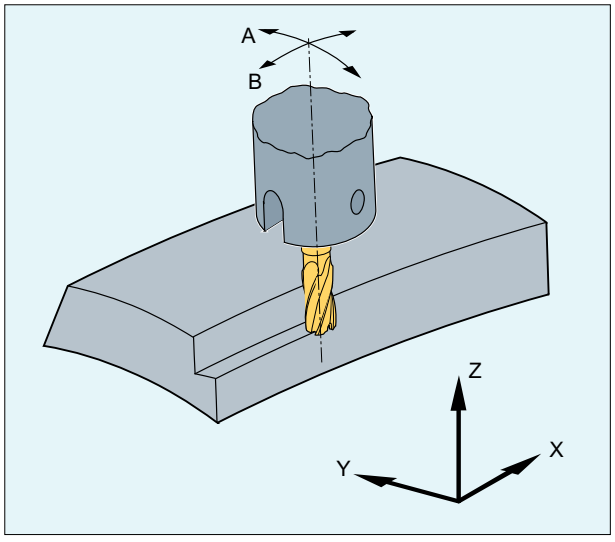


Figura 3-8 Fresado periférico

Método de aproximación

El método de aproximación es siempre NORM con las variantes 3D de la corrección de radio de herramienta.

Comportamiento en esquinas exteriores

En las esquinas exteriores se evalúan los comandos G del grupo 18 (Comportamiento en esquinas, corrección de herramienta) durante el fresado periférico con 3D-WRK de forma análoga al comportamientos durante 2½-D-WRK:

- G450: Circunferencia de transición (la herramienta pasa por las esquinas de la pieza siguiendo una trayectoria circular)
  Al contrario de la solución en 2½-D-WRK, el elemento de contorno insertado en una esquina exterior es siempre un arco con radio 0 al que la corrección de radio de herramienta afecta como a cualquier otra trayectoria programada. No es posible insertar una sección cónica en lugar del arco. Por eso, la dirección DISC carece de significado en este caso y no se evalúa.
- G451: Punto de intersección de las equidistantes (la herramienta sale en el ángulo de la pieza)
  El punto de intersección se determina, prolongando las curvas offset de las dos secuencias afectadas y determinando su punto de intersección en el plano vertical respecto a la orientación de herramienta en la esquina.

El método de punto de intersección (G451) no se utiliza si entre las secuencias de desplazamiento implicadas está insertada al menos una secuencia que contiene un cambio de orientación de herramienta. En tal caso, siempre se inserta un arco en la esquina.

Comportamiento con cambios de orientación en esquinas exteriores

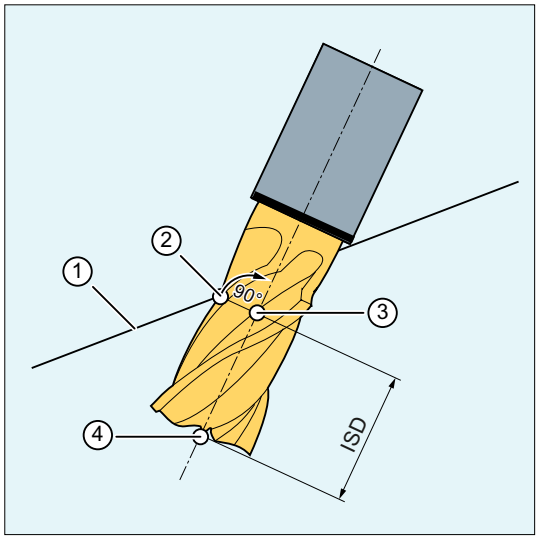
Con los comandos G ORIC y ORID se puede definir si los cambios de orientación programados entre las dos secuencias de desplazamiento que forman la esquina se ejecutan antes del inicio de la secuencia circular insertada (ORID) o al mismo tiempo que esta (ORIC).

Profundidad de penetración

La profundidad de penetración de la fresa es la distancia del punto auxiliar de la fresa a la punta de la herramienta.

El punto auxiliar de la fresa es la proyección vertical del punto de mecanizado de la fresa en la trayectoria programada sobre el eje longitudinal de la herramienta.

Mediante la profundidad de penetración se ajusta por tanto la posición del punto de mecanizado en la envolvente de la herramienta.



① Trayectoria programada
② Punto de mecanizado de la fresa
③ Punto auxiliar de la fresa
④ Punta de la fresa
ISD Profundidad de penetración (InSertion Depth)

Figura 3-9 Profundidad de penetración

Corrección del radio de herramienta relativa a una herramienta diferencial

La corrección de radio de herramienta 3D para el fresado periférico relativa a una herramienta diferencial se activa con el comando CUT3DCD. Debe utilizarse cuando el contorno programado se refiere a la trayectoria del centro de una herramienta normalizada y el mecanizado se realiza con otra herramienta distinta. En tal caso, en el cálculo de la corrección del radio de herramienta en 3D se incluirán solo el valor de desgaste del radio de la herramienta activa ($TC_DP15) y el offset de corrección de herramienta OFFN y TOFFR/TOFFLR, en caso de haberse programado. El radio base ($TC_DP6) de la herramienta activa **no** se incluye en el cálculo.

Fresado de cajas con paredes laterales inclinadas para el fresado periférico con CUT3DC

En esta corrección de radio de herramienta 3D se compensa una desviación del radio de la fresa, penetrando en dirección de la normal de superficie de la superficie mecanizada. Sin embargo, el plano en el que se halla el lado frontal de la fresa no se modifica si la profundidad de penetración ISD no cambia. En tal caso, una fresa que, p. ej., tuviera un radio más pequeño que una herramienta normalizada, no alcanzaría el fondo de la caja, que constituye también la superficie de limitación. Para la penetración automática de la herramienta, el control debe conocer esta superficie de limitación, ver apartado "Fresado periférico 3D considerando una superficie de limitación (CUT3DCC, CUT3DCCD) (Página 787)".

Advanced Surface/Top Surface

Nota

Si se utiliza la corrección de radio de herramienta CUT3DCD en combinación con la función sujeta a licencia "Advanced Surface" o "Top Surface", deben tenerse en cuenta las recomendaciones de ajuste para "Advanced Surface"/"Top Surface".

En el SIOS Portal hay un programa verificador especial que puede utilizarse para comprobar los datos ajustados.

→ Programa verificador para Advanced Surface/Top Surface (https://support.industry.siemens.com/cs/ww/es/view/109738423/en)

3.13.5.2 Seleccionar la corrección de radio de herramienta 3D para el fresado frontal 3D (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD)

La corrección de radio de herramienta 3D (3D-WRK) para el fresado frontal 3D se selecciona con el comando de efecto modal CUT3DF, CUT3DFS, CUT3DFF o CUT3DFD.

La activación propiamente dicha se efectúa con G41 o G42

Para calcular la corrección de radio de herramienta, en el fresado frontal 3D es necesario definir la normal a la superficie del plano que se va a mecanizar. Esto debe hacerse en la secuencia con G41 o G42 a través de las direcciones A4, B4, C4 y A5, B5, C5.

La corrección de radio de herramienta se desactiva con G40.

Sintaxis

```
G41/G42 ORIC/ORID CUT3DF/CUT3DFS/CUT3DFF/CUT3DFD X... Y... Z... A4=... B4=...
C4=... A5=... B5=... C5=...
...
G40 X... Y... Z...
```

Descripción

CUT3DFS:	3D-WRK para fresado frontal con orientación constante. La orientación de herramienta se define con G17 - G19 y no depende de los frames.
CUT3DFF:	3D-WRK para fresado frontal con orientación constante. La orientación de la herramienta es la dirección definida con G17 - G19 y, dado el caso, girada por un frame.
CUT3DF:	3D-WRK para fresado frontal con cambio de orientación (solo con transformada de 5 ejes activa).

CUT3DFD:	3D-WRK relativa a una herramienta diferencial para fresado frontal con cambio de orientación (solo con transformada de 5 ejes activa). La diferencia de radio se determina a través del parámetro de herramienta $TC_DP15. **Nota**: CUT3DFD solo es posible en combinación con el "Alisado de las normales a la superficie con fresado frontal 3D". Este se activa llamando la función sujeta a licencia "Top Surface" mediante CYCLE832(...).	
G41/G42 X... Y... Z...:	Activar la corrección de herramienta El comportamiento con G41 y con G42 es el mismo con el fresado frontal 3D. **Nota:** La activación debe efectuarse en una secuencia lineal (G0/G1).	
A4/5=... B4/5=... C4/5=...:	Definición de las normales a la superficie del plano que se va a mecanizar	
	A4=... B4=... C4=...:	Definición al inicio de secuencia
	A5=... B5=... C5=...:	Definición al final de secuencia
ORIC/ORID:	El comportamiento en caso de cambio de la orientación en esquinas exteriores se determina con los comandos G ORIC y ORID.	
	ORIC:	Los cambios de orientación en esquinas exteriores se superponen a la secuencia circular a insertar.
	ORID:	Los cambios de orientación se ejecutan antes de la secuencia circular.
G40 X... Y... Z...:	Desactivación de la corrección del radio de corte/herramienta **Nota:** La desactivación debe efectuarse en una secuencia lineal (G0/G1) con desplazamientos de ejes geométricos.	

Nota

G41 o G42 también pueden estar programados en secuencias sin desplazamiento en los ejes geométricos relevantes para la corrección. En este caso, la secuencia de aproximación es la primera secuencia de desplazamiento que sigue a una secuencia de este tipo.

Cuando la corrección de radio de herramienta está activa, se ignora sin generar alarma cualquier cambio de la variante de corrección de radio de herramienta 3D.

Ejemplos

Ejemplo 1: Fresado frontal 3D con CUT3DF

Código de programa	Comentarios
N10	; Definición de la herramienta D1:
N20 $TC_DP1[1,1]=121	; Tipo de herramienta (fresa tórica)

Código de programa	**Comentarios**
N30 $TC_DP3[1,1]=20	; Corrección longitudinal
N40 $TC_DP6[1,1]=5	; Radio
N50 $TC_DP7[1,1]=3	; Radio de redondeo
N60	
N70	
N80 X0 Y0 Z0 A0 B0 C0 G17 T1 D1 F12000	; Selección de la herramienta.
N90 TRAORI(1)	; Seleccionar transformada de orientación.
N100 B4=-1 C4=1	; Definición del plano.
N110 **G41** ORID **CUT3DF** G64 X10 Y0 Z0	; Activar la corrección de herramienta.
N120 X30	
N130 Y20 A4=1 C4=1	; Esquina exterior, nueva definición del plano.
N140 B3=1 C3=5	; Cambio de orientación con ORID.
N150 B3=1 C3=1	; Cambio de orientación con ORID.
N160 X-10 A5=1 C5=2 ORIC	
N170 A3=-2 C3=1	; Cambio de orientación con ORIC
N180 A3=-1 C3=1	; Cambio de orientación con ORIC
N190 Y-10 A4=-1 C4=3	; Nueva definición del plano.
N200 X-20 Y-20 Z10	; Esquina interior con la secuencia anterior.
N210 X-30 Y10 A4=1 C4=1	; Esquina interior, nueva definición del plano.
N220 A3=1 B3=0.5 C3=1.7	; Cambio de orientación con ORIC.
N230 X-20 Y30 A4=1 B4=-2 C4=3 ORID	
N240 A3 = 0.5 B3=-0.5 C3=1	; Cambio de orientación.
N250 X0 Y30 C4=1	; Desplazamiento de trayectoria, nuevo plano, ; Orientación con programación relativa.
N260 BSPLINE X20 Z15	; Inicio de spline, programación relativa de la orientación
N270 X30 Y25 Z18	; Permanece activo durante el 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, orientación con programación de eje.
N320 G1 X0 A3=-0.123 B3=0.456 C3=2.789 B4=-1 C4=5 B5=-1 C5=2	; Desplazamiento de trayectoria, orientación, plano no constante.
N330 X-20 G40	; Desactivar corrección del radio de herramienta.
N340 M30	

Ejemplo 2: Programa NC (sección) generado a partir de un sistema CAD con CUT3DFD

Código de programa	**Comentarios**
N01 G710	
N03 T="12"	
N06 S5305 M03	
N07 G642	
; Desplazamiento a la posición inicial en el MKS teniendo en cuenta la longitud de herramienta.	

Código de programa	**Comentarios**
G00 G90 X-250.62787 Y-38.37944 A=DC(253.12719) B-12.49543	
G00 G90 Z251.80052	
; Fin del posicionamiento en el MKS.	
;	
TRAORI(1)	; Seleccionar transformada de orientación.
G500	
D1	
CYCLE832(0.01, _TOP_SURFACE_SMOOTH_ON + _ORI_FINISH, 1)	; Llamada CYCLE832 con: ; Tolerancia de contorno = 0,01 mm ; Tipo de mecanizado: Top Surface con alisado, ; Acabado con introducción de una tolerancia de orientación, ; Tolerancia de orientación = 1 grado
CUT3DFD	
N08 G90 G94	
N09 G00 X-269.21195 Y128.32027 Z1.18577 A3=-.216361688 B3=.934284397 C3=-.283373051	; Las secuencias de la N09 a la N10 son el componente rápido del desplazamiento de aproximación a la pieza con orientación constante.
N10 G00 X-251.90301 Y53.57752 Z23.85561	
N11 G01 X-247.57578 Y34.89183 Z29.52308 F50000.00000	; En las secuencias de la N11 a la N21 se crea el componente lento del desplazamiento de aproximación. La herramienta ya se encuentra cerca de la pieza. Además, los ajustes de matricería y moldes (p. ej., COMPSURF) de CYCLE832 ya están activos (a causa de G01 activo). La trayectoria de esta fase, llamada de estabilización, para el comportamiento de matricería y moldes debería ser de 1000 veces la tolerancia de contorno (en este ejemplo, por tanto, 10 mm).
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	

Código de programa	**Comentarios**
N22 **G41** X-248.06237 Y30.32400 Z9.36695 A5=-.060431854 B5=.973045457 C5=-.222554556 A3=-.214974689 B3=.929398552 C3=-.300007025 F1061.03295	; A partir de N22, la normal a la superficie está definida por primera vez completamente a lo largo de toda la secuencia (es decir, la normal a la superficie está presente al final de la secuencia anterior N21 y, con ello, la normal a la superficie al inicio de la secuencia N22 y la normal a la superficie al final de la secuencia N22). Con ello se cumplen los requisitos para la activación de la corrección de herramienta con G41/G42.
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	
...	

Nota

Con el fresado frontal 3D con CUT3DFD, para la activación de la corrección de herramienta con G41/G42 es necesaria la definición de las normales a la superficie. La programación de G41/G42 sin definición de las normales a la superficie provoca la emisión de una alarma.

Consulte también

CYCLE832, High Speed Settings (Página 1146)

Información adicional

Fresado frontal 3D

Para este tipo de fresado 3D se necesita una descripción línea a línea de las trayectorias 3D sobre la superficie de la pieza. Los cálculos se realizan generalmente en sistemas CAM teniendo en cuenta la forma y las dimensiones de la herramienta. El postprocesador escribe en el programa de pieza, además de las secuencias CN, las orientaciones de la herramienta (si está activa la transformada de 5 ejes) y el comando G para la corrección de herramienta 3D deseada. Esto ofrece al operador de la máquina la posibilidad de utilizar herramientas ligeramente más pequeñas, apartándose de la herramienta utilizada para el cálculo de las trayectorias de CN.

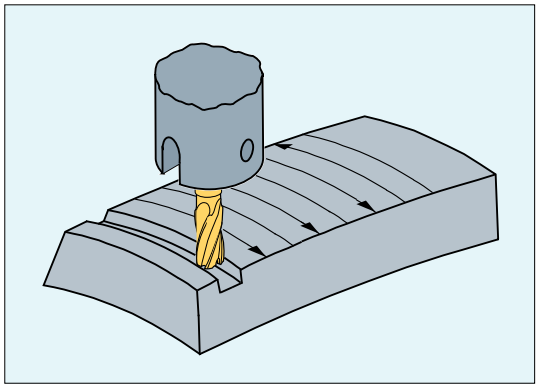


Figura 3-10 Fresado frontal

Método de aproximación

El método de aproximación es siempre NORM con las variantes 3D de la corrección de radio de herramienta.

Comportamiento en esquinas exteriores

En el fresado frontal, las esquinas exteriores se tratan como arcos con radio 0, en los que el plano del círculo se define por la tangente final de la primera secuencia y la tangente inicial de la segunda secuencia. De este modo también es posible un cambio de orientación en la transición de secuencia. Así pues, en una esquina exterior se inserta siempre un arco como elemento de contorno. En el fresado frontal no está disponible el método de punto de intersección.

Comportamiento con cambios de orientación en esquinas exteriores

Con los comandos G ORIC y ORID se puede definir si los cambios de orientación programados entre las dos secuencias de desplazamiento que forman la esquina se ejecutan antes del inicio de la secuencia circular insertada (ORID) o al mismo tiempo que esta (ORIC).

Corrección del radio de herramienta relativa a una herramienta diferencial

La corrección de radio de herramienta 3D relativa a una herramienta diferencial se selecciona con el comando CUT3DFD. Debe utilizarse cuando el contorno programado se refiere a la trayectoria del centro de una herramienta normalizada y el mecanizado se realiza con otra herramienta distinta. En tal caso, en el cálculo de la corrección del radio de herramienta en 3D se incluirán solo el valor de desgaste del radio de la herramienta activa ($TC_DP15) y el offset de corrección de herramienta OFFN y TOFFR/TOFFLR, en caso de haberse programado. El radio base ($TC_DP6) de la herramienta activa **no** se incluye en el cálculo.

El fresado frontal 3D con CUT3DFD solo es posible en combinación con el "Alisado de las normales a la superficie con fresado frontal 3D". Este se activa llamando la función sujeta a licencia "Top Surface" mediante CYCLE832(...). La activación debe realizarse **antes** de activar la corrección de herramienta con G41/G42, pero no directamente antes de la intervención de la herramienta, sino una longitud de trayectoria antes que corresponda a aproximadamente 1000 veces la tolerancia de contorno (p. ej., 1000 × 0,01 mm = 10 mm). La desactivación debe realizarse en orden inverso: primero desactivar la corrección de herramienta con G40; a continuación, tras una longitud de trayectoria de aproximadamente 1000 veces la tolerancia de contorno, realizar la desactivación con, p. ej., CUT2D (o similar).

Para poder utilizar el "Alisado de las normales a la superficie con fresado frontal 3D", también es necesario habilitar la función "Interpolación de las normales a la superficie mediante polinomios":

MD28291 $MC_MM_SMOOTH_SURFACE_NORMALS = TRUE

Nota

Para el fresado frontal 3D con CUT3DFD en combinación con "Top Surface", se deben tener en cuenta las recomendaciones de ajuste para "Top Surface".

En el SIOS Portal hay un programa verificador especial que puede utilizarse para comprobar los datos ajustados.

→ Programa verificador para Top Surface (https://support.industry.siemens.com/cs/ww/es/view/109738423/en)

Consulte también

CYCLE832, High Speed Settings (Página 1146)

3.13.5.3 Fresado periférico 3D considerando una superficie de limitación (CUT3DCC, CUT3DCCD)

En el fresado periférico 3D con modificación continua o constante de la orientación de la herramienta se programa frecuentemente la trayectoria del centro de la herramienta para una herramienta normalizada definida. Dado que, en la práctica, ocurre a menudo que no se dispone de las herramientas normalizadas adecuadas, se puede emplear una herramienta que no difiera excesivamente (≤ 5%) de una herramienta normalizada.

Con CUT3DCCD se considera para una herramienta diferencial real una superficie de limitación que describiría la herramienta normalizada programada. El programa CN describe la trayectoria del centro de la herramienta normalizada.

Con CUT3DCC se considera, en caso de utilizar herramientas cilíndricas, una superficie de limitación que habría alcanzado la herramienta normalizada programada. El programa CN describe el contorno en la superficie de mecanizado.

Como en el fresado frontal 3D, el vector normal a la superficie de limitación se especifica con A4, B4, C4 y A5, B5, C5.

Sintaxis

```
G41/G42 CUT3DCCD/CUT3DCC CDOF2 X... Y... Z... A4=... B4=... C4=... A5=... B5=...
C5=...
...
G40 X... Y... Z...
```

Descripción

CUT3DCCD:	3D-WRK para fresado periférico considerando una superficie de limitación con herramienta diferencial en la trayectoria del centro de la herramienta: Penetración en la superficie de limitación	
CUT3DCC:	3D-WRK para el fresado periférico considerando una superficie de limitación con corrección de radio 3D: Contorno en la superficie de mecanizado	
G41/G42 X... Y... Z...:	Activar la corrección de radio de herramienta	
	G41:	Corrección del radio de la herramienta a la izquierda del contorno
	G42:	Corrección del radio de la herramienta a la derecha del contorno
	Nota: La activación debe efectuarse en una secuencia lineal (G0/G1).	
CDOF2:	Desactivar vigilancia de colisión para fresado periférico 3D	
A4/5=... B4/5=... C4/5=...:	Definición de las normales a la superficie de limitación	
	A4=... B4=... C4=...:	Definición al inicio de secuencia
	A5=... B5=... C5=...:	Definición al final de secuencia
G40 X... Y... Z...:	Desactivación de la corrección del radio de corte/herramienta **Nota:** La desactivación debe efectuarse en una secuencia lineal (G0/G1) con desplazamientos de ejes geométricos.	

Nota

Los comandos G para la selección de la corrección de radio de herramienta 3D se evalúan en la secuencia de aproximación, es decir, normalmente en la secuencia que contiene G41 o G42.

G41 o G42 también pueden estar programados en secuencias sin desplazamiento en los ejes geométricos relevantes para la corrección. En este caso, la secuencia de aproximación es la primera secuencia de desplazamiento que sigue a una secuencia de este tipo.

Cuando la corrección de radio de herramienta está activa, se ignora sin generar alarma cualquier cambio de la variante de corrección de radio de herramienta 3D.

Ejemplo

```
Código de programa                          Comentarios
N10 $TC_DP1[1,1]=120                        ; Fresa cilíndrica
N20 $TC_DP6[1,1]=10
N30 $TC_DP15[1,1]=-3
...
; Mecanizado con fresa cilíndrica y CUT3DCCD
N110 TRAORI                                 ; Activación de la transformada.
N120 A4=0 B4=0 C4=1                         ; Definición de las normales a la super-
                                            ficie de limitación al inicio de la se-
                                            cuencia.
N130 X0 Y0 Z0 A0 C0 T1 D1 F20000
```

Código de programa	Comentarios
N140 X10 Y0 Z0 G41 **CUT3DCCD** CDOF2 G64	; Activación del fresado periférico 3D teniendo en cuenta la superficie de limitación + desactivación de la vigilancia de colisión.
N150 X20	
N160 X30 A45	; Ángulo obtuso ==> sin penetración
N170 X40 A-45	; Ángulo agudo ==> penetración
N180 X55	
N190 Y10 Z10	; Desplazamiento en dirección a la herramienta.
N200 Y20	
N210 C45	; Puro cambio de orientación.
N220 Y30 C90	
N230 A5=-1 B5=0 C5=2 Y40	; Cambio de la superficie.
N240 Y50 G40	; Desactivación de la corrección de radio de herramienta.
...	

Información adicional

Tipo de herramienta

Se evalúa el tipo de herramienta (parámetro de herramienta $TC_DP1). Solo se admiten fresas con mango cilíndrico (fresa cilíndrica o de mango, así como fresa tórica y, en caso extremo, las fresas cilíndricas de matrices). Esto corresponde a los tipos de herramienta 1 a 399, con excepción de los números 111 y 155 a 157.

Herramientas normalizadas con redondeo de esquinas

El redondeo de esquinas de la herramienta normalizada se describe con el parámetro de herramienta $TC_DP7. La desviación del redondeo de esquinas de la herramienta real con respecto a la herramienta normalizada resulta del parámetro de herramienta $TC_DP16.

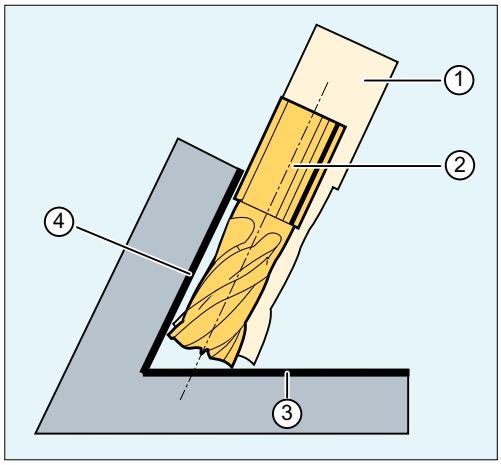
Ejemplo: Fresa tórica con radio reducido frente a la herramienta normalizada

Tipo de herramienta	Radio del vástago (R)	Radio de redondeo (r)
Herramienta normalizada con redondeo de esquinas	R = $TC_DP6	r = $TC_DP7
Herramienta real con redondeo de esquinas Tipos de herramienta 121 y 131 fresas tóricas (fresas de mango con redondeo de esquinas)	R' = $TC_DP6 + $TC_DP15 + OFFN	r' = $TC_DP7 + $TC_DP16

En este ejemplo son negativos tanto $TC_DP15 + OFFN como $TC_DP16.

3D-WRK con CUT3DCCD: Trayectoria del centro de la herramienta con penetración hasta la superficie de limitación

Si se utiliza una herramienta con un radio inferior al de la herramienta normalizada, la fresa que penetra en dirección longitudinal se sigue desplazando hasta que vuelve a tocar el fondo de la caja. De este modo, la esquina formada por las superficies de mecanizado y de limitación se vacía tanto como lo permita la herramienta. Se trata de un método de mecanización mixto que combina el fresado periférico y el fresado frontal. De modo análogo a la herramienta con radio reducido, en el caso de una herramienta con el radio aumentado, la penetración se produce en la dirección opuesta.



① Herramienta normalizada
② Herramienta con radio más pequeño que penetra hasta la superficie de limitación
③ Superficie de limitación
④ Superficie de mecanizado

A diferencia de todas las demás correcciones de herramienta del grupo G 22, un parámetro de herramienta $TC_DP6 especificado para CUT3DCCD no influye en el radio de la herramienta y, por lo tanto, tampoco en la corrección resultante. El offset de corrección resulta de la suma del valor de desgaste del radio de la herramienta (parámetro de herramienta $TC_DP15) y de un offset de herramienta programado para el cálculo del offset vertical a la superficie de limitación OFFN.

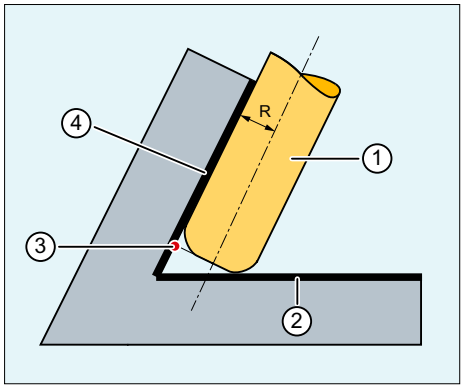
No se puede determinar a partir del programa de pieza generado si la superficie a mecanizar se halla a la izquierda o a la derecha de la trayectoria. Por ello se asume un radio positivo y un valor de desgaste negativo de la herramienta original. Un valor de desgaste negativo genera siempre una herramienta con diámetro reducido.

Uso de herramientas cilíndricas

Para el uso de herramientas cilíndricas, sólo es necesaria la penetración en caso de que la superficie de mecanizado y la de limitación formen un ángulo agudo (inferior a 90 grados). Si se usan fresas tóricas (fresas de mango con redondeo de esquinas), será necesaria una penetración en la dirección longitudinal de la herramienta tanto con ángulo agudo como con ángulo obtuso.

3D-WRK con CUT3DCC: Contorno en la superficie de mecanizado

Si CUT3DCC está activo con una fresa tórica, la trayectoria programada hace referencia a una fresa cilíndrica ficticia del mismo diámetro. En la figura siguiente se muestra el punto de referencia de la trayectoria resultante utilizando una fresa tórica.



① Fresa tórica
② Superficie de limitación
③ Punto de referencia de la trayectoria
④ Superficie de mecanizado
R Radio del vástago (radio de la herramienta)

Es admisible que el ángulo entre las superficies de mecanizado y de limitación se transforme de agudo a obtuso o viceversa incluso dentro de una misma secuencia.

La herramienta real utilizada puede ser mayor o menor que la herramienta normalizada. Sin embargo, el radio de redondeo resultante no debe ser negativo, y debe mantenerse el signo del radio de herramienta resultante.

Con CUT3DCC, el programa de pieza CN hace referencia al contorno en la superficie de mecanizado. Como sucede con la corrección de radio de herramienta convencional, se toma en consideración el radio total compuesto por la suma de los siguientes componentes:

- El radio de herramienta (parámetro de herramienta $TC_DP6)
- El valor de desgaste (parámetro de herramienta $TC_DP15)
- Un offset de herramienta programado para el cálculo del offset vertical con la superficie de limitación OFFN

La posición de la superficie de limitación se determina a partir de la siguiente diferencia:

Dimensiones de la herramienta normalizada - radio de herramienta (parámetro de herramienta $TC_DP6)

Advanced Surface/Top Surface

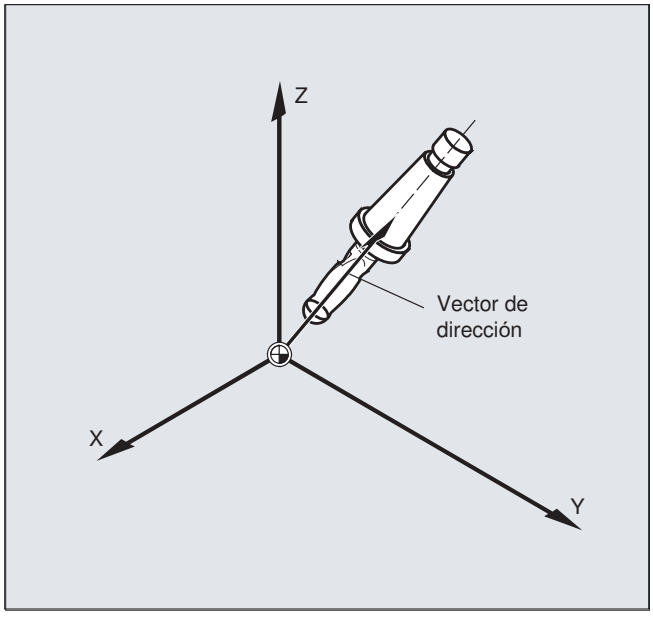
Nota

Si se utiliza la corrección de radio de herramienta CUT3DCC/CUT3DCCD en combinación con la función sujeta a licencia "Advanced Surface" o "Top Surface", deben tenerse en cuenta las recomendaciones de ajuste para "Advanced Surface"/"Top Surface". En el SIOS Portal hay un programa verificador especial que puede utilizarse para comprobar los datos ajustados.

→ Programa verificador para Advanced Surface/Top Surface (https://support.industry.siemens.com/cs/ww/es/view/109738423/en)

3.13.6 Orientación de herramienta (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST)

Se entiende por orientación de la herramienta la alineación geométrica de la herramienta en el espacio. En una máquina de mecanizado de 5 ejes es posible definir la orientación de la herramienta mediante comandos de programa.



Los movimientos de transición de secuencias de la orientación activados con `OSD` y `OST` se producen de forma diferente según el tipo de interpolación para la orientación de la herramienta.

En el caso de interpolación activa vectorial, el desarrollo alisado de la orientación se interpola también por medio de interpolación vectorial. Por el contrario, en el caso de interpolación activa de ejes giratorios, la orientación se alisa directamente mediante movimientos de los ejes giratorios.

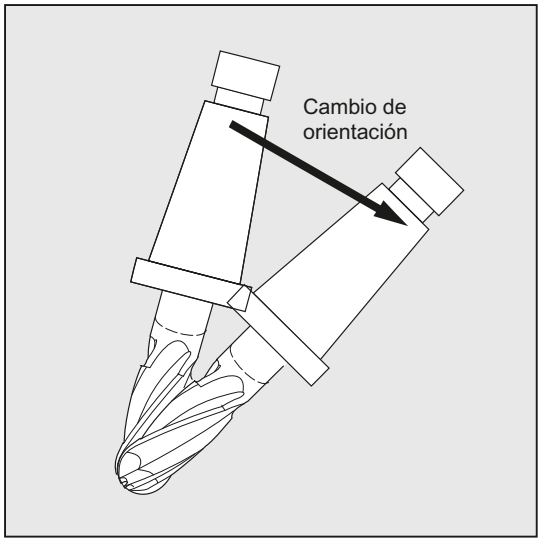
Programación

Programar el cambio de orientación:

Se puede programar un cambio en la orientación de la herramienta mediante:

- programación directa de los ejes giratorios `A`, `B`, `C` (interpolación de ejes giratorios)
- Ángulo euleriano o RPY
- Vector de dirección (interpolación vectorial a través de la indicación de `A3` o `B3` o `C3`)
- `LEAD/TILT` (fresado frontal)

El sistema de coordenadas de referencia puede ser
el sistema de coordenadas de máquina (`ORIMKS`) o bien el sistema de coordenadas actualmente asociado a la pieza (`ORIWKS`).



Programar la orientación de la herramienta:

`ORIC`:	Orientación y desplazamiento de trayectoria paralelos
`ORID`:	Orientación y desplazamiento de trayectoria sucesivos
`OSOF`:	Sin alisado de la orientación
`OSC`:	Orientación constante
`OSS`:	Alisado de la orientación sólo al principio de la secuencia
`OSSE`:	Alisado de la orientación al principio y al final de la secuencia
`ORIS`:	Velocidad del cambio de orientación con el alisado de orientación activado en grados por mm (válido para `OSS` y `OSSE`)

OSD:	Matado de esquinas de la orientación mediante la especificación de la longitud de matado de esquinas con el dato de operador: DO42674 $SC_ORI_SMOOTH_DIST
OST:	Matado de esquinas de la orientación mediante la especificación de la tolerancia angular en grados en la interpolación vectorial con el dato de operador: DO42676 $SC_ORI_SMOOTH_TOL En el caso de interpolación de ejes giratorios, la tolerancia definida se toma como desviación máxima de los ejes de orientación.

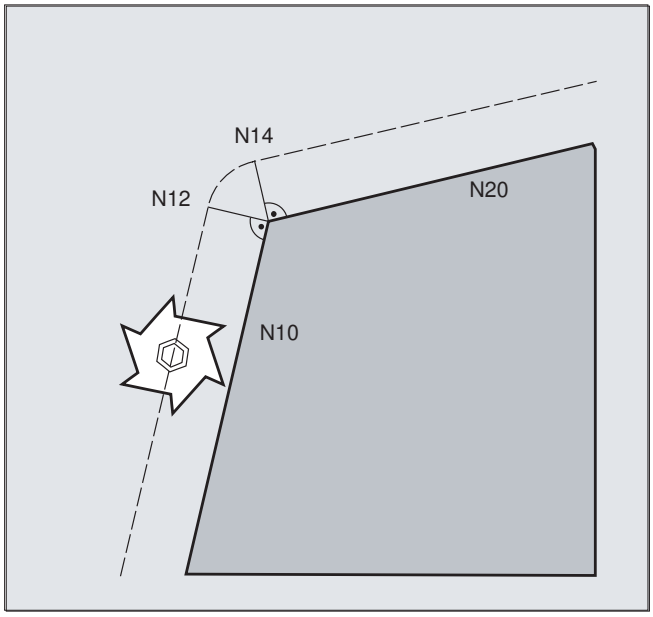
Nota

Todos los comandos para el matado de esquinas de la orientación de herramienta (OSOF, OSC, OSS, OSSE, OSD y OST) están reunidos en el grupo G 34. Son modalmente activos, es decir, sólo puede estar activo uno de estos comandos cada vez.

Ejemplos

Ejemplo 1: ORIC

Si entre las secuencias de desplazamiento N1 y N20 se han programado dos o más secuencias con cambios de orientación (p. ej.:A2=... B2=... C2=...) y está activo ORIC, la secuencia circular insertada se dividirá entre dichas secuencias intermedias en función del valor de los cambios de ángulo.

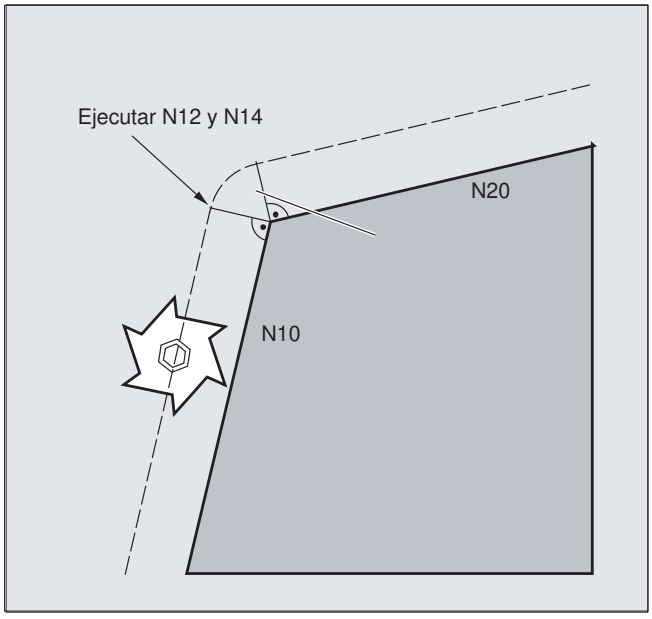


Código de programa	Comentarios
ORIC	
N8 A2=… B2=… C2=…	
N10 X… Y… Z…	

Código de programa	Comentarios
N12 C2=… B2=…	; La secuencia circular que se inserta en la esquina exterior se reparte entre N12 y N14, en función de la modificación de la orientación. El movimiento circular y el cambio de orientación se ejecutan paralelamente.
N14 C2=… B2=…	
N20 X =…Y=… Z=… G1 F200	

Ejemplo 2: ORID

Cuando `ORID` está activo, todas las secuencias entre las dos secuencias de desplazamiento se ejecutan al final de la primera secuencia de desplazamiento. La secuencia circular con orientación constante se ejecuta inmediatamente antes de la segunda secuencia de desplazamiento.



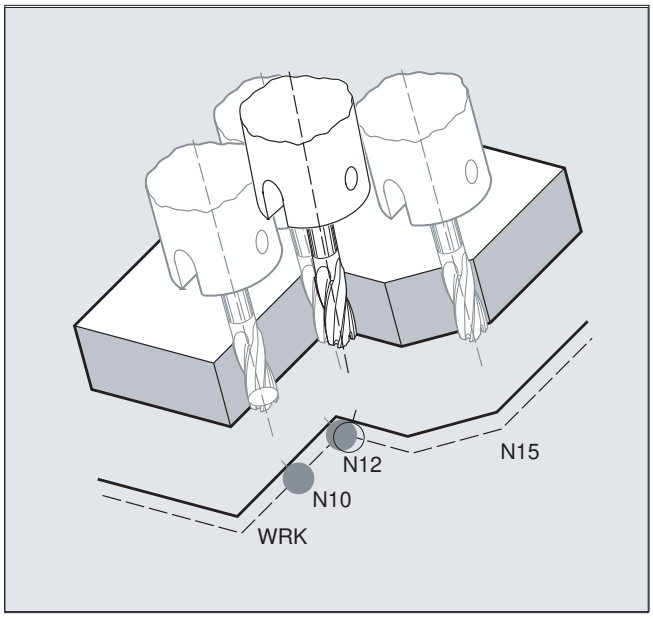
Código de programa	Comentarios
ORID	
N8 A2=… B2=… C2=…	
N10 X… Y… Z…	
N12 A2=… B2=… C2=…	; Las secuencias N12 y N14 se ejecutan al final de N10. A continuación se ejecuta la secuencia circular con la orientación actual.
N14 M20	; Funciones auxiliares, etc.
N20 X… Y… Z…	

Nota

El tipo del cambio de orientación en una esquina exterior viene determinado por el comando de programa activo en la primera secuencia de desplazamiento de una esquina exterior.

Sin cambio de orientación: Si la orientación no se modifica en el límite de la secuencia, la sección de la herramienta es un círculo que toca ambos contornos.

Ejemplo 3: cambio de la orientación en una esquina interior



Código de programa

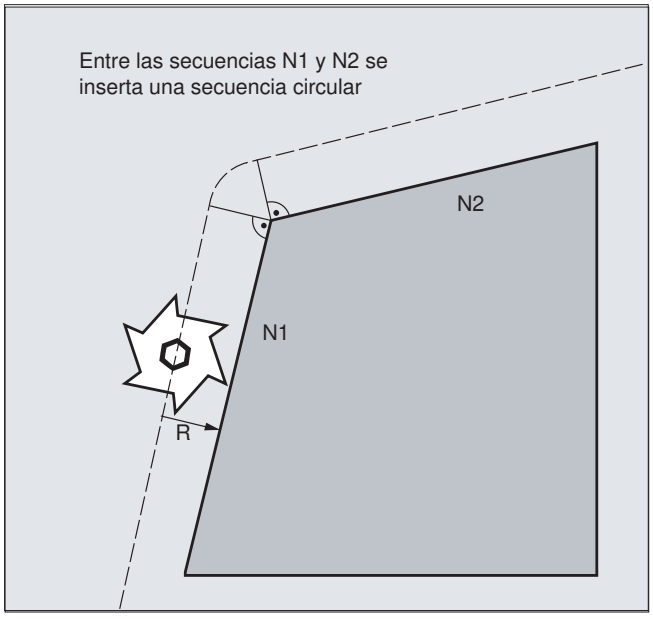
```
ORIC
N10 X …Y… Z… G1 F500
N12 X …Y… Z… A2=… B2=… C2=…
N15 X …Y… Z… A2=… B2=… C2=…
```

Información adicional

Comportamiento en esquinas exteriores

En una esquina exterior se inserta siempre una secuencia circular con el radio de la fresa.

Con los comandos `ORIC` y `ORID` se puede definir si los cambios de orientación programados entre las secuencias `N1` y `N2` se ejecutan antes del inicio de la secuencia circular insertada o simultáneamente con ésta.



Si es necesario efectuar un cambio de orientación en esquinas exteriores, éste se puede realizar paralelamente a la interpolación o por separado con el movimiento interpolado.

Con `ORID` se ejecutan primero las secuencias insertadas sin desplazamiento de trayectoria. La secuencia circular se inserta inmediatamente delante de la segunda de las dos secuencias de desplazamiento que forman la esquina.

Si, en una esquina exterior, se han insertado varias secuencias de orientación y seleccionado `ORIC`, el movimiento circular se distribuye entre las distintas secuencias insertadas en función de las magnitudes del cambio de orientación.

Matado de esquinas de la orientación con OSD u OST

En el matado de esquinas con `G642`, la desviación máxima para los ejes de contorno y los ejes de orientación no puede ser muy diferente. La tolerancia menor de ambos determina la forma del movimiento de matado de esquinas o la tolerancia angular para alisar con relativa intensidad el desarrollo de la orientación, sin que ello implique desviaciones importantes del contorno.

Mediante la activación de `OSD` y `OST` es posible alisar "generosamente" desviaciones muy pequeñas del desarrollo de la orientación sin desviaciones graves de contorno, con una longitud de matado de esquinas y una tolerancia angular definidas.

Nota

A diferencia del matado de esquinas del contorno (y del desarrollo de la orientación) con `G642`, en la transición de secuencias con `OSD` u `OST` no se genera ninguna secuencia propia, sino que el movimiento de transición de secuencias se agrega directamente a las secuencias originales programadas.

Con `OSD` u `OST` no puede hacerse un matado de esquinas en ninguna transición de secuencias en la que tenga lugar un cambio del tipo de interpolación (vector → eje giratorio, eje giratorio → vector) para la orientación de herramienta. En estas transiciones de secuencias pueden matarse las esquinas, en caso necesario, con las funciones habituales de matado `G641`, `G642` o `G643`.

3.13.7 Asignación libre de números D, número de filo

3.13.7.1 Asignación libre de números D, número de filo (dirección CE)

Número D

Los números D se pueden utilizar como números de corrección. El número de filo se puede direccionar también a través de la dirección CE. Es posible describir el número de filo a través de la variable de sistema $TC_DPCE.

Ajuste predeterminado: N.º de corrección == n.º de filo

La cantidad máxima de números D (números de filo) y la cantidad máxima de filos por herramienta (→ fabricante de la máquina) se definen a través de datos de máquina. Los siguientes comandos solo son útiles si la cantidad máxima de filos (MD18105) se ha definido mayor que la cantidad de filos por herramienta (MD18106). Tenga en cuenta las indicaciones del fabricante de la máquina herramienta.

Información adicional

Manual de funciones Herramientas

3.13.7.2 Asignación libre de números D: Comprobar números D (CHKDNO)

El comando `CKKDNO` permite comprobar si los números D existentes se han asignado de manera unívoca. Los números D de todas las herramientas definidas en una unidad TO sólo deben aparecer una vez. No se tienen en cuenta las herramientas de recambio.

Sintaxis

```
state=CHKDNO(Tno1,Tno2,Dno)
```

Descripción

state:	=TRUE:	Los números D han sido asignados de forma unívoca para el rango comprobado.
	=FALSE:	Se ha producido una colisión de números D, o la parametrización es inválida. A través de Tno1, Tno2 y Dno se transfieren los parámetros que han producido la colisión. Estos datos se pueden evaluar en el programa de pieza.
CHKDNO(Tno1,Tno2):	Se comprueban todos los números D de las herramientas indicadas.	
CHKDNO(Tno1):	Se comprueban todos los números D de Tno1 frente a todas las demás herramientas.	
CHKDNO:	Se comprueban todos los números D de todas las herramientas frente a todas las demás herramientas.	

3.13.7.3 Asignación libre de números D: Cambiar nombre de números D (GETDNO, SETDNO)

Los números D se tienen que asignar de forma unívoca. Dos filos distintos de una herramienta no pueden tener el mismo número D.

GETDNO

Este comando suministra el número D de un determinado filo (ce) de una herramienta con el número T "t". Si no existe ningún número D para los parámetros introducidos, se ajusta d=0. Si el número D es inválido, se devuelve un valor superior a 32000.

SETDNO

Con este comando se asigna el valor d del número D a un filo CE de la herramienta t. A través de state se devuelve el resultado de esta instrucción (TRUE o FALSE). Si no existe ningún registro para los parámetros introducidos, se devuelve FALSE. Los errores de sintaxis generan una alarma. El número D no se puede poner explícitamente a 0.

Sintaxis

```
d = GETDNO (t,ce)
state = SETDNO (t,ce,d)
```

Descripción

d:	Número D del filo de la herramienta
t:	Número T de la herramienta
ce:	Número de filo (número CE) de la herramienta
state:	Indica si el comando se ha podido ejecutar sin errores (TRUE o FALSE).

Ejemplo: renombrar un número D

Programación	Comentarios
$TC_DP2[1,2] = 120	

```
Programación                                          Comentarios
$TC_DP3[1,2]  = 5.5
$TC_DPCE[1,2] = 3                                     ; Número del filo CE
...
N10 def int N°Dantiguo, N°Dnuevo = 17
N20 N°Dantiguo = GETDNO(1,3)
N30 SETDNO(1,3,N°Dnuevo)
```

De este modo, se asigna al filo CE=3 el nuevo valor D de 17. Entonces, los datos de este filo se activan a través del número D 17, tanto a través de las variables de sistema como también en la programación con la dirección CN.

3.13.7.4 Asignación libre de números D: Determinar el número T para el número D predeterminado (GETACTTD)

La función predefinida GETACTTD calcula el número T correspondiente a un número D absoluto. No tiene lugar ninguna comprobación del carácter unívoco. Si existen varios números D iguales dentro de una unidad TO, se devolverá el número T de la primera herramienta encontrada.

Sintaxis

```
<status>=GETACTTD(<TNr>,<DNr>)
```

Descripción

GETACTTD():	Llamada de función		
<DNr>:	Número D para el cual se tiene que buscar el número T.		
	Tipo de dato:	INT	
<TNr>:	Número T encontrado		
	Tipo de dato:	VAR INT	
<status>:	Resultado		
	Tipo de dato:	INT	
	Valor:	0	Se ha encontrado el número T. <TNr> contiene el valor del número T.
		-1	Para el número D indicado no existe ningún número T, <TNr> = 0.
		-2	El número D no es absoluto. <TNr> obtiene el valor de la primera herramienta encontrada que contiene el número D con el valor <DNr>.
		-5	La función no pudo ejecutarse por otro motivo.

3.13.7.5 Asignación libre de números D: Fijación como no válidos de los números D (DZERO)

El comando DZERO sirve de refuerzo durante el Reset. Los registros de corrección marcados de esta manera ya no son comprobados por el comando CHKDNO. Para que vuelvan a ser accesibles, el número D debe fijarse de nuevo con SETDNO.

Sintaxis

```
DZERO
```

Descripción

DZERO:	Marca todos los números D de la unidad TO como no válidos.

3.13.8 Cinemática del portaherramientas

Requisitos

Un portaherramientas sólo puede orientar una herramienta en todas las direcciones posibles en el espacio si

- existen dos ejes giratorios V_1 y V_2.
- los ejes giratorios están situados en posición perpendicular entre ellos.
- el eje longitudinal de la herramienta se encuentra en posición perpendicular frente al segundo eje giratorio V_2.

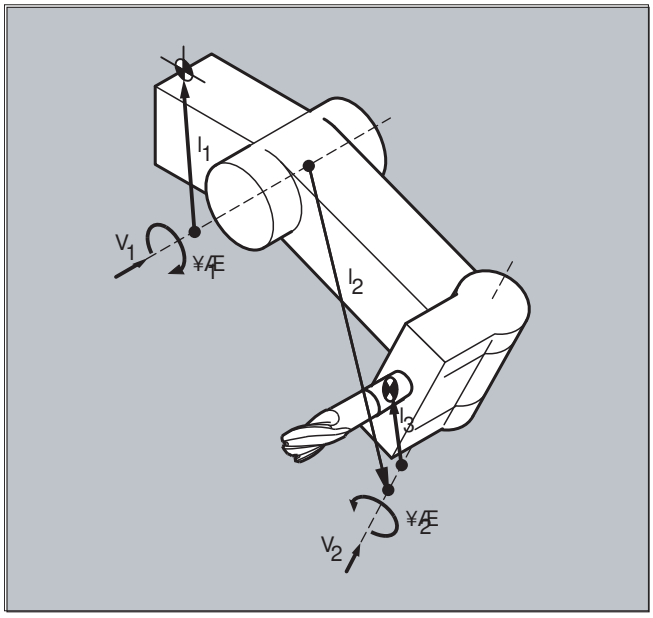
Adicionalmente, se aplica el siguiente requisito para máquinas en las cuales se tienen que poder ajustar todas las orientaciones posibles:

- la orientación de la herramienta tiene que ser perpendicular al primer eje giratorio V_1.

Función

La cinemática de portaherramientas con máx. dos ejes giratorios v_1 o v_2 se describe a través de las 17 variables de sistema $TC_CARR1[m] a $TC_CARR17[m]. La descripción del portaherramientas se compone de:

- la distancia vectorial del primer eje giratorio al punto de referencia del portaherramientas I_1, la distancia vectorial entre el primer y el segundo eje giratorio I_2, la distancia vectorial entre el segundo eje giratorio y el punto de referencia de la herramienta I_3.
- los vectores de dirección de ambos ejes giratorios V_1, V_2.
- los ángulos de giro α1, α2 en torno a los dos ejes. Los ángulos de giro se cuentan en positivo en sentido horario, con vista en la dirección de los vectores de eje giratorio.



Para máquinas con **cinemática simple** (tanto la herramienta como la pieza se pueden girar), las variables de sistema se han ampliado con las entradas $TC_CARR18[m] a $TC_CARR23[m].

Parámetros

Función de las variables de sistema para portaherramientas orientables			
Denominación	**Componente x**	**Componente y**	**Componente z**
I_1 Vector de corrección	$TC_CARR1[m]	$TC_CARR2[m]	$TC_CARR3[m]
I_2 Vector de corrección	$TC_CARR4[m]	$TC_CARR5[m]	$TC_CARR6[m]
v_1 Eje giratorio	$TC_CARR7[m]	$TC_CARR8[m]	$TC_CARR9[m]
v_2 Eje giratorio	$TC_CARR10[m]	$TC_CARR11[m]	$TC_CARR12[m]

Función de las variables de sistema para portaherramientas orientables			
α_1 Ángulo de rotación α_2 Ángulo de rotación	$TC_CARR13[m] $TC_CARR14[m]		
l_3 Vector de corrección	$TC_CARR15[m]	$TC_CARR16[m]	$TC_CARR17[m]

Ampliaciones de las variables de sistema para portaherramientas orientables			
Denominación	**Componente x**	**Componente y**	**Componente z**
l_4 Vector de corrección	$TC_CARR18[m]	$TC_CARR19[m]	$TC_CARR20[m]
Identificador de eje Eje giratorio v_1 Eje giratorio v_2	Denominador de eje de los ejes giratorios v_1 y v_2 (la ocupación previa es cero) $TC_CARR21[m] $TC_CARR22[m]		
Tipo de cinemática **T**ool **P**art **M**ixed mode	$TC_CARR23[m]		
	Tipo de cinemática T ->	Tipo de cinemática P ->	Tipo de cinemática M
	Sólo la herramienta es giratoria (ocupación previa)	Sólo la pieza es giratoria	La pieza y la herramienta son giratorias
Corrección del eje giratorio v_1 eje giratorio v_2	Ángulo en grados de los ejes giratorios v_1 y v_2 al ocupar la posición básica $TC_CARR24[m] $TC_CARR25[m]		
Offset de ángulo del eje giratorio v_1 eje giratorio v_2	Corrección del dentado Hirth en grados de los ejes giratorios v_1 y v_2 $TC_CARR26[m] $TC_CARR27[m]		
Incremento de ángulo v_1 eje giratorio v_2 eje giratorio	Incremento del dentado Hirth en grados de los ejes giratorios v_1 y v_2 $TC_CARR28[m] $TC_CARR29[m]		
Posición mín. eje giratorio v_1 eje giratorio v_2	Límite de software para la posición mínima de los ejes giratorios v_1 y v_2 $TC_CARR30[m] $TC_CARR31[m]		
Posición máx. eje giratorio v_1 eje giratorio v_2	Límite de software para la posición máxima de los ejes giratorios v_1 y v_2 $TC_CARR32[m] $TC_CARR33[m]		
Nombre del portaherramientas	En lugar de un número se puede asignar un nombre al portaherramientas. $TC_CARR34[m]		
Usuario: Nombre de eje 1 Nombre de eje 2 Identificación **Posición**	Uso previsto por el usuario dentro de los ciclos de medida. $TC_CARR35[m] $TC_CARR36[m] $TC_CARR37[m]		
	$TC_CARR38[m]	$TC_CARR39[m]	$TC_CARR40[m]
Decalaje fino	Parámetros que se pueden sumar **a los valores en los parámetros básicos**.		
l_1 Vector de corrección	$TC_CARR41[m]	$TC_CARR42[m]	$TC_CARR43[m]
l_2 Vector de corrección	$TC_CARR44[m]	$TC_CARR45[m]	$TC_CARR46[m]

Ampliaciones de las variables de sistema para portaherramientas orientables			
l_3 Vector de corrección	$TC_CARR55[m]	$TC_CARR56[m]	$TC_CARR57[m]
l_4 Vector de corrección	$TC_CARR58[m]	$TC_CARR59[m]	$TC_CARR60[m]
v_1 Eje giratorio	$TC_CARR64[m]		
v_2 Eje giratorio	$TC_CARR65[m]		

Nota

Explicaciones sobre los parámetros

Con "m" se indica el número del portaherramientas a describir.

$TC_CARR47 a $TC_CARR54, así como $TC_CARR61 a $TC_CARR63 no están definidas y generan una alarma en caso de intento de acceso de lectura o escritura.

Los puntos iniciales y finales de los vectores de distancia en los ejes se pueden elegir libremente. Los ángulos de giro α_1, $\alpha_2$alrededor de los dos ejes se definen en el estado básico del portaherramientas con 0°. De este modo, la cinemática de un portaherramientas se puede describir con un número ilimitado de posibilidades.

Los portaherramientas con un único eje o sin eje giratorio se pueden describir mediante la puesta a cero de los vectores de dirección en uno o ambos ejes giratorios.
En un portaherramientas sin eje giratorio, los vectores de distancia actúan como correcciones de herramienta adicionales en cuyos componentes no se influye al conmutar los planos de mecanizado (G17 a G19).

Ampliaciones de los parámetros

Parámetros de los ejes giratorios

Las variables de sistema se ampliaron con las entradas $TC_CARR24[m] a $TC_CARR33[m] y se describen como sigue:

La **corrección** de los ejes giratorios v_1, v_2	Modificación de la posición del eje giratorio v_1 o v_2 en la posición preferencial del portaherramientas orientable.
Offset de ángulo/ incremento de ángulo ejes giratorios v_1, v_2	Offset o incremento de ángulo del dentado Hirth de los ejes giratorios v_1 y v_2. El ángulo programado o calculado se redondea al siguiente valor que se obtiene con un n entero de phi = s + n * d.
Posición mínima y máxima ejes giratorios v_1, v_2	La posición mínima/máxima del eje giratorio, Ángulo límite (límite de software) del eje giratorio v1 y v2.

Parámetros para el usuario

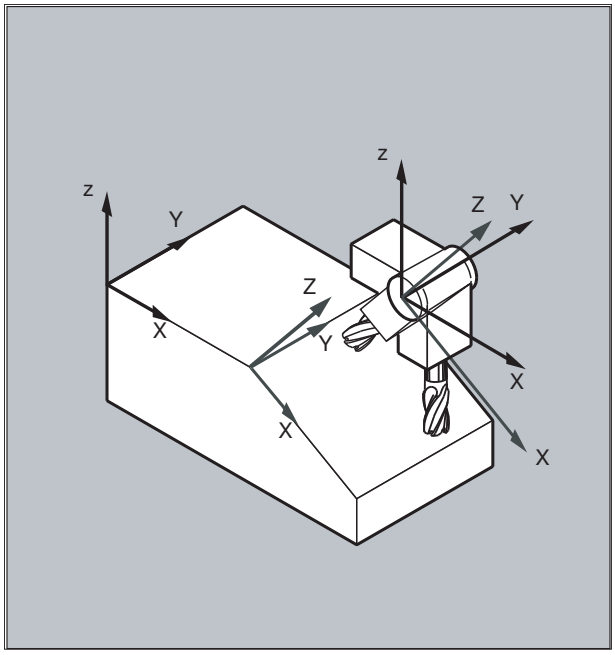
$TC_CARR34 a $TC_CARR40 contienen parámetros disponibles para el usuario que, de manera estándar no se evalúan o no tienen ninguna importancia dentro de NCK hasta el SW 6.4.

Parámetros del decalaje fino

$TC_CARR41 a $TC_CARR65 contienen parámetros de decalaje fino que pueden añadirse a los valores de los parámetros básicos. El valor de decalaje fino asignado a un parámetro básico se obtiene sumando el valor 40 al número de parámetro.

Ejemplo

El portaherramientas utilizado en el siguiente ejemplo se puede describir completamente con un giro alrededor del eje Y.



Código de programa	**Comentarios**
N10 $TC_CARR8[1]=1	; Definición de la componente Y del primer eje giratorio del portaherramientas 1.
N20 $TC_DP1[1,1]=120	; Definición de una fresa normal.
N30 $TC_DP3[1,1]=20	; Definición de una fresa normal con una longitud de 20 mm.
N40 $TC_DP6[1,1]=5	; Definición de una fresa normal con un radio de 5 mm.
N50 ROT Y37	; Definición de frame con giro de 37° alrededor del eje Y.
N60 X0 Y0 Z0 F10000	; Desplazamiento a la posición inicial.
N70 G42 CUT2DF TCOFR TCARR=1 T1 D1 X10	; Corrección del radio, ajustar corrección longitudinal de herramienta en el frame girado, portaherramientas 1, seleccionar herramienta 1.
N80 X40	; Realizar el mecanizado con un giro de 37°.

Código de programa	Comentarios
N90 Y40	
N100 X0	
N110 Y0	
N120 M30	

Información adicional

Cinemática simple

Para máquinas con cinemática simple (tanto la herramienta como también la pieza se pueden girar), las variables de sistema se amplían con las entradas `$TC_CARR18[m]` a `$TC_CARR23[m]` y se describen como sigue:

La mesa de herramientas giratoria, compuesta de:

- La distancia vectorial del segundo eje giratorio V_2 al punto de referencia de una mesa de herramientas giratoria I_4 del tercer eje giratorio.

Los ejes giratorios compuestos de:

- Los dos identificadores de canal para la referencia de los ejes giratorios V_1 y V_2, a cuya posición se tendrá acceso, dado el caso, durante la determinación de la orientación del portaherramientas orientable.

El tipo de cinemática con uno de los valores T, P o M:

- Tipo de cinemática T: Sólo la herramienta es giratoria.
- Tipo de cinemática P: Sólo la pieza es giratoria.
- Tipo de cinemática M: La herramienta y la pieza son giratorias.

Borrar los datos del portaherramientas

Con `$TC_CARR1[0]=0` pueden borrarse los datos de todos los juegos de datos de portaherramientas.

El tipo de cinemática `$TC_CARR23[T]=T` debe definirse con una de las tres letras mayúsculas o minúsculas permitidas (T,P,M) y, por este motivo, no debe borrarse.

Modificar los datos del portaherramientas

Cada uno de los valores descritos se puede modificar asignando un nuevo valor en el programa de pieza. Cualquier otro carácter que no sea T, P o M conlleva una alarma si se intenta activar el portaherramientas orientable.

Lectura de los datos de portaherramientas

Cada uno de los valores descritos se puede leer asignándolo a una variable en el programa de pieza.

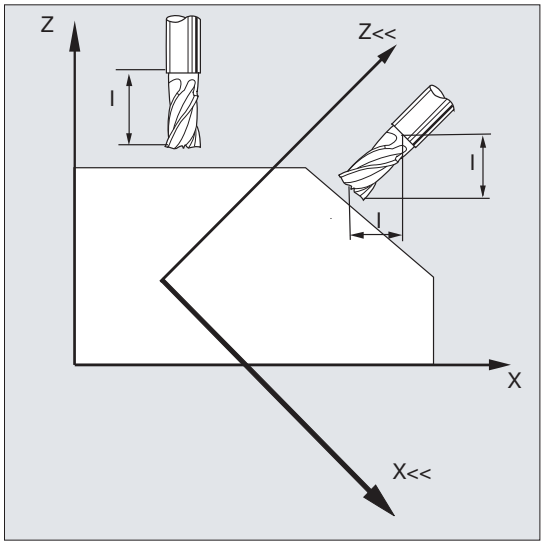
Decalajes finos

Un valor de decalaje fino no permitido se detecta por primera vez cuando un portaherramientas orientable que contiene uno de estos valores se activa y al mismo tiempo el dato de operador DO42974 $SC_TOCARR_FINE_CORRECTION = TRUE.

La magnitud del decalaje fino admisible queda limitada a través de datos de máquina a un valor máximo admisible.

3.13.9 Corrección longitudinal de herramienta para portaherramientas orientables (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)

Cuando cambia la orientación espacial de la herramienta, también se modifican los componentes longitudinales de la misma.



Después de un Reset (p. ej.: tras un ajuste manual o un cambio del portaherramientas con una orientación definida y fija en el espacio) se deberán de recalcular de nuevo los componentes longitudinales de la herramienta. Esto se realiza con los comandos de desplazamiento `TCOABS` y `TCOFR`.

Con un portaherramientas orientable de un frame activo es posible determinar, en la selección de la herramienta, con `TCOFRZ`, `TCOFRY` y `TCOFRX`, la dirección en la que ha de apuntar la herramienta.

Sintaxis

```
TCARR=[<m>]
TCOABS
TCOFR
TCOFRZ
TCOFRY
TCOFRX
```

Descripción

`TCARR=[<m>]`:	Solicitar portaherramientas con el número "m"
`TCOABS`:	Calcular las componentes longitudinales de la herramienta a partir de la orientación actual del portaherramientas.

TCOFR:	Determinar las componentes longitudinales de la herramienta a partir de la orientación del frame activo
TCOFRZ:	Portaherramientas orientable del frame activo cuya herramienta apunta en la dirección Z
TCOFRY:	Portaherramientas orientable del frame activo cuya herramienta apunta en la dirección Y
TCOFRX:	Portaherramientas orientable del frame activo cuya herramienta apunta en la dirección X

Información adicional

Corrección longitudinal de herramienta a partir de la orientación del portaherramientas (TCOABS)

TCOABS calcula la corrección longitudinal de la herramienta a partir de los ángulos de orientación actuales del portaherramientas (que se encuentran en las variables del sistema $TC_CARR13 y $TC_CARR14).

Para la definición de la cinemática del portaherramientas con variables de sistema, ver " Cinemática del portaherramientas (Página 801)".

Para recalcular la corrección longitudinal de herramienta al producirse un cambio de frame, se deberá volver a seleccionar de nuevo la herramienta.

Dirección de la herramienta del frame activo

El portaherramientas orientable se puede ajustar de modo que la herramienta apunte en las siguientes direcciones:

- con TCOFR o TCOFRZ en dirección Z
- con TCOFRY en dirección Y
- con TCOFRX en dirección X

El conmutar entre TCOFR y TCOABS provoca el recálculo de la corrección longitudinal de herramienta.

Solicitar portaherramientas (TCARR)

Con TCARR se solicitan los datos geométricos asociados al número de portaherramientas m (memoria de correcciones).

Con m=0 se deselecciona el portaherramientas activo.

Los datos geométricos del portaherramientas solamente se activan tras la llamada a la herramienta. La herramienta seleccionada permanece activa hasta que se realice un cambio de portaherramientas.

Los datos geométricos del portaherramientas actual se pueden definir desde el programa de pieza mediante las correspondientes variables del sistema.

Nuevo cálculo de la corrección longitudinal de herramienta (TCOABS) al producirse un cambio de frame

Para recalcular la corrección longitudinal de herramienta al producirse un cambio de frame, se deberá volver a seleccionar de nuevo la herramienta.

Nota

La orientación de la herramienta se deberá adaptar al frame activo de forma manual.

Para el cálculo de la corrección longitudinal de la herramienta también se determina en un paso intermedio el ángulo de giro del portaherramientas. Como para portaherramientas con dos ejes giratorios generalmente existen dos pares de ángulos mediante los cuales se puede realizar la adaptación de la orientación de la herramienta al frame activo, los valores angulares de giro memorizados en las variables del sistema deberán de corresponder aproximadamente a los ángulos de giro ajustados mecánicamente.

Nota

Orientación de la herramienta

El control no puede verificar si los ángulos de giro calculados mediante la orientación del frame se pueden ajustar en la máquina o no.

Si los ejes giratorios del portaherramientas tienen una disposición constructiva que impide alcanzar la orientación calculada por el frame, entonces se emite una alarma.

La combinación de la corrección fina de herramienta y las funciones para la corrección longitudinal de herramienta en portaherramientas desplazables, no está permitida. Si se llama a las dos funciones simultáneamente, se emite un mensaje de error.

`TOFRAME` permite definir un frame en base a la dirección de la orientación del portaherramientas seleccionado. Para más información, ver el capítulo "Frames".

Cuando está activa la transformada de orientación (transformada de 3, 4, 5 ejes), es posible seleccionar un portaherramientas con una orientación que se desvíe de la posición cero sin que se señalice una alarma.

Parámetros de transferencia de ciclos estándar y de medida

Para los parámetros de transferencia de ciclos estándar y de medida rigen unos rangos de valores definidos.

En valores angulares, el rango de valores está definido como sigue:

- Giro alrededor del 1er eje geométrico: -180 grados a +180 grados
- Giro alrededor del 2º eje geométrico: -90 grados a +90 grados
- Giro alrededor del 3.er eje geométrico: -180 grados a +180 grados

Ver capítulo Frames, "Rotación programable (ROT, AROT, RPL)".

Nota

En la transferencia de valores angulares a un ciclo estándar o de medida se deberá observar lo siguiente:

¡Los valores menores a la precisión de cálculo del CN se tienen que redondear a cero!

La precisión de cálculo del CN para posiciones angulares está definida en el dato de máquina:

DM10210 $MN_INT_INCR_PER_DEG

3.13.10 Modificar los portaherramientas orientables después de la medición de la máquina (CORRTC)

La función `CORRTC` permite escribir elementos de cadena cinemática medidos de un ToolCarrier en elementos de corrección especiales.

Nota

Los valores de corrección escritos con la función CORRTC no son efectivos de inmediato en el ToolCarrier. Los valores de corrección solo se aplican tras deseleccionar el ToolCarrier, NEWCONF y seleccionar el ToolCarrier.

Sintaxis

```
<_Corr_Status> = CORRTC(<_Corr_Vect>, <_Corr_Index>, <_Corr_Mode>,
[ <_No_Alarm>])
```

Descripción

CORRTC:	Llamada de función		
<_Corr_Status>:	Valor de retorno de la función		
	Tipo de dato:	INT	
	Valores:	0	La función se ha ejecutado sin errores.
		1	No hay ningún ToolCarrier activo.
		2	El ToolCarrier activo no se ha definido con cadenas cinemáticas.
		10	El parámetro de llamada <_Corr_Index> es negativo.
		11	El parámetro de llamada <_Corr_Mode> es negativo.
		12	Referencia no válida a una sección de una subcadena (_CORR_INDEX).
		13	En la sección a la que se hace referencia con el parámetro _CORR_INDEX no se ha definido ningún elemento de corrección ($TC_CARR_CORR_ELEM).
		20	El dígito de centenas de _CORR_MODE no es válido. Solo se permiten los valores 0 y 1.
		21	El dígito de millares de _CORR_MODE no es válido. Solo se permiten los valores 0 y 1.
		30	En la corrección de un vector de offset, la desviación respecto a los valores actuales es mayor que el valor máximo especificado por el dato de operador SD41612 $SN_CORR_TOCARR_LIN_MAX en al menos una coordenada.
		31	Se ha rechazado el intento de escribir una variable de sistema debido a la falta de derechos de escritura.
<_Corr_Vect>:	Vector de corrección El contenido del vector de corrección se define mediante los parámetros siguientes <Corr_Index> y <Corr_Mode>. Si <_Corr_Status> = 30, se sobrescribe el contenido del vector (ver arriba).		
	Tipo de dato:	REAL	
<_Corr_Index>:	Designa la sección para la que se debe corregir el vector de dirección del elemento de corrección.		
	Tipo de dato:	INT	

<_Corr_Mode>:	Modo de corrección		
	Tipo de dato:	INT	
	El parámetro <Corr_Mode> está codificado en decimal (dígito de unidades a millares):		
	Dígito de unidades:	Reservado	
	Dígito de decenas:	Determina cómo debe modificarse el elemento de corrección al que hace referencia el contenido de <_Corr_Index>.	
		xx**0**x	El vector de corrección se escribe inmediatamente en el elemento de corrección. Esta variante puede utilizarse para escribir inmediatamente el elemento de corrección sin que tenga que conocerse el índice <n> del dato del sistema en cuestión ($NK_OFF_DIR[<n>, ...]).
		xx**1**x	Como 0, pero con la diferencia de que el valor de corrección transferido se interpreta en coordenadas mundiales. Siempre puede surgir una diferencia entre las variantes 0 y 1 si la cadena cinemática incluye giros adicionales en la posición básica (posiciones de todos los ejes giratorios iguales a 0).
		xx**2**x	Como 1, pero con la diferencia de que el valor de corrección hace referencia a la sección completa, es decir, en el elemento de corrección se ha introducido un valor con el que la sección completa alcanza la longitud definida a través del valor de corrección.
	Dígito de centenas:	Determina cómo hay que interpretar el contenido del parámetro <_Corr_Vect>.	
		x**0**xx	El vector de corrección transferido <_Corr_Vect> contiene la nueva longitud completa del elemento de corrección o de la sección a los que hace referencia <_Corr_Index> en combinación con el dígito de decenas de <_Corr_Mode> (corrección absoluta).
		x**1**xx	El vector de corrección transferido <_Corr_Vect> contiene solo la diferencia respecto a la longitud actual del elemento de corrección o de la sección a los que hace referencia <_Corr_Index> en combinación con el dígito de decenas de <_Corr_Mode> (corrección incremental).
	Dígito de millares:	Determina si debe limitarse o no la máxima corrección admisible con el dato de operador $SN_ CORR_TOCARR_LIN_MAX.	
		0xxx	La vigilancia de valor límite está activa.
		1xxx	Se suprime la vigilancia de valor límite.
<_No_Alarm>:	Respuesta en caso de error (valor de retorno >0) (**opcional**)		
	Tipo de dato:	BOOL	
	Valor:	FALSE (predeterminado)	En caso de fallo se detiene la ejecución del programa y se emite una alarma.
		TRUE	En caso de error no se detiene la ejecución del programa ni se emite alarma alguna. Caso de aplicación: reacción específica de usuario, conforme al valor de retorno

Más información sobre CORRTC

La estructura cinemática de un ToolCarrier se describe por medio de una cadena cinemática (tipo T y tipo P) o dos cadenas cinemáticas (tipo M) (subcadenas) que parten del correspondiente punto de referencia de la máquina o del portaherramientas). Una de las dos cadenas, la cadena de herramientas, finaliza en el punto de referencia de la herramienta; la otra, la cadena de piezas, en el origen del sistema de coordenadas básico.

La función CORRTC escribe direcciones de eje en elementos de corrección especiales en máquinas con ToolCarrier. Una cadena cinemática se describe, entre otros, con elementos del tipo OFFSET, que se definen mediante $NK_TYPE.

CORRTC trabaja con secciones

Las dos subcadenas se pueden dividir cada una en cuatro secciones como máximo:

- La sección 1 empieza en el punto inicial de la cadena y termina en el primer eje giratorio.
- La sección 2 es la que hay entre el eje giratorio 1 y el eje giratorio 2.
- La sección 3 es la que hay entre el eje giratorio 2 y el final de la cadena.

El gráfico siguiente muestra un portaherramientas orientable con 2 ejes giratorios.

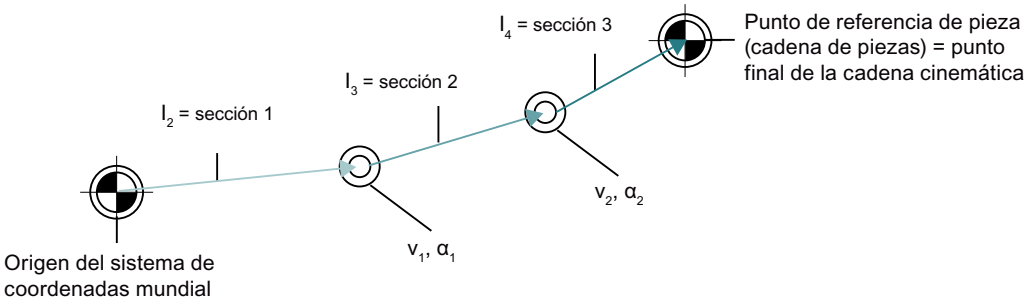


Figura 3-11 Ejemplo de CORRTC

Las secciones se definen de forma unívoca: Cuando se recorre la subcadena cinemática desde el punto inicial hasta el punto final, la primera sección tiene el índice 1, la siguiente el índice 2 y así sucesivamente.

Elementos de corrección

Se puede hacer referencia un elemento de cadena cinemática constante (elemento de cadena del tipo $NK_TYPE[<n>] = "OFFSET") en cada una de estas secciones con la variable de sistema $TC_CARR_CORR_ELEM [, 0 ... 3]. En estos elementos denominados de este modo se introducen por medio de la función CORRTC los valores de corrección determinados en la medición de máquina.

El orden de las referencias en $TC_CARR_CORR_ELEM[m, 0..3] debe corresponderse con las secciones descritas anteriormente, es decir, en $TC_CARR_CORR_ELEM[m, 0] solo puede haber un elemento de cadena perteneciente al vector de offset l1, y así sucesivamente.

El valor de referencia es siempre el valor que actúa en el ToolCarrier activo cuando se llama CORRTC. Los contenidos de datos cinemáticos modificados en la gestión de datos después de seleccionar el ToolCarrier no afectan a la aplicación de la función CORRTC.

3.13.11 Corrección longitudinal de herramienta online (TOFFON, TOFFOF)

A través de la variable de sistema $AA_TOFF[<n>], las longitudes de herramienta efectivas según las tres direcciones de herramienta se pueden superponer de forma tridimensional en tiempo real.

Cómo índice <n> se utilizan los tres identificadores de eje geométrico. De este modo, se define el número de direcciones de corrección a través de los ejes geométricos activos al mismo tiempo.

Todas las correcciones pueden estar activas a la vez.

La función Corrección de longitud de herramienta online se puede utilizar en:

- Transformada de orientación TRAORI
- Portaherramientas orientable TCARR

Nota

La corrección de longitud de herramienta online es una **opción** que se tiene que desbloquear previamente. Esta función sólo tiene sentido en combinación con una transformada de orientación activa o un portaherramientas orientable activo.

Sintaxis

```
TRAORI
TOFFON(<Sentido de corrección>[,<Valor de corrección>])
WHEN TRUE DO $AA_TOFF[<Sentido de corrección>]          ; En acciones síncronas.
...
TOFFOF(<Sentido de corrección>)
```

Descripción

TOFFON:	**Activación** de la corrección longitudinal de herramienta online	
	<Sentido de corrección>:	Dirección de herramienta (X, Y, Z) en la que debe ser efectiva la corrección longitudinal de herramienta online.
	<Valor de corrección>:	En la activación se puede indicar, para el correspondiente sentido de corrección, un valor de corrección que se aplica inmediatamente.
TOFFOF:	**Desactivación** de la corrección longitudinal de herramienta online Los valores de corrección en el sentido de corrección indicado se ponen a cero y se inicia una parada de decodificación previa.	

Ejemplos

Ejemplo 1: selección de la corrección longitudinal de herramienta

Código de programa	Comentarios
MD21190 $MC_TOFF_MODE =1	; Se posiciona en valores absolutos.
MD21194 $MC_TOFF_VELO[0] =1000	
MD21196 $MC_TOFF_VELO[1] =1000	
MD21194 $MC_TOFF_VELO[2] =1000	
MD21196 $MC_TOFF_ACCEL[0] =1	
MD21196 $MC_TOFF_ACCEL[1] =1	
MD21196 $MC_TOFF_ACCEL[2] =1	
N5 DEF REAL XOFFSET	
N10 TRAORI(1)	; Transformada activada.
N20 TOFFON(Z)	; Activación de la corrección longitudinal de herramienta online para la dirección de herramienta Z.
N30 WHEN TRUE DO $AA_TOFF[Z]=10 G4 F5	; Para la dirección de herramienta Z se interpola una corrección longitudinal de herramienta de 10.
...	
N100 XOFFSET=$AA_TOFF_VAL[X]	; Asignar corrección actual en dirección X.
N120 TOFFON(X,-XOFFSET) G4 F5	; Para la dirección de herramienta X, la corrección longitudinal de herramienta vuelve a 0.

Ejemplo 2: cancelación de la corrección de la longitud de la herramienta

Código de programa	Comentarios
N10 TRAORI(1)	; Transformada activada.
N20 TOFFON(X)	; Activación de la corrección longitudinal de herramienta online para la dirección de herramienta X.
N30 WHEN TRUE DO $AA_TOFF[X] = 10 G4 F5	; Para la dirección de herramienta X se interpola una corrección longitudinal de herramienta de 10.
...	
N80 TOFFOF(X)	; El decalaje de posición de la dirección de herramienta X se borra: ...$AA_TOFF[X]=0 No se desplaza ningún eje. A la posición actual en el WKS se suma el decalaje de posición conforme a la orientación actual.

Información adicional

Preparación de secuencias

En la preparación de secuencias en el preprocesamiento se considera también la corrección de longitud de herramienta actual que se encuentra activa en la marcha principal. Para poder aprovechar al máximo las máximas velocidades de eje admisibles, es necesario detener la preparación de secuencias con una parada de decodificación previa STOPRE mientras se establece una corrección de herramienta.

En el momento del preprocesamiento, la corrección de herramienta también se conoce siempre cuando las correcciones de longitud de herramienta ya no se modifican después del inicio del programa, o si, después de una modificación de las correcciones de longitud de herramienta, se han ejecutado más secuencias de las que puede alojar la memoria IPO entre el preprocesamiento y la marcha principal.

Variable $AA_TOFF_PREP_DIFF

La magnitud de la diferencia entre la corrección actual, activa en el interpolador, y la corrección activa en el momento de la preparación de la secuencia se puede consultar en las variables $AA_TOFF_PREP_DIFF[<n>].

Ajuste de datos de máquina y datos de operador

Para la corrección de longitud de herramienta online se dispone de los siguientes datos de sistema:

- MD20610 $MC_ADD_MOVE_ACCEL_RESERVE (Reserva de aceleración para movimiento superpuesto)
- MD21190 $MC_TOFF_MODE
  El contenido de la variable de sistema $AA_TOFF[<n>] se ejecuta como valor absoluto o se integra.
- MD21194 $MC_TOFF_VELO (Velocidad de la corrección de longitud de herramienta online)
- MD21196 $MC_TOFF_ACCEL (Aceleración de la corrección de longitud de herramienta online)
- Dato de operador para la definición de valores límite:
  SD42970 $SC_TOFF_LIMIT (límite superior del valor de corrección de longitud de herramienta)

3.13.12 Modificación de los datos de corrección en herramientas giratorias

3.13.12.1 Cálculo de orientaciones (ORISOLH)

La función predefinida ORISOLH ayuda al usuario a ajustar las posiciones de los ejes giratorios de una máquina, de modo que una herramienta de tornear se lleve a una posición definida relativa a la herramienta e independiente de la cinemática. El requisito es que esté activa una transformada de 6 ejes parametrizada con cadenas cinemáticas.

Hay dos funciones básicas disponibles:

- Alinear herramienta
  Se especifican los dos ángulos β y γ. La función calcula los ángulos necesarios de los tres ejes de orientación.
- Alinear herramienta directamente
  Se especifican los ángulos del segundo y del tercer eje de orientación. La función calcula a partir de estos datos los ángulos β y γ correspondientes, así como el ángulo del primer eje de orientación que falta.

Nota

Orden de los ejes de orientación

Cuando se recorre la cadena cinemática que describe la estructura de la máquina, desde la pieza hasta la herramienta, se aplica lo siguiente para el orden de los tres ejes de orientación de una transformada de 6 ejes:

- El eje de orientación más próximo a la pie**za** es el **primer** eje de orientación.
- El eje de orientación más próximo a la herra**mienta** es el **tercer** eje de orientación.

Por lo general el primer eje de orientación es un husillo, por lo que, en estos casos, el giro correspondiente se implementa con un frame giratorio.

Sintaxis

```
<RetVal> = ORISOLH(<Cntrl>,<W1>,<W2>)
```

Significado

ORISOLH:	Llamada de función		
<RetVal>:	Valor de retorno de la función		
	Tipo de dato:	INT	
	Rango de valores:	0, -2, -3, ..., -17	
	Valores:	0	La función ha finalizado sin errores.
		-2	No hay activa ninguna transformada válida (transformada de orientación de 6 ejes).
		-3	El primer parámetro (<Cntrl>) es negativo.
		-4	El dígito de unidades del primer parámetro (<Cntrl>) no es válido. Solo se permiten los valores 0 y 1.
		-5	El dígito de decenas del primer parámetro (<Cntrl>) no es válido. Solo se permiten los valores 0 a 3.
		-6	El dígito de centenas del primer parámetro (<Cntrl>) no es válido. Solo se permiten los valores 0 y 1.
		-7	El dígito de millares del primer parámetro (<Cntrl>) no es válido. Solo se permiten los valores 0 a 3.
		-8	En la función "Alinear la herramienta directamente" el ángulo γ es demasiado grande.
		-9	En la función "Alinear la herramienta directamente", al menos una de las posiciones de eje especificadas transgrede un límite de eje.
		-10	No hay ninguna herramienta activa.
		-11	La orientación requerida no puede ajustarse.
		-12	La adaptación de los ángulos de eje libres con dentado Hirth no es posible para la primera o la única solución.
		-13	La adaptación de los ángulos de eje libres con dentado Hirth no es posible para la segunda solución.
		-14	La adaptación de los ángulos de eje libres con dentado Hirth no es posible para ninguna de las dos soluciones.
		-15	El primer eje de orientación está parametrizado como eje Hirth.
		-16	Tanto el segundo como el tercer eje giratorio están parametrizados como eje Hirth. Solo uno de los dos ejes puede ser un eje Hirth.
		-17	En la función "Orientación directa", al menos una de las posiciones de eje especificadas no es compatible con el dentado Hirth correspondiente.

<Cntrl>:	Controla el comportamiento de la función		
	Tipo de dato:	INT	
	El parámetro <Cntrl> está codificado en decimal (dígito de unidades a millares):		
	Dígito de unidades:	El dígito de unidades controla el comportamiento en caso de fallo.	
		xxx**0**	En caso de fallo (valor de retorno <0) se cancela la ejecución del programa y se emite la alarma 14106. **Nota:** También se emite la alarma cuando el parámetro <Cntrl> es negativo, independientemente del valor del dígito de unidades.
		xxx**1**	En caso de fallo (valor de retorno <0) **no** se emite ninguna alarma. El usuario tiene la posibilidad de reaccionar adecuadamente en el programa.
	Dígito de decenas:	Controla el comportamiento si existe un eje de orientación con dentado Hirth. **Nota:** Este parámetro solo se evalúa con la función "Alinear herramienta" (es decir, cuando el dígito de centenas tiene el valor "0").	
		xx**0**x	La posición del eje se redondea a la posición más cercana.
		xx**1**x	Las posiciones del eje se redondean de forma que la desviación del ángulo β con respecto a su valor programado sea mínima.
		xx**2**x	Las posiciones del eje se redondean de forma que el ángulo β sea igual al máximo valor posible menor que el valor programado (se redondea β).
		xx**3**x	Las posiciones del eje se redondean de forma que el ángulo β sea igual al mínimo valor posible mayor que el valor programado (se redondea β).
	Dígito de centenas:	Indica qué función debe ejecutarse o qué significado tienen los dos parámetros siguientes <W1> y <W2>.	
		x**0**xx	Función "Alinear herramienta" Los parámetros <W1> y <W2> tienen el siguiente significado: • <W1> = β • <W2> = γ Se calculan los ángulos correspondientes de los ejes de orientación.
		x**1**xx	Función "Alinear herramienta directamente" <W1> es la especificación de posición para el segundo eje de orientación; <W2> la especificación de posición para el tercer eje de orientación de una transformada de 6 ejes. Se determina la posición del primer eje de orientación y de los ángulos β y γ compatibles con las dos especificaciones de posición. Si no se produce ningún error, siempre se emiten dos soluciones en la variable de sistema $P_ORI_POS[<n>, <m>]. El primer índice <n> (0 o 1) remite a la solución y el segundo índice <m> (0 ... 2) al eje de orientación: • $P_ORI_POS[0/1, **0**]: Posición del primer eje de orientación • $P_ORI_POS[0/1, **1**]: Ángulo β

			• $P_ORI_POS[0/1, **2**]: Ángulo γ Se comprueba si las especificaciones de posición <W1> y <W2> son compatibles con posibles dentados Hirth activos o límites de software activos. Si no es así, se devuelve un número de error correspondiente (ver parámetro <RetVal>). Cuando se selecciona cualquiera de los dos ángulos <W1> y <W2>, el filo de la herramienta no está por regla general en el plano de mecanizado. El ángulo γ alrededor del cual gira el filo en el plano de mecanizado no puede ser mayor que el valor límite determinado con el dato de operador SD42999 $SC_ORISOLH_INCLINE_TOL.
	Dígito de millares:	Indica cómo se modifican las posiciones de las soluciones en caso necesario, cuando el dígito de centenas tiene el valor "0", es decir, con la función "Alinear herramienta".	
		0xxx	Las posiciones de eje calculadas deben estar lo más cerca posible de las posiciones de ejes de máquina actuales.
		1xxx	Las posiciones de eje calculadas deben estar en los ejes módulo lo más cerca posible del centro del margen módulo; en los otros ejes, lo más cerca posible de 0. Para los ejes no módulo esto significa que las funciones de eje se reducen al rango de –180° … +180°.
		2xxx	Las posiciones de eje calculadas deben reducirse al rango de –180° … +180° independientemente del tipo de eje.
<W1>:	Primer ángulo El significado se obtiene del dígito de centenas del parámetro <Cntrl>.		
	Tipo de dato:	REAL	
<W1>:	Segundo ángulo El significado se obtiene del dígito de centenas del parámetro <Cntrl>.		
	Tipo de dato:	REAL	

Nota

Los parámetros no programados tienen el valor predeterminado "0".

Información adicional

El número de soluciones encontrada y la información adicional durante la ejecución de la función ORISOLH, pueden leerse a través de las siguientes variables de sistema:

Variable de sistema	Significado		
$P_ORI_POS [<n>, <m>]	Proporciona los ángulos de los ejes de orientación que resultan de la programación de la orientación.		
	<n>:	Índice de la solución	
		Rango de valores:	0, 1
	<m>:	Índice del eje de orientación	
		Rango de valores:	0 ... 2
			El orden de los ejes de orientación (1 ... 3) se refiere a la definición de los ejes en $NT_ROT_AX_NAME.
	Al llamar la función ORISOLH en el modo "Alinear herramienta directamente", las variables $P_ORI_POS[0/1, 1] y P_ORI_POS[0/1, 2] incorporan los valores de los ángulos β y γ pertenecientes a ambas soluciones. La primera solución introducida en $P_ORI_POS[<n>, <m>], es decir, con el índice <n> = 0, es siempre la que selecciona el control cuando arranca directamente la orientación requerida. El segundo índice <m> se refiere al eje de orientación, es decir, a $NT_ROT_AX_NAME. Las posiciones de eje introducidas en $P_ORI_POS[<n>, <m>] tienen en cuenta los offset introducidos en $NK_OFF y $NK_OFF_FINE, es decir, estos ángulos de eje pueden utilizarse en secuencias posteriores para ajustar la orientación deseada sin realizar más modificaciones. Si un eje giratorio es un eje Hirth, las posiciones de la solución se redondean a la posición de retícula más cercana del dentado Hirth. Para los ejes giratorios con dentado Hirth, en la variable de sistema $P_ORI_DIFF pueden leerse las diferencias de las posiciones de eje para las soluciones exactas y las soluciones adaptadas a los pasos Hirth.		
$P_ORI_DIFF [<n>, <m>]	Proporciona la diferencia entre las posiciones de los ejes de orientación exactas y las que se facilitan en $P_ORI_POS resultantes de la programación de la orientación.		
	<n>:	Índice de la solución	
		Rango de valores:	0, 1
	<m>:	Índice del eje de orientación	
		Rango de valores:	0 ... 2
			El orden de los ejes de orientación (1 ... 3) se refiere a la definición de los ejes en $NT_ROT_AX_NAME.
	El contenido solo puede ser distinto de cero en el caso de posiciones escalonadas (dentado Hirth), es decir, cuando el dato de sistema $NT_HIRTH_INCR del eje correspondiente es distinto de cero y se trata de un eje giratorio manual.		

Variable de sistema	Significado		
$P_ORI_SOL	Cuando en una transformada de orientación con más de un eje de orientación se calculan los ángulos de eje que producirán una orientación especificada, hay por lo general más de una solución. La variable de sistema $P_ORI_SOL contiene el número de soluciones válidas junto con información de estado adicional. El contenido de $P_ORI_SOL está codificado de la siguiente manera:		
	Valores <0	Estados de error generales	
		-1	Para la transformada activa no se ha calculado todavía ninguna solución (falta llamada de ORISOLH).
		-2	No hay ninguna transformada activa o la transformada activa no es una transformada de orientación (transformada de 6 ejes) que pueda proporcionar posiciones según una programación de orientación especificada.
		-4	La orientación deseada no se puede ajustar con la cinemática existente.
		-5	Con la llamada de la función ORISOLH en el modo "Alinear herramienta directamente" no se ha encontrado ninguna solución.
		-6	Con la llamada de la función ORISOLH en el modo "Alinear herramienta directamente" el ángulo γ es demasiado grande.
		-7	Con la llamada de la función ORISOLH en el modo "Alinear herramienta directamente" se ha especificado un ángulo que no puede ajustarse debido al dentado Hirth.
		-8	El primer eje de orientación (eje frame) no puede estar parametrizado como eje Hirth.
		-9	Tanto el segundo como el tercer eje giratorio están parametrizados como eje Hirth. Solo uno de los dos ejes puede ser un eje Hirth.
		-10	No se ha encontrado ninguna adaptación de la(s) solución(es) al dentado de eje Hirth.
	Valores >0 Dígito de unidades	Número de las soluciones posibles matemáticamente sin tener en cuenta los límites de eje o las posibles condiciones de error.	
		0	No existe ninguna solución, es decir, la orientación requerida no se puede ajustar. Este caso puede tener tres causas distintas: • La orientación requerida no se puede alcanzar básicamente debido a la cinemática de la máquina (los ejes de orientación no se encuentran en ángulo recto) en cualquier rango de desplazamiento de los ejes de orientación. En este caso, los dígitos de unidades y centenas de $P_ORI_SOL son cero; las variables de estado $P_ORI_STAT asignadas a los ejes de orientación contienen el valor "-4". • No se pueden alcanzar las soluciones calculadas, ya que infringirían las limitaciones de eje. Las posiciones de los ejes de orientación que se hubieran obtenido sin limitaciones de eje pueden leerse en $P_ORI_POS. • Con la llamada de la función ORISOLH en el modo "Alinear herramienta directamente" se han establecido posi-

Variable de sistema	Significado		
			ciones de eje tales que o bien el vector de orientación o bien el vector normal de orientación de la herramienta son paralelos al primer eje de orientación cuya posición se va a calcular. En estos casos la posición de este eje no está definida.
		1	Existe una solución. Este caso puede tener tres causas distintas: • Solo existe una solución debido a la orientación especificada y a la cinemática de la máquina incluso sin tener en cuenta los límites de eje (desde el punto de vista matemático, se trata de dos soluciones coincidentes). Este caso se da con cinemáticas no ortogonales en el borde del rango de orientación. $P_ORI_POS contiene las dos soluciones (idénticas). • Solo existe una solución porque una segunda solución no es válida por transgredir los límites de eje. La solución válida es siempre por tanto la primera solución en $P_ORI_POS. La segunda solución que se obtendría sin tener en cuenta los límites de eje también se puede leer en $P_ORI_POS. • Este es el caso normal cuando se llama la función ORISOLH en el modo "Alinear herramienta directamente". Con las posiciones de eje especificadas de dos ejes de orientación solo existe por regla general una posición válida del eje de orientación que falta por calcular.
		2	Existen dos soluciones.
		8	Existen infinitas soluciones, es decir, la posición de un eje de orientación (el eje de polo) es discrecional. No obstante, de las dos posibles posiciones del otro(s) eje(s), una se excluye debido a la transgresión de los límites de eje.
		9	Existen infinitas soluciones, es decir, la posición de un eje de orientación (el eje de polo) es indeterminada. El eje indeterminado puede obtenerse a partir del dígito de centenas o de las variables de sistema $P_ORI_STAT.
	Valores >0 Dígito de decenas	Indicación codificada en bits para límites de eje infringidos. La causa exacta de error puede determinarse a partir de las variables de sistema $P_ORI_STAT.	
		Bit 0 (valor 10):	Para al menos una solución se ha vulnerado al menos un límite de eje del 1.er eje de orientación.
		Bit 1 (valor 20):	Para al menos una solución se ha vulnerado al menos un límite de eje del 2.º eje de orientación.
		Bit 2 (valor 40):	Para al menos una solución se ha vulnerado al menos un límite de eje del 3.er eje de orientación.
	Valores >0 Dígito de centenas	Indicación codificada en bits para posiciones de eje no definidas (solo puede aparecer cuando hay soluciones infinitas, es decir, cuando el dígito de unidades es igual a "9").	
		Bit 0 (valor 100):	La posición del 1.er eje de orientación no está definida.

Variable de sistema	Significado		
		Bit 1 (valor 200):	La posición del 2.º eje de orientación no está definida.
		Bit 2 (valor 400):	La posición del 3.er eje de orientación no está definida.
	Las denominaciones de 1.er, 2.º y 3.er eje de orientación se refieren a la definición de los ejes en $NT_ROT_AX_NAME.		

Variable de sistema	Significado			
$P_ORI_STAT [<n>]	Proporciona el estado para cada uno de los 3 ejes de orientación como máximo tras la llamada de ORISOLH.			
	<n>:	Índice del eje de orientación (corresponde al índice del eje de orientación en cuestión en $NT_ROT_AX_NAME)		
		Rango de valores:	0 ... 2	
			El orden de los ejes de orientación (1 ... 3) se refiere a la definición de los ejes en $NT_ROT_AX_NAME.	
	El contenido de $P_ORI_STAT está codificado de la siguiente manera:			
	Valores <0	Estados de error generales		
		-1	El estado no está definido (falta la llamada de ORISOLH).	
		-2	No hay ninguna transformada activa o la transformada activa no es una transformada de orientación (transformada de 6 ejes) que pueda proporcionar posiciones según una programación de orientación especificada.	
		-3	El eje no está incluido en la transformada activa.	
		-4	No puede calcularse la posición del eje, ya que con la cinemática dada no puede alcanzarse la orientación requerida incluso con cualquier rango de desplazamiento del eje.	
		-5	Con la llamada de la función ORISOLH en el modo "Alinear herramienta directamente" se han establecido posiciones de eje tales que o bien el vector de orientación o bien el vector normal de orientación de la herramienta son paralelos al primer eje de orientación cuya posición se va a calcular. En estos casos la posición de este eje no está definida.	
		-6	Con la llamada de la función ORISOLH en el modo "Alinear herramienta directamente" el ángulo γ es demasiado grande.	
		-7	Con la llamada de la función ORISOLH en el modo "Alinear herramienta directamente" se ha especificado un ángulo que no puede ajustarse debido al dentado Hirth.	
		-8	El primer eje de orientación (eje frame) no puede estar parametrizado como eje Hirth.	
		-9	Tanto el segundo como el tercer eje giratorio están parametrizados como eje Hirth. Solo uno de los dos ejes puede ser un eje Hirth.	
		-10	No se ha encontrado ninguna adaptación de la(s) solución(es) al dentado de eje Hirth.	
	Valores >0 Dígito de unidades	Indicación codificada en bits para límites de eje infringidos de la primera solución.		
		Bit 0 (valor 1):	La primera solución vulnera el límite de eje inferior.	
		Bit 1 (valor 2):	La primera solución vulnera el límite de eje superior.	
	Valores >0 Dígito de decenas	Indicación codificada en bits para límites de eje infringidos de la segunda solución.		
		Bit 0 (valor 10):	La segunda solución vulnera el límite de eje inferior.	

Variable de sistema	Significado		
		Bit 1 (valor 20):	La segunda solución vulnera el límite de eje superior.
	Valores >0	Indicación de una posición de eje no definida.	
	Dígito de centenas	Bit 0 (valor 100):	La posición del eje de orientación no está definida, es decir, la orientación requerida se alcanza con cualquier ajuste del eje giratorio (posición polar). Esta información también está incluida en la variable de sistema $P_ORI_SOL.
	Pueden aparecer al mismo tiempo varios números de error que indiquen una vulneración de los límites de eje. Puesto que, en caso de vulneración de un límite de eje, se intenta alcanzar una posición dentro de los límites de eje permitidos mediante la suma o resta de múltiplos de 360°, no se define claramente - en caso de que no sea posible - si se ha vulnerado el límite de eje superior o inferior. Si no hay ninguna solución para la orientación requerida ($P_ORI_SOL = 0), el estado de los ejes de orientación incluidos en la transformada es "0".		

Nota

$NT_ROT_AX_NAME

Con esta variable de sistema se hace referencia a 3 ejes como máximo que sirven para ajustar la orientación. Incluye los nombres de los miembros de la cadena ($NK_NAME) que definen los ejes de máquina (ejes giratorios) que deben efectuar los movimientos de orientación que resultan de una transformada cinemática. El orden en el que están incluidos los hasta tres ejes giratorios en esta variable de sistema carece de importancia para la cinemática de la máquina, ya que esta se deriva de la estructura de las cadenas cinemáticas. Sin embargo, ya que define el orden en el que otras variables acceden a los ejes giratorios, el orden de los ejes de orientación en $NT_ROT_AX_NAME debe coincidir con la descripción de cinemática.

Nota

Informaciones de estado

Las informaciones de estado que indican, por ejemplo, que no se puede alcanzar una orientación o que solo podría alcanzarse vulnerando los límites de eje relevantes, no disparan una alarma de CN. Es responsabilidad del usuario reaccionar de forma adecuada a las condiciones mencionadas.

3.13.12.2 Activar la modificación de los datos de corrección en herramientas giratorias (CUTMOD, CUTMODK)

La modificación de los datos de corrección en herramientas giratorias se activa en el programa CN a través del comando CUTMOD (en combinación con portaherramientas orientables) o CUTMODK (para transformadas de orientación definidas mediante cadenas cinemáticas).

Nota

Dado que los portaherramientas orientables y las transformadas de orientación definidas con cadenas cinemáticas no pueden estar activos al mismo tiempo, no se producen conflictos entre ambas variantes.

Sintaxis

```
CUTMOD = <Value>
```

o bien

```
CUTMODK = <Command>
```

Descripción

CUTMOD:	Llamada de función en combinación con portaherramientas orientables		
<Value>:	Valor asignado		
	Tipo de dato:	INT	
	Valor:	0	La función está desactivada. Los valores suministrados por las variables del sistema $P_AD... equivalen a los parámetros de herramienta correspondientes.
		> 0	La función se activa si un portaherramientas orientable con el número indicado está activo; es decir, la activación va ligada a un portaherramientas orientable determinado. Los valores suministrados por las variables del sistema $P_AD... también se han modificado respecto a los parámetros de herramienta correspondientes en función del giro activo. La desactivación del portaherramientas orientable designado desactiva la función temporalmente; la activación de otro portaherramientas orientable la desactiva de forma permanente. En el primer caso, la función se activa de nuevo al seleccionar otra vez el mismo portaherramientas orientable; en el segundo caso, es necesaria una nueva selección, incluso si más adelante el portaherramientas orientable con el número indicado se activa de nuevo. La función no se ve afectada por el Reset.
		-1	La función siempre se activa cuando un portaherramientas orientable está activo. Al cambiar de portaherramientas o al deseleccionarlo y volverlo a seleccionar más adelante, no es necesario volver a ajustar CUTMOD.
		-2	La función siempre se activa cuando está activo un portaherramientas orientable cuyo número es igual al del portaherramientas orientable activo actualmente. Si no hay ningún portaherramientas orientable activo, esto equivale a CUTMOD=0. Si hay un portaherramientas orientable activo, éste equivale a la indicación inmediata del número de portaherramientas actual.
		< -2	Los valores inferiores a 2 no se tienen en cuenta, es decir, este caso se trata como si CUTMOD no estuviera programado. **Nota:** Este rango de valores no debe utilizarse, ya que está reservado para posibles ampliaciones posteriores.
CUTMODK:	Llamada de función en combinación con transformadas de orientación definidas mediante cadenas cinemáticas		

<Command>:	Comando asignado		
	Tipo de dato:	STRING	
	Valor:	"NEW"	Se guardan los estados relevantes para la "Modificación de los datos de corrección" de una transformada activa definida con cadenas cinemáticas, el nombre de la transformada y el frame de contorno actual. **Nota:** Este comando solo está permitido cuando está activa una transformada adecuada (TRAORI_DYN, TRAORI_STAT o TRAANG_K).
		"OFF"	Desactiva la "Modificación de los datos de corrección" activa. Se conservan los datos guardados previamente con "NEW". **Nota:** Este comando también está permitido cuando CUTMODK no está activo. Entonces no tiene efecto. Se conserva un juego de datos de la "Modificación de los datos de corrección", si lo hay.
		"ON"	Con este comando se activa de nuevo la "Modificación de los datos de corrección" con un juego de datos guardado previamente con el comando "NEW". En caso de que durante la ejecución de este comando esté activa una transformada con el nombre del juego de datos guardado, se aplica de inmediato la "Modificación de los datos de corrección". En caso contrario, se retarda la activación hasta que esté activa una transformada.
		"CLEAR"	Desactiva la "Modificación de los datos de corrección" como el comando "OFF" y además borra el juego de datos guardado. **Nota:** Este comando también está permitido cuando CUTMODK no está activo.

Nota

SD42984 $SC_CUTDIRMOD

La función activable mediante el comando CUTMOD o CUTMODK sustituye a la función activable con el dato de operador SD42984 $SC_CUTDIRMOD. Sin embargo, esta función sigue estando disponible como antes. No obstante, dado que no es conveniente utilizar ambas funciones en paralelo, solo puede activarse cuando CUTMOD es igual a cero y CUTMODK es igual a la cadena de caracteres cero.

Información adicional

Lectura de los datos de corrección modificados

Los datos de corrección modificados se pondrán a disposición en las variables del sistema y variables BTSS siguientes:

Descripción	Variable de sistema	Variable BTSS
Posición del filo	$P_AD[2]	cuttEdgeParam2
Ángulo de soporte	$P_AD[10]	cuttEdgeParam10
Sentido de corte	$P_AD[11]	cuttEdgeParam11
Ángulo de despulla	$P_AD[24]	cuttEdgeParam24

Los datos están modificados respecto a los parámetros de herramienta correspondientes ($TC_DP2[..., ...], ...) siempre que la función "Modificación de datos de corrección en herramientas giratorias" se haya activado con el comando `CUTMOD` o `CUTMODK` y se haya girado la herramienta con un portaherramientas orientable o una transformada de orientación adecuada.

Otras variables de sistema relevantes para el funcionamiento

Variable de sistema	Descripción
$P_CUTMOD_ANG / $AC_CUTMOD_ANG	Proporciona el ángulo alrededor del cual ha girado una herramienta en el plano de mecanizado activo y en el que se basa el cálculo de los datos de filo modificados en las funciones CUTMOD o CUTMODK.
$P_CUTMOD / $AC_CUTMOD	Lee el valor válido actual que se ha programado en último lugar con el comando CUTMOD (número del portaherramientas para el cual debe activarse la modificación de los datos de corrección). SI el último valor programado fue `CUTMOD=-2` (activación con el portaherramientas orientable activo actualmente), no se devuelve el valor "-2" en la variable de sistema, sino el número del portaherramientas orientable activo en el momento de la programación.
$P_CUTMODK / $AC_CUTMODK	Lee el nombre de la transformada en la que se ha creado el juego de datos válido actualmente para la "Modificación de los datos de corrección".

Variable de sistema	Descripción	
$P_CUT_INV / $AC_CUT_INV	Suministra el valor TRUE si la herramienta está girada de manera que deba invertirse el sentido de giro del cabezal. Además, en la secuencia a la que se refiere la operación de lectura en cuestión, deben cumplirse las cuatro condiciones siguientes: 1. Hay una muela o herramienta de tornear activa (tipos de herramienta 400 a 599 o SD42950 $SC_TOOL_LENGTH_TYPE = 2). 2. La modificación de los datos de corrección se ha activado con el comando `CUTMOD` o `CUTMODK`. 3. Hay activo un portaherramientas orientable o una transformada de orientación definida con cadenas cinemáticas seleccionados con el comando CUTMOD o CUTMODK. 4. El portaherramientas orientable o la transformada de orientación cinemática giran la herramienta de manera que las normales resultantes del filo de herramienta estén girados más de 90º (normalmente 180º) respecto a la posición de partida. Si al menos una de las cuatro condiciones mencionadas no se cumple, la variable emite el valor FALSE. Para herramientas cuya posición del filo no esté definida, el valor de la variable es siempre FALSE.	
$P_CUTMOD_ERR	Estado de error tras la última llamada de la función CUTMOD La función CUTMOD también puede llamarse implícitamente en el cambio de herramientas. La variable se pone a cero al hacer un reset. Con cada cambio de herramienta se resetea primero y, dado el caso, se vuelve a escribir. La variable está codificada en bits. Los bits tienen el siguiente significado:	
	Bit 0:	No hay ningún sentido de corte definido para la herramienta activa.
	Bit 1:	Los ángulos de corte (ángulo de despulla y ángulo de soporte) de la herramienta activa son los dos cero.
	Bit 2:	El ángulo de despulla de la herramienta activa tiene un valor no admisible (<0° o >180°).
	Bit 3:	El ángulo de soporte de la herramienta activa tiene un valor no admisible (<0° o >90°).
	Bit 4:	El ángulo de plaquita de la herramienta activa tiene un valor no admisible (<0° o >90°).
	Bit 5:	La combinación posición del filo - ángulo de soporte de la herramienta activa no es admisible (para las posiciones de filo 1 a 4 el ángulo de soporte debe ser ≤90°; para las posiciones de filo 5 a 8, debe ser ≥90°).
	Bit 6:	Giro inadmisible de la herramienta activa. La herramienta se ha girado ± 90° (con una tolerancia de aprox. 1°) hacia fuera del plano de mecanizado activo. Con ello, la posición del filo en el plano de mecanizado ya no está definida.
	Bit 7:	La plaquita de corte no está en el plano de mecanizado y el ángulo entre la plaquita de corte y el plano de mecanizado es mayor que el límite superior especificado con el dato de operador SD42998 $SC_CUTMOD_PLANE_TOL.
	Bit 8:	La plaquita de corte no está en el plano de mecanizado. El ángulo α es mayor que 1°. El ángulo α es el ángulo de giro en torno al eje de coordenadas que es perpendicular a los ejes giratorios de los ángulos β y γ (el eje X en G18).

$P_...: Variables de decodificación previa

$AC_...: Variables de proceso principal

Todas las variables de proceso principal pueden leerse en acciones síncronas. Un acceso de lectura del avance genera una parada de decodificación previa.

Cambio de plano

Para la determinación de una posición de filo, sentido de corte y ángulo de soporte o de despulla modificados, la referencia es el filo en el plano activo correspondiente (G17 - G19).

Sin embargo, si el dato de operador SD42940 $SC_TOOL_LENGTH_CONST (cambio de las componentes de longitud de herramienta al cambiar nivel) contiene un valor válido que no sea cero (más o menos 17, 18 o 19), su contenido determina el plano donde se tienen en cuenta las dimensiones relevantes.

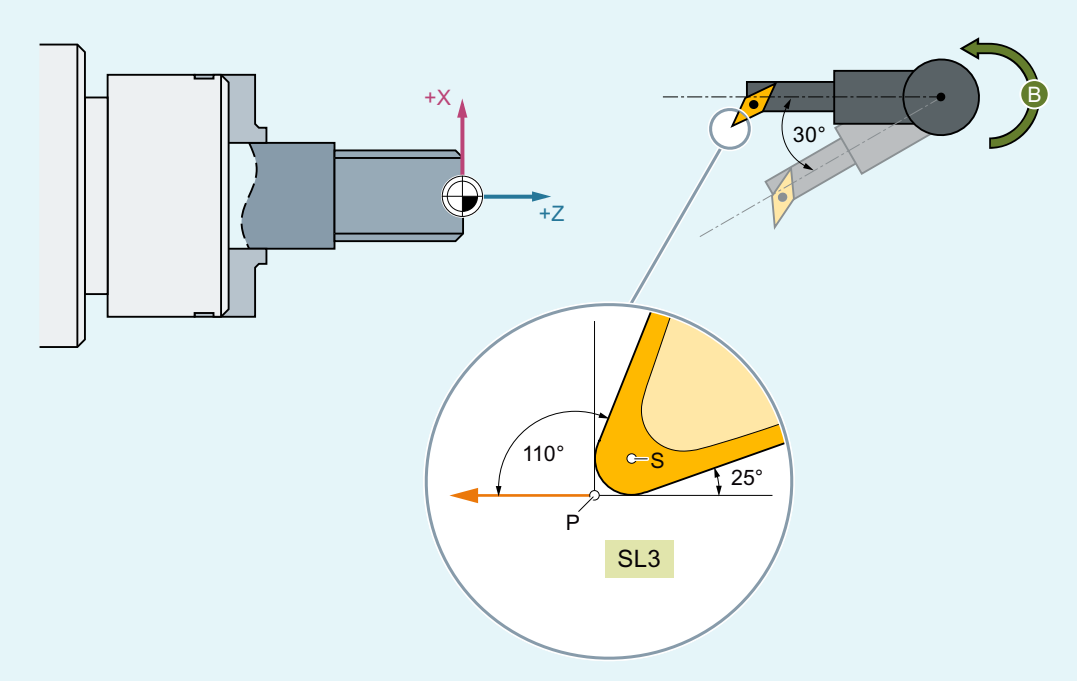
Es posible desactivar esta regla de prioridad de los datos de operador sobre el código G seteando el bit 18 del dato de máquina $MC_TOOL_PARAMETER_DEF_MASK. Esto significa que, cuando este bit está activado, se aplica siempre el plano definido con el comando G del grupo 6.

Eficacia de los datos de filos modificados

La posición de filo modificada y el punto de referencia del filo modificado serán efectivos de inmediato tras la programación, incluso para una herramienta ya activa. Para ello no se precisa una nueva selección de herramienta.

Ejemplo

Con una herramienta con la posición de filo 3 y un portaherramientas orientable que puede girar la herramienta alrededor del eje B, se debe modificar la posición de filo tras un giro de la herramienta con ayuda del comando CUTMOD.



S: Centro del filo
P: Punto de referencia del filo
SL: Posición del filo

Código de programa	Comentarios
N10 $TC_DP1[1,1]=500	
N20 $TC_DP2[1,1]=3	; Posición del filo
N30 $TC_DP3[1,1]=12	
N40 $TC_DP4[1,1]=1	
N50 $TC_DP6[1,1]=6	
N60 $TC_DP10[1,1]=110	; Ángulo de soporte
N70 $TC_DP11[1,1]=3	; Sentido de corte
N80 $TC_DP24[1,1]=25	; Ángulo de despulla
N90 $TC_CARR7[2]=0 $TC_CARR8[2]=1 $TC_CARR9[2]=0	; Eje B
N100 $TC_CARR10[2]=0 $TC_CARR11[2]=0 $TC_CARR12[2]=1	; Eje 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"	

Código de programa	Comentarios
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	

Los valores numéricos de los comentarios indican las distintas posiciones de final de secuencia en coordenadas de máquina (MKS) en el orden X → Y → Z.

Explicación

En la secuencia `N180` se selecciona en primer lugar la herramienta en `CUTMOD=0` y en un portaherramientas orientable sin girar. Dado que todos los vectores offset del portaherramientas orientable son 0, se alcanza la posición que corresponde a la longitud de la herramienta indicada en `$TC_DP3[1,1]` y `$TC_DP4[1,1]`.

En la secuencia `N200` se activa el portaherramientas orientable con un giro de 30º alrededor del eje B. Dado que la posición del filo no se modifica debido a `CUTMOD=0`, el antiguo punto de referencia del filo sigue siendo la referencia. Por esta razón se alcanza la posición en la secuencia `N210`, posición que mantiene el antiguo punto de referencia del filo en el origen (es decir, que el vector (1, 12) se gira 30º en el plano Z/X).

En la secuencia `N260`, a diferencia de la secuencia `N200 CUTMOD=2`, es efectivo. Debido al giro del portaherramientas orientable, la posición del filo modificada es 8. De ello también resultan posiciones de eje divergentes.

En las secuencias `N220` o `N270` se activa respectivamente la corrección de radio de herramienta (WRK). La posición del filo diferente en ambas partes de programa no tiene ninguna influencia sobre las posiciones finales de las secuencias donde la WRK está activa; por lo tanto, las posiciones correspondientes son idénticas. De nuevo en las secuencias de desactivación `N260` o `N300` vuelven a tener efecto las posiciones del filo diferentes.

3.13.13 Trabajar con entornos de herramientas

Resumen de funciones

- Guardar entorno de herramienta (TOOLENV) (Página 835)
- Borrar entorno de herramienta (DELTOOLENV) (Página 838)
- Leer números T, D y DL (GETTENV) (Página 839)
- Lectura de longitudes o componentes longitudinales de herramienta (GETTCOR) (Página 840)
- Modificar componentes de herramienta (SETTCOR) (Página 847)

Resumen de variables del sistema

- Lectura de la información sobre los entornos de herramienta guardados ($P_TOOLENVN, ($P_TOOLENV) (Página 840)

3.13.13.1 Guardar entorno de herramienta (TOOLENV)

La función TOOLENV sirve para guardar todos los estados actuales relevantes para la evaluación de los datos de herramienta guardados en la memoria

Son los datos siguientes:

- El comando G activo del grupo:
  - 6 (G17, G18, G19)
  - 56 (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)
- El eje de refrentado activo
- Los datos de máquina:
  - MD18112 $MN_MM_KIND_OF_SUMCORR (propiedades de las correcciones aditivas en el campo TO)
  - MD20360 $MC_TOOL_PARAMETER_DEF_MASK (definición de los parámetros de herramienta)

- Los datos de operador:
  - SD42900 $SC_MIRROR_TOOL_LENGTH (cambio de signo de longitud de herramienta en la simetría)
  - SD42910 $SC_MIRROR_TOOL_WEAR (cambio de signo en simetría del desgaste de herramienta)
  - SD42920 $SC_WEAR_SIGN_CUTPOS (signo del desgaste en herramientas con sistemas de filo)
  - SD42930 $SC_WEAR_SIGN (signo del desgaste)
  - SD42935 $SC_WEAR_TRANSFORM (transformadas para componentes de herramienta)
  - SD42940 $SC_LENGTH_CONST (cambio de componentes longitudinales de herramienta en cambio de plano)
  - SD42942 $SC_TOOL_LENGTH_CONST_T (cambio de componentes longitudinales de herramientas de torneado en cambio de plano)
  - SD42950 $SC_TOOL_LENGTH_TYPE (asignación de los componentes longitudinales de herramienta independientemente del tipo de herramienta)
  - SD42954 $SC_TOOL_ORI_CONST_M (cambio de componentes de orientación de las fresas en cambio de plano)
  - SD42956 $SC_TOOL_ORI_CONST_T (cambio de componentes de orientación de herramientas de torneado en cambio de plano)
- El componente de orientación del frame total actual (giros y simetrías, sin decalajes de origen ni escalas)
- El componente de orientación y la longitud resultante del portaherramientas orientable activo
- El componente de orientación y la longitud resultante de una transformada activa

Además de los datos mencionados que describen el entorno de la herramienta, también se guardan el número T, el número D y el número DL de la herramienta activa, para que se pueda acceder posteriormente a esta herramienta en el mismo entorno que en la llamada a TOOLENV, sin necesidad de volver a designar la herramienta.

Sintaxis

```
<Estado> = TOOLENV(<Nombre>)
```

Descripción

`TOOLENV(...)`:	Función predefinida para guardar un entorno de herramienta	
	Único elemento de la secuencia:	sí

<Estado>:	Valor de retorno de la función. Los valores negativos indican estados de error.		
	Tipo de dato:		INT
	Valor:	0	Función OK
		-1	No hay reservado ningún espacio de memoria para entornos de herramienta: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Es decir, no existe la funcionalidad "Entornos de herramienta".
		-2	Ya no hay ningún espacio disponible de memoria para entornos de herramienta.
		-3	No es admisible la cadena de caracteres cero como nombre de un entorno de herramienta.
		-4	Ningún parámetro (<Nombre>) indicado.
Parámetros			
1 <Nombre>:	Nombre con el que debe guardarse el juego de datos actual Si ya existe un juego de datos con ese nombre, se sobrescribe. En este caso, el estado es "0".		
	Tipo de dato:		STRING

Información adicional

Acotado básico/adaptador - Corrección longitudinal de herramienta

La decisión de si la longitud de herramienta o el acotado básico (parámetros específicos de corte $TC_DP21, $TC_DP22 y $TC_DP23) se incluyen en el cálculo de la longitud de herramienta, estando activa la gestión de almacén de herramientas (¡disponible solo con la opción "Gestión de herramienta"!), corresponde al valor del dato de máquina:

MD18104 $MN_MM_NUM_TOOL_ADAPTER (adaptador de herramienta en campo TO)

Un cambio de este dato de máquina solo puede ser efectivo con el arranque del control, por lo que no está guardado en el entorno de herramienta.

Longitud resultante de los portaherramientas orientables y las transformadas

Nota

En los portaherramientas orientables y en las transformadas hay variables de sistema o datos de máquina que actúan como componentes longitudinales de herramienta adicionales, y que están sujetos total o parcialmente a los giros ejecutables por ellos. Las partes longitudinales de herramienta adicionales que resultan de ello deben guardarse en la llamada a TOOLENV, ya que forman una parte del entorno en el que se utiliza la herramienta.

Transformación del adaptador

La transformación del adaptador en una propiedad del adaptador de herramienta y con ello de la herramienta completa. Por tanto, no es un componente de un entorno de herramienta que pueda utilizarse en otra herramienta.

Si se guardan todos los datos necesarios para determinar la longitud total de herramienta, se puede calcular la longitud efectiva de una herramienta en un momento posterior, incluso si esta ya no estuviera activa o si las condiciones del entorno fueran distintas (p. ej., comandos G o

datos de operador). Asimismo, puede calcularse la longitud efectiva de otra herramienta con la premisa de que será utilizada en las mismas condiciones que la herramienta para la que se ha guardado el estado.

Número máximo de juegos de datos para entornos de herramienta

Con el dato de máquina MD18116 $MN_MM_NUM_TOOL_ENV se determina cuántos juegos de datos para la descripción de entornos de herramienta pueden guardarse como máximo. Los datos se encuentran en el campo TOA. Estos se conservan aunque se desconecte el control.

No es posible una copia de seguridad. Esto significa que estos datos no pueden transmitirse entre controles distintos.

3.13.13.2 Borrar entorno de herramienta (DELTOOLENV)

Con la función DELTOOLENV se pueden borrar juegos de datos para la descripción de entornos de herramienta. Borrar significa que ya no se tendrá acceso al juego de datos guardado con un nombre determinado (si se intenta acceder se emite una alarma).

Nota

Los juegos de datos solo pueden borrarse con la función DELTOOLENV, mediante una descarga INITIAL.INI o con un arranque en frío (arranque del CN con datos de máquina estándar). No existen otros procesos automáticos de borrado.

Sintaxis

```
<Estado> = DELTOOLENV(<Nombre>)
<Estado> = DELTOOLENV()
```

Significado

DELTOOLENV(...):		Función predefinida para borrar un entorno de herramienta		
		Único elemento de la secuencia:		sí
<Estado>:		Valor de retorno de la función. Los valores negativos indican estados de error.		
		Tipo de dato:		INT
		Valor:	0	Función OK
			-1	No hay reservado ningún espacio de memoria para entornos de herramienta: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Es decir, no existe la funcionalidad "Entornos de herramienta".
			-2	No existe un entorno de herramienta con el nombre indicado.
Parámetros				
1	<Nombre>:	Nombre del juego de datos que se quiere borrar		
		Tipo de dato:		STRING

DELTOOLENV():	DELTOOLENV() sin especificar un nombre borra todos los juegos de datos para la descripción de entornos de herramienta

3.13.13.3 Leer números T, D y DL (GETTENV)

La función GETTENV sirve para leer los números T, D y DL guardados en un entorno de herramienta.

Sintaxis

```
<Estado> = GETTENV(<Nombre>, <TDDL>)
```

Significado

GETTENV(...):	Función predefinida para leer los números T, D y DL en un juego de datos para la descripción de un entorno de herramienta		
	Único elemento de la secuencia:		sí
<Estado>:	Valor de retorno de la función. Los valores negativos indican estados de error.		
	Tipo de dato:		INT
	Valor:	0	Función OK
		-1	No hay reservado ningún espacio de memoria para entornos de herramienta: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Es decir, no existe la funcionalidad "Entornos de herramienta".
		-2	No existe un entorno de herramienta con el nombre indicado.
Parámetros			
1 <Nombre>:	Nombre del juego datos en el que deben leerse los números T, D y DL		
	Tipo de dato:		STRING
2 <TDDL>:	El campo de este parámetro de resultados incluye los números T, D y DL de la herramienta cuyo entorno está guardado en el juego de datos especificado: • <TDDL> [0]: Número T • <TDDL> [1]: Número D • <TDDL> [2]: Número DL		
	Tipo de dato:		INT[3]
GETTENV(,<TDDL>), GETTENV("",<TDDL>):	Cuando se llama la función GETTENV se puede prescindir del primer parámetro o transferir la cadena de caracteres cero como primer parámetro. En ambos casos especiales se devuelven a <TDDL> los números T, D y DL de la herramienta **activa**.		

3.13.13.4 Lectura de la información sobre los entornos de herramienta guardados ($P_TOOLENVN, ($P_TOOLENV)

La información sobre los entornos de herramienta guardados se puede leer mediante las siguientes variables de sistema:

`$P_TOOLENVN`:	Proporciona el número de los juegos de datos definidos mediante TOOLENV (y todavía no borrados) para la descripción de entornos de herramienta			
	Sintaxis:	`<n> = $P_TOOLENVN`		
	Significado:	`<n>`:	Número de juegos de datos definidos	
			Tipo de dato:	INT
			Rango de valores:	0 ... MD18116 $MN_MM_NUM_TOOL_ENV
	También se puede acceder a esta variable de sistema aunque no sea posible ningún entorno de herramienta (MD18116 = 0). En tal caso, el valor de retorno es "0".			
`$P_TOOLENV`:	Proporciona el nombre del juego de datos <i>-ésimo para la descripción de un entorno de herramienta			
	Sintaxis:	`<Nombre> = $P_TOOLENV[<i>]`		
	Significado:	`<Nombre>`:	Nombre del juego de datos con el número <i>	
			Tipo de dato:	STRING
		`<i>`:	Número del juego de datos	
			Tipo de dato:	INT
			Rango de valores:	1 ... $P_TOOLENVN
	La asignación de los números a los juegos de datos no es fija, sino que puede cambiarse borrando o creando nuevos juegos de datos. La numeración de los juegos de datos se efectúa internamente. Cuando <i> remite a un juego de datos no definido, se devuelve la cadena de caracteres cero. Si el índice <i> no es válido, es decir, si <i> es inferior a 1 o mayor que el número máximo de juegos de datos para los entornos de herramienta (MD18116 $MN_MM_NUM_TOOLENV), se emite la siguiente alarma: Alarma 17020 "Índice de array 1 no autorizado"			

3.13.13.5 Lectura de longitudes o componentes longitudinales de herramienta (GETTCOR)

La función GETTCOR sirve para leer las longitudes o componentes longitudinales de herramienta.

Puede indicarse mediante los parámetros correspondientes los componentes que deben tenerse en cuenta y las condiciones de uso que deben considerarse para la herramienta.

Sintaxis

```
<Estado> = GETTCOR(<Lon>[, <Comp>, <Est>, <T>, <D>, <DL>])
```

Significado

GETTCOR(...):	Función predefinida para leer longitudes y componentes longitudinales de herramienta		
	Único elemento de la secuencia:		sí
<Estado>:	Valor de retorno de la función. Los valores negativos indican estados de error.		
	Tipo de dato:		INT
	Valor:	0	Función OK
		-1	No hay reservado ningún espacio de memoria para entornos de herramienta: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Es decir, no existe la funcionalidad "Entornos de herramienta".
		-2	No existe un entorno de herramienta con el nombre indicado en <Est>.
		-3	Cadena de caracteres no válida en el parámetro <Comp>. La causa de este error pueden ser caracteres no válidos o programados por duplicado.
		-4	Número T no válido
		-5	Número D no válido
		-6	Número DL no válido
		-7	Intento de acceso a un módulo de memoria inexistente.
		-8	Intento de acceso a una opción inexistente (p. ej., orientación programable de herramienta, gestión de herramienta).
		-9	La cadena <Comp> incluye un carácter de dos puntos (identificador para las especificaciones de un sistema de coordenadas); sin embargo, a continuación no sigue ningún carácter válido para la designación del sistema de coordenadas exigido.
Parámetros			

1	<Len>:	Vector de resultado	
		Tipo de dato:	REAL[11]
		Los componentes de vector están dispuestos en el orden siguiente: • <Lon> [0]: Tipo de herramienta • <Lon> [1]: Posición del filo • <Lon> [2]: Abscisa • <Lon> [3]: Ordenada • <Lon> [4]: Eje de aplicadas • <Lon> [5]: Radio de la herramienta Como sistema de coordenadas de referencia para los componentes longitudinales sirve el sistema de coordenadas definido en <Comp> y <Est>. Si no hay ningún sistema de coordenadas definido en <Comp>, se representan las longitudes de herramienta en el sistema de coordenadas de máquina. La asignación de abscisa, ordenada y eje de aplicadas a los ejes geométricos depende del plano activo en el entorno de herramienta utilizado. Es decir, con G17 la abscisa es paralela a X, con G18 a Z, etc. Los componentes <Lon>[6] a <Lon>[10] incluyen los parámetros adicionales que pueden especificarse para la descripción geométrica de una herramienta (p. ej., $TC_DP7 a $TC_DP11 para la geometría o los componentes correspondientes para el desgaste o correcciones aditivas o de preparación). Estos 5 elementos adicionales y el radio de la herramienta están definidos solo para los componentes E, G, S y W. Su evaluación no depende de <Est>. De este modo, los valores correspondientes en <Lon>[6] a <Lon>[10] solo pueden ser distintos a cero si en el cálculo de longitud de herramienta participa al menos uno de los cuatro componentes mencionados. Los componentes restantes no influyen en el resultado. Las dimensiones hacen referencia al sistema básico del control (pulgadas o métrico).	

2	<Comp>:	Componentes longitudinales de herramienta (opcional)			
		Tipo de dato:	STRING		
		La cadena de caracteres está compuesta por dos cadenas parciales separadas mediante un carácter de dos puntos. Formato general: "<CadPar_1> [: <CadPar_2]"			
		<CadPar_1>:	La **primera cadena parcial** designa los componentes longitudinales de herramienta que deben tenerse en cuenta en el cálculo de longitud de herramienta. El orden de los caracteres en la cadena parcial, así como la notación (mayúscula o minúscula), son indiferentes. Entre los caracteres pueden añadirse tantos espacios en blanco o caracteres de tabulación (white spaces) como se desee. **Nota:** ¡Los caracteres de la secuencia parcial **no pueden programarse por duplicado**!		
			Carácter:	–	Signo menos (¡válido solo como primer carácter!) Se calcula la longitud total de la herramienta excepto los componentes que se especifican en la cadena siguiente.
				C	Cota del adaptador o acotado básico (aquel componente de los dos existentes de forma alternativa que esté activo para la herramienta utilizada)
				E	Correcciones de preparación
				G	Geometría
				K	Transformada cinemática (¡se evalúa solo con la transformada genérica de 3, 4 y 5 ejes!)
				S	Correcciones aditivas
				T	Portaherramientas orientable
				W	Desgaste
			Si la primera cadena parcial está vacía (excepto los espacios en blanco), significa que la longitud total de la herramienta debe calcularse teniendo en cuenta todos los componentes. Lo mismo sucede cuando no se ha indicado el parámetro <Comp>.		
		<CadPar_2>:	La **segunda cadena parcial** opcional designa el sistema de coordenadas en el que debe emitirse la longitud de herramienta. La segunda cadena parcial está compuesta solamente por un único carácter relevante.		
			Carácter:	A	Sistema de coordenadas ajustable (ACS)
				B	Sistema de coordenadas básico (BKS)
				K	Sistema de coordenadas de herramienta de la transformada cinemática (KCS)
				M	Sistema de coordenadas de máquina (MKS)
				T	Sistema de coordenadas de herramienta (TCS)
				W	Sistema de coordenadas de pieza (WKS)
			Si no se indica ningún sistema de coordenadas, la evaluación se realiza en el sistema de coordenadas de máquina (MKS).		

			Los giros que tuvieran que tenerse en cuenta se determinan mediante el entorno de herramienta definido en <Est>.
3	<_Est>:	Nombre del juego de datos para la descripción de un entorno de herramienta (opcional)	
		Tipo de dato:	STRING
		Si el valor de este parámetro es la cadena de caracteres cero ("") o si no se ha indicado, se utiliza el estado actual. Si no se ha especificado ninguna herramienta, se utiliza la herramienta actual.	
4	<T>:	Número T interno de la herramienta (opcional)	
		Tipo de dato:	INT
		Si no se indica este parámetro o si su valor es "0", se utiliza la herramienta guardada en <Est>. Si el valor de este parámetro es "-1", se utiliza el número T de la herramienta activa. También se permite indicar explícitamente el número de la herramienta activa. **Nota:** Si no se indica <Est>, se utiliza el estado actual como entorno de herramienta. <T> = 0 hace referencia al número T guardado en el entorno de herramienta, por lo que se utiliza en este la herramienta activa, es decir, los datos <T> = 0 y <T> = -1 significan lo mismo en este caso especial.	
5	<D>:	Número de filo de la herramienta (opcional)	
		Tipo de dato:	INT
		Si no se indica este parámetro o si su valor es "0", el número D utilizado depende del origen del número T. Si se utiliza el número T del entorno de herramienta, también se lee el número D del entorno de herramienta; de lo contrario, el de la herramienta actualmente activa.	
6	<DL>:	Número de la corrección dependiente de la posición (opcional)	
		Tipo de dato:	INT
		Si no se indica este parámetro, el número DL utilizado depende del origen del número T. Si se utiliza el número T del entorno de herramienta, también se lee el número D del entorno de herramienta; de lo contrario, el de la herramienta actualmente activa.	

Ejemplos

`GETTCOR(_LEN)`	Calcula la longitud de herramienta de la herramienta actualmente activa en el sistema de coordenadas de máquina teniendo en cuenta todos los componentes.
`GETTCOR(_LEN,"CGW:W")`	Se calcula la longitud de herramienta a partir de la de cota del adaptador o el acotado básico, la geometría y el desgaste para la herramienta activa. No se tienen en cuenta otros componentes como, p. ej., portaherramientas orientable o transformadas cinemáticas. La salida se realiza en el sistema de coordenadas de pieza.

`GETTCOR (_LEN,"-K:B")`	Se calcula la longitud total de herramienta de la herramienta activa sin tener en cuenta los componentes longitudinales de una posible transformada cinemática activa. Salida en el sistema de coordenadas básico.
`GETTCOR (_LEN,":M","Testenv1",,3)`	Se calcula en el sistema de coordenadas de máquina la longitud total de herramienta para la herramienta guardada en el entorno de herramienta con el nombre "Testenv1". El cálculo se realiza sin embargo independientemente del número de filo guardado para el número de filo D3.

información adicional

Transformada de adaptación/portaherramientas orientable/transformada cinemática

Los giros o cambios de componentes que pudieran ser efectuados por la transformada de adaptación, el portaherramientas orientable y una transformada cinemática forman parte del entorno de la herramienta. Por esta razón siempre se ejecutan, incluso cuando no se tenga en cuenta el correspondiente componente longitudinal. En caso de que no se desee, deben definirse entornos de herramienta en los que no estén activas las transformadas correspondientes. En muchos casos (concretamente, siempre que en la máquina no se utilicen transformadas o portaherramientas orientables), los juegos de datos guardados para entornos de herramienta cumplen de forma automática estas condiciones, por lo que el usuario no necesita tenerlos en cuenta de forma específica.

Herramientas de torneado y rectificado: Cálculo de la longitud de herramienta según MD20360 $MC_TOOL_PARAMETER_DEF_MASK

La forma en la que, para herramientas de torneado y rectificado, deben evaluarse el desgaste o la longitud de herramienta en un posible eje de diámetro se determina en el dato de máquina:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK (definición de los parámetros de herramienta)

Bit	Valor	
0	En las herramientas de torneado y rectificado el **parámetro de desgaste** del eje de refrentado se calcula como valor de diámetro:	
	= 0 (predeterminado)	no
	= 1	sí
1	En las herramientas de torneado y rectificado, el **componente longitudinal de herramienta** del eje de refrentado se calcula como valor de diámetro:	
	= 0 (predeterminado)	no
	= 1	sí

Si están seteados los bits correspondientes, se valora la entrada con el factor 0,5. Esta evaluación también se proyecta en la longitud de herramienta suministrada por GETTCOR.

Ejemplo:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK = 3

MD20100 $MC_DIAMETER_AX_DEF (eje geométrico con función de eje de refrentado) = "X"

X es el eje de diámetro (configuración estándar de la máquina de torneado)

Código de programa	Comentarios
N30 $TC_DP1[1,1]=500	
N40 $TC_DP2[1,1]=2	
N50 $TC_DP3[1,1]=3.0	; Geometría L1
N60 $TC_DP4[1,1]=4.0	
N70 $TC_DP5[1,1]=5.0	
N80 $TC_DP12[1,1]=12.0	; Desgaste 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	

Componentes longitudinales de transformada cinemática y portaherramientas orientable

Si en el cálculo de la longitud de herramienta se tiene en cuenta un **portaherramientas orientable**, en el cálculo de la longitud de herramienta se introducen los siguientes vectores:

Tipo	Vectores
M	l1 e l2
T	l1, l2 e l3
P	En la longitud de herramienta no influye el portaherramientas orientable.

En la **transformada de 5 ejes** genérica, con los tipos de transformador 24 y 56 se incluyen en el cálculo de longitud de herramienta los siguientes datos de máquina:

Tipo de transformador	Datos de máquina
24	MD24550/24650 $MC_TRAFO5_BASE_TOOL_1/2 MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 MD24558/24658 $MC_TRAFO5_PART_OFFSET_1/2
56	MD24550/24650 $MC_TRAFO5_BASE_TOOL_1/2 MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2

El tipo de transformador 56 (herramienta móvil y pieza móvil) corresponde al tipo M con el portaherramientas orientable.

En esta transformada de 5 ejes, en las versiones de software anteriores, el vector MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 (vector del decalaje cinemático de la 1.ª/2.ª transformada de 5 ejes en el canal) corresponde a la suma de ambos vectores l_1 e l_3 con el portaherramientas orientable de tipo M.

Para la transformada solo es relevante en ambos casos la suma. El tipo de composición de ambos componentes individuales es irrelevante. Sin embargo, en el cálculo de la longitud de herramienta sí es significativo qué partes deben asignarse a la herramienta y qué partes a la mesa de herramienta respectivamente. Por esta razón se ha introducido el dato de máquina

MD24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 (vector de decalaje cinemático en la mesa). Corresponde al vector l3. El dato de máquina MD24560/24660 $MC_TRAFO5_JOINT_OFFSET_1/2 ya no corresponde a la suma de l1 e l3, sino solamente al vector l1. Si el dato de máquina MD24558/24658 $MC_TRAFO5_JOINT_OFFSET_PART_1/2 es cero, el comportamiento es idéntico al comportamiento anterior.

Compatibilidad

La función GETTCOR, junto con las funciones TOOLENV y SETTCOR entre otras, sirve para reemplazar parte de la funcionalidad que antes se implementaba de forma externa en los ciclos de medición.

En los ciclos de medición solo se ha evaluado una parte de los parámetros que determinan finalmente la longitud efectiva de herramienta. Las funciones mencionadas pueden parametrizarse de forma que se reproduzca el comportamiento de los ciclos de medición respecto al cálculo de la longitud de herramienta.

3.13.13.6 Modificar componentes de herramienta (SETTCOR)

La función SETTCOR sirve para modificar componentes de herramienta, teniendo en cuenta todas las condiciones que pudieran entrar en la evaluación de cada componente por separado.

Nota

Respecto a la terminología: a partir de ahora, cuando se hable de la longitud de herramienta de componentes de herramienta, se hace referencia a los componentes considerados de forma vectorial, a partir de los cuales se compone la longitud total de herramienta (p. ej., geometría o desgaste). Por tanto, un componente de este tipo consta de tres valores individuales (L1, L2, L3), que a partir de ahora se denominan valores de coordenadas.

El componente de herramienta "Geometría" está compuesto por ejemplo de tres valores de coordenadas: $TC_DP3 a $TC_DP5.

Sintaxis

```
<Estado> = SETTCOR(<CorVal>, <Comp>, [<CorComp>, <CorModo>,
<EjeGeo>, <Est>, <T>, <D>, <DL>])
```

Significado

`SETTCOR(...)`:	Función predefinida para modificar los componentes de herramienta	
	Único elemento de la secuencia:	sí

<Estado>:	Valor de retorno de la función. Los valores negativos indican estados de error.		
	Tipo de dato:		INT
	Valor:	0	Función OK
		-1	No hay reservado ningún espacio de memoria para entornos de herramienta: MD18116 $MN_MM_NUM_TOOL_ENV = 0 Es decir, no existe la funcionalidad "Entornos de herramienta".
		-2	No existe un entorno de herramienta con el nombre indicado en <Est>.
		-3	Cadena de caracteres no válida en el parámetro <Comp>. La causa de este error pueden ser caracteres no válidos o programados por duplicado.
		-4	Número T no válido.
		-5	Número D no válido.
		-6	Número DL no válido.
		-7	Intento de acceso a un módulo de memoria inexistente.
		-8	Intento de acceso a una opción inexistente (p. ej., orientación programable de herramienta, gestión de herramienta).
		-9	Valor numérico no válido para el parámetro <CorComp>.
		-10	Valor numérico no válido para el parámetro <CorModo>.
		-11	Los contenidos de los parámetros <Comp> y <CorComp> son contradictorios.
		-12	Los contenidos de los parámetros <Comp> y <CorModo> son contradictorios.
		-13	El contenido del parámetro <EjeGeo> no designa ningún eje geométrico.
		-14	Intento de escritura en una corrección de preparación inexistente.
Parámetros			
1	<CorVal>:	Vector de corrección En el sistema de coordenadas de pieza (WKS) definido mediante <Est> se aplica la siguiente asignación: • <CorVal> [0]: Abscisa • <CorVal> [1]: Ordenada • <CorVal> [2]: Eje de aplicadas Si solo debe corregirse un componente de herramienta (es decir, sin corrección vectorial, ver parámetro <CorModo>), el valor de corrección está siempre en <CorVal>[0], independientemente del eje sobre el que actúa. Entonces no se evalúa el contenido de los dos componentes restantes. Si <CorVal> o un componente de <CorVal> hacen referencia al eje de refrentado, la indicación se valora como **cota de radio**. Esto significa que una herramienta, por ejemplo, es "más larga" que la cota indicada, por lo que el diámetro de pieza duplica su tamaño. Las dimensiones hacen referencia al **sistema básico** (pulgadas o métrico) del control.	
		Tipo de dato:	REAL[3]

2	<Comp>:	Componente(s) de herramienta			
		Tipo de dato:	STRING		
		La cadena de caracteres está compuesta por dos cadenas parciales separadas mediante un carácter de dos puntos. Formato general: "<CadPar_1> [: <CadPar_2]"			
		<CadPar_1>:	Siempre debe estar presente la **primera cadena parcial**, que puede estar compuesta por uno o dos caracteres. El primer o único carácter hace referencia al 1.er componente (Val_1) y el segundo carácter, al 2.º componente (Val_2), que deben procesarse según los siguientes parámetros <CorComp> y <CorModo>.		
			Carácter:	C	Cota del adaptador o acotado básico (aquel componente de los dos existentes de forma alternativa que esté activo para la herramienta utilizada)
				E	Correcciones de preparación
				G	Geometría
				S	Correcciones aditivas
				W	Desgaste
		<Substr_2>:	La **segunda cadena parcial** es opcional. Puede estar compuesta de forma alternativa por la (única) letra "W" o "T".		
			Carácter:	W	Cuando la segunda cadena está vacía o incluye la letra "W", los valores de corrección se calculan como si hubieran sido medidos en el sistema de coordenadas de **pieza** (WKS).
				T	Cuando la segunda cadena parcial incluye la letra "T", los valores de corrección se calculan como si hubieran sido medidos en el sistema de coordenadas de **herramienta** (**T**ool Coordinate System, TCS).
		La notación de los caracteres en la cadena (mayúscula o minúscula) es indiferente. Pueden añadirse tantos espacios en blanco o caracteres de tabulación (white spaces) como se desee.			

3	<CorComp>:	Especifica los componentes del juego de datos de herramienta que deben escribirse (opcional)		
		Tipo de dato:		INT
		Valor:	0	El valor de corrección <CorVal>[0] hace referencia al eje geométrico transferido en el parámetro <EjeGeo>, en el sistema de coordenadas de pieza o en el sistema de coordenadas de herramienta (ver al respecto la descripción del parámetro <Comp>). Es decir, debe incluirse el valor de corrección en los componentes de herramienta designados de forma que, teniendo en cuenta todos los parámetros que puedan influir en el cálculo de longitud de herramienta, el resultado sea una modificación, correspondiente al valor especificado, de la longitud total de herramienta en la dirección de eje indicada. Esta modificación debe alcanzarse mediante la corrección de los componentes indicados en <Comp> y la regla de cálculo simbólica indicada en <CorModo> (ver siguiente parámetro). Por esa razón, la corrección resultante repercute en los tres componentes del eje.
			1	Como "0", pero de forma vectorial. El contenido del vector <CorVal> hace referencia a la abscisa, ordenada y eje de aplicadas en el sistema de coordenadas de pieza o en el sistema de coordenadas de herramienta (ver al respecto la descripción del parámetro <Comp>). El siguiente parámetro <EjeGeo> no se evalúa.
			2	Corrección vectorial, es decir, L1, L2 y L3 pueden modificarse simultáneamente. Al contrario que las variantes "0" y "1", los valores de corrección incluidos en <CorVal> se refieren a las coordenadas del componente Val_1 (ver siguiente parámetro <CorModo>) de la herramienta. Una posible posición inclinada de la herramienta respecto al sistema de coordenadas de pieza no influiría en la corrección.
			3 - 5	Corrección de la longitud de herramienta L1 a L3 ($TC_DP3 a $TC_DP5) o de los valores correspondientes en caso de desgaste, correcciones de preparación o sumas. El valor de corrección está incluido en <CorVal>[0]. Se ha medido en las coordenadas del componente Val_1 (ver siguiente parámetro <CorModo>) de la herramienta. Una posible posición inclinada de la herramienta respecto al sistema de coordenadas de pieza no influiría en la corrección.
			6	Corrección del radio de la herramienta ($TC_DP6) o de los valores correspondientes en caso de desgaste, correcciones de preparación o de suma. Se tienen en cuenta los bits 10 y 11 (valoración de los datos de diámetro o desgaste de diámetro opcionalmente como datos de radio o diámetro) en el dato de máquina MD20360 $MC_TOOL_PARAMETER_DEF_MASK.
			7 – 11	Corrección de $TC_DP7 a $TC_DP11 o de los valores correspondientes en caso de desgaste, correcciones de preparación o de suma. Estos parámetros se tratan como el radio de la herramienta.
		Si no se indica este parámetro, su valor es "0".		

<table>
<tr><td>4</td><td><CorModo>:</td><td colspan="3">Especifica el tipo de operación de escritura que debe ejecutarse (opcional)</td></tr>
<tr><td></td><td></td><td colspan="2">Tipo de dato:</td><td>INT</td></tr>
<tr><td></td><td></td><td rowspan="4">Valor:</td><td>0</td><td>Val_{1nuevo} = <CorVal></td></tr>
<tr><td></td><td></td><td>1</td><td>Val_{1nuevo} = $Val_{1antiguo}$ + <CorVal></td></tr>
<tr><td></td><td></td><td>2</td><td>Val_{1nuevo} = <CorVal>
Val_{2nuevo} = 0</td></tr>
<tr><td></td><td></td><td>3</td><td>Val_{1nuevo} = $Val_{1antiguo}$ + $Val_{2antiguo}$ + <CorVal>
Val_{2nuevo} = 0</td></tr>
<tr><td></td><td></td><td colspan="3">La notación $Val_{1antiguo}$ + $Val_{2antiguo}$ debe entenderse de forma simbólica. Si ambos componentes se evalúan de forma distinta (debido al estado de <_Est>), es decir, si hay un giro efectivo entre ambos componentes, $Val_{2antiguo}$ se transforma antes de la suma de modo que la longitud de herramienta resultante después de borrar Val_{2nuevo} y antes de sumar <CorVal> no varía.
<CorVal> se refiere siempre a Val_1. <CorVal> es un valor que se mide dependiendo de la segunda parte del parámetro <Comp> en el sistema de coordenadas de pieza (WKS) o en el sistema de coordenadas de herramienta (TCS). Por esa razón, dado el caso, ya está transformado respecto al componente de herramienta en el que debe incluirse. Con ello, no puede computarse inmediatamente con el valor guardado, sino que debe transformarse de nuevo antes de la suma a Val_1 o Val_2. Como consecuencia, la corrección puede repercutir en otro eje diferente al definido mediante <CorComp> o actuar en varios ejes.
En el caso de <CorComp> = 0, es decir, cuando <CorVal> no incluye ningún vector sino un valor individual, se ejecutan las operaciones descritas en las coordenadas en las que se ha medido <CorVal> (WKS/TCS). Esto se aplica especialmente para poner a cero Val_{2nuevo} en las variantes 2 y 3. Entonces el resultado se transforma de nuevo en las coordenadas de la herramienta. Como consecuencia, existe la posibilidad que no se ponga a cero ninguno de los valores de coordenadas (L1, L2, L3), o que los valores de coordenadas que antes eran cero sean ahora distintos de cero. Sin embargo, si las operaciones correspondientes se efectúan de forma sucesiva para los tres ejes geométricos, se garantiza que todos los valores de coordenadas de los componentes que deben eliminarse también serán cero. Si la herramienta no está girada respecto al sistema de coordenadas de pieza, o está girada de forma que todos los componentes de herramienta permanecen paralelos a los ejes de coordenadas (permutaciones de ejes), se garantiza que en cada caso solo se cambia una coordenada de herramienta.
La ejecución sucesiva de la misma operación (<CorModo>) con <CorComp> = 0 para los tres ejes de coordenadas en cualquier orden es idéntica a la única ejecución de la misma operación con <CorComp>=1.
Para los valores de parámetro "0" y "1", el parámetro <Comp> debe incluir un carácter, y para los valores de parámetro "2" y "3" debe incluir dos caracteres.
Ejemplo:
<Comp> incluye la cadena "ES", <CorModo> el valor "2"
⇒ Corrección de preparación$_{nueva}$ = <CorVal>, corrección de suma$_{nueva}$ = 0
Si no se indica el parámetro <CorModo>, su valor es "0".</td></tr>
</table>

5	<EjeGeo>:	Indica el índice del eje geométrico en el que se ha medido el valor de corrección <CorVal>[0] (opcional)	
		Tipo de dato:	INT
		Rango de valores:	0 ... 2
		Los índices 0 a 2 se refieren a abscisa, ordenada y eje de aplicadas en el plano efectivo (G17/G18/G19) del entorno actual de herramienta. El contenido de este parámetro se evalúa entonces solamente cuando el parámetro <CorComp> tiene el valor "0".	
6	<Est>:	Nombre del juego de datos para la descripción de un entorno de herramienta (opcional)	
		Tipo de dato:	STRING
		Si el valor de este parámetro es la cadena de caracteres cero ("") o si no se ha indicado, se utiliza el estado actual. Si no se ha especificado ninguna herramienta, se utiliza la herramienta actual.	
7	<T>:	Número T interno de la herramienta (opcional)	
		Tipo de dato:	INT
		Si no se indica este parámetro o si su valor es "0", se utiliza la herramienta guardada en <Est>. Si el valor de este parámetro es "-1", se utiliza el número T de la herramienta activa. También se permite indicar explícitamente el número de la herramienta activa. **Nota:** Si no se indica <Est>, se utiliza el estado actual como entorno de herramienta. <T> = 0 hace referencia al número T guardado en el entorno de herramienta, por lo que se utiliza en este la herramienta activa, es decir, los datos <T> = 0 y <T> = -1 significan lo mismo en este caso especial.	
8	<D>:	Número de filo de la herramienta (opcional)	
		Tipo de dato:	INT
		Si no se indica este parámetro o si su valor es "0", el número D utilizado depende del origen del número T. Si se utiliza el número T del entorno de herramienta, también se lee el número D del entorno de herramienta; de lo contrario, el de la herramienta actualmente activa.	
9	<TL>:	Número de la corrección dependiente de la posición (opcional)	
		Tipo de dato:	INT
		Si no se indica este parámetro, el número DL utilizado depende del origen del número T. Si se utiliza el número T del entorno de herramienta, también se lee el número D del entorno de herramienta; de lo contrario, el de la herramienta actualmente activa. Si T, D y DL especifican una herramienta sin correcciones dependientes de la posición, no pueden especificarse correcciones de suma o preparación en el parámetro <Comp> (código de error en <Estado>).	

Nota

No todas las combinaciones posibles de los tres parámetros <Comp>, <CorComp> y <CorModo> tienen sentido. Por ejemplo, la regla de cálculo 3 en <CorComp> necesita que se indiquen dos caracteres en <Comp>. Si se indica una combinación de parámetros inadmisible, en el <Estado> se devuelve un código de error correspondiente.

Ejemplos

Ejemplo 1

Código de programa	**Comentarios**
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fresa
N30 $TC_DP3[1,1]=10.0	; Geometría L1
N40 $TC_DP12[1,1]=1.0	; Desgaste 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	; ==> Posición MKS X0.000 Y0.000 **Z1.333**
N90 M30	

<CorComp> es "0", por esa razón se sustituye el valor de coordenadas efectivo en dirección Z de los componentes geométricos por el valor de corrección 0,333.

De esta forma, la longitud total resultante de herramienta es: L1 = 0,333 + 1,000 = 1,333

Ejemplo 2

Código de programa	**Comentarios**
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fresa
N30 $TC_DP3[1,1]=10.0	; Geometría L1
N40 $TC_DP12[1,1]=1.0	; Desgaste 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	; ==> Posición MKS X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> es "1", y por esa razón se añade el valor de coordenadas efectivo en dirección Z de 0,333 al valor de desgaste de 1,0.

De esta forma, la longitud total resultante de herramienta es: L1 = 10,0 + 1,333 = 11,333

Ejemplo 3

Código de programa	**Comentarios**
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=120	; Fresa
N30 $TC_DP3[1,1]=10.0	; Geometría L1
N40 $TC_DP12[1,1]=1.0	; Desgaste 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	; ==> Posición MKS X0,000 Y0,000 Z0,333
N90 M30	

<CorComp> es "2", y por esa razón se introduce en el componente geométrico la corrección efectiva en dirección Z (se sobrescribe el valor antiguo) y se elimina el valor de desgaste.

De esta forma, la longitud total resultante de herramienta es: L1 = 0,333 + 0,0 = 0,333

Ejemplo 4

```
Código de programa                          Comentarios
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=120                        ; Fresa
N30 $TC_DP3[1,1]=10.0                       ; Geometría L1
N40 $TC_DP12[1,1]=1.0                       ; Desgaste 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                          ; ==> Posición MKS X0.000 Y0.000 Z11.333
N90 M30
```

<CorComp> es "3", y por esa razón se añaden el valor de desgaste y el valor de corrección al componente geométrico, y se elimina el componente de desgaste.

De esta forma, la longitud total resultante de herramienta es: L1 = 11,333 + 0,0 = 11,333.

Ejemplo 5

```
Código de programa                          Comentarios
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=120                        ; Fresa
N30 $TC_DP3[1,1]=10.0                       ; Geometría L1
N40 $TC_DP12[1,1]=1.0                       ; Desgaste 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                          ; ==> Posición MKS X0,333 Y0,000 Z11,000
N90 M30
```

<CorComp> es "3" como en el ejemplo anterior; sin embargo, la corrección actúa ahora sobre el eje geométrico con el índice "0" (eje X), a la que se ha asignado en una fresa el componente de herramienta L3 debido a G17. Por esta razón, la llamada de SETTCOR no influye en los parámetros de herramienta $TC_DP3 y $TC_DP12. En su lugar se introduce el valor de corrección en $TC_DP5.

Ejemplo 6

```
Código de programa                          Comentarios
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=500                        ; Herramienta de torneado
N30 $TC_DP3[1,1]=10.0                       ; Geometría L1
N40 $TC_DP4[1,1]=15.0                       ; Geometría L2
N50 $TC_DP12[1,1]=10.0                      ; Desgaste L1
N60 $TC_DP13[1,1]=0.0                       ; Desgaste L2
```

```
Código de programa                          Comentarios
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                         ; ==> Posición MKS X24,330 Y0,000 Z17,500
N120 M30
```

La herramienta es una herramienta de torneado. En la secuencia N80 se activa un giro de frame, de forma que el sistema de coordenadas básico (BKS) esté girado respecto al sistema de coordenadas de pieza (WKS). El valor de corrección (N70) actúa en el WKS sobre el eje geométrico con el índice "1", es decir, sobre el eje X ya que G18 está activo. Puesto que <CorModo> = 3, tras la ejecución de N100 el desgaste de herramienta en dirección del eje X del WKS debe ser cero.

Por lo tanto, el contenido de los parámetros de herramienta relevantes al final del programa es:

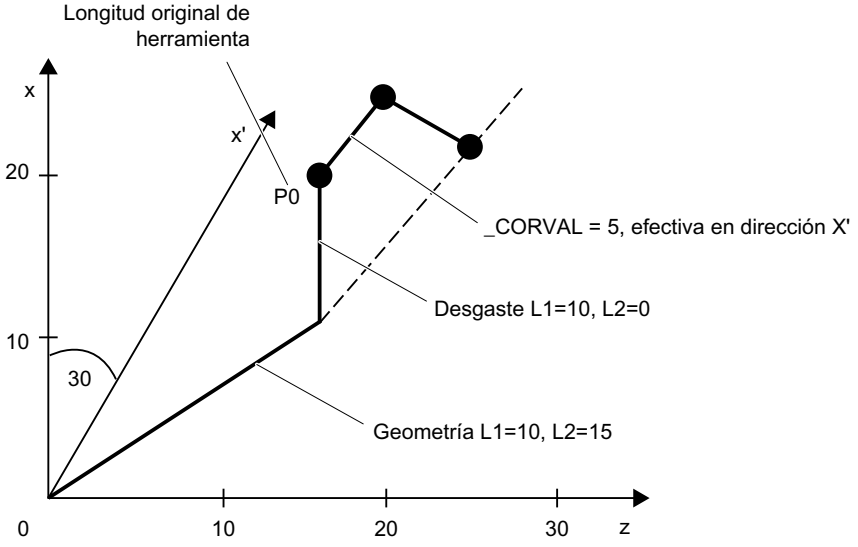
$TC_DP3[1,1] : 21.830 ; Geometría L1

$TC_DP4[1,1] : 21.830 ; Geometría L2

$TC_DP12[1,1] : 2.500 ; Desgaste L1

$TC_DP13[1,1] : -4.330 ; Desgaste L2

Los comportamientos geométricos se representan en la siguiente imagen. El desgaste total incluido _CORVAL se proyecta en dirección X' en el WKS. El resultado es el punto P2. Las coordenadas de este punto (medidas en coordenadas X-Y) se introducen en el componente geométrico de la herramienta. En el desgaste queda todavía el vector diferencial P_2 - P_1. De esta forma, el desgaste ya no tiene ningún componente en la dirección de _CORVAL.



Si se continúa el programa de ejemplo según N110 con las siguientes instrucciones, se adopta totalmente en la geometría el desgaste restante, ya que ahora la corrección actúa sobre el eje Z' (parámetro <EjeGeo> = 0):

```
N120 _CORVAL[0]=0.0
```

```
N130 R1=SETTCOR(_CORVAL,"GW",0,3,0)
N140 T1 D1 X0 Y0 Z0 ; ==> Posición MKS X24,330 Y0,000 Z17,500
```

El nuevo valor de corrección es "0", por tanto, no puede modificarse la longitud total de herramienta y, con ello, tampoco la posición alcanzada en N140. Si _CORVAL en N120 fuera distinto de "0", resultaría una nueva longitud total de herramienta y, con ello, también una posición modificada en N140; sin embargo, en cualquier caso, la parte de desgaste de la longitud de herramienta sería cero, es decir, la longitud total de herramienta siempre se incluye a continuación en la parte geométrica de la herramienta.

Llamando una vez con <CorComp> = 1 (corrección vectorial) se consigue el mismo resultado que llamando dos veces la función SETTCOR con el parámetro <CorComp> = 0:

Código de programa	**Comentarios**
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; Herramienta de torneado
N30 $TC_DP3[1,1]=10.0	; Geometría L1
N40 $TC_DP4[1,1]=15.0	; Geometría L2
N50 $TC_DP12[1,1]=10.0	; Desgaste L1
N60 $TC_DP13[1,1]=0.0	; Desgaste 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	; ==> Posición MKS X24,330 Y0,000 Z17,500
N120 M30	

En ese caso, tras la primera llamada de SETTCOR en N100, todos los componentes de desgaste de la herramienta se ponen de inmediato en cero.

Ejemplo 7

Código de programa	**Comentarios**
N10 DEF REAL _CORVAL[3]	
N20 $TC_DP1[1,1]=500	; Herramienta de torneado
N30 $TC_DP3[1,1]=10.0	; Geometría L1
N40 $TC_DP4[1,1]=15.0	; Geometría L2
N50 $TC_DP12[1,1]=10.0	; Desgaste L1
N60 $TC_DP13[1,1]=0.0	; Desgaste 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	; ==> Posición MKS X25,000 Y0,000 Z15,000
N120 M30	

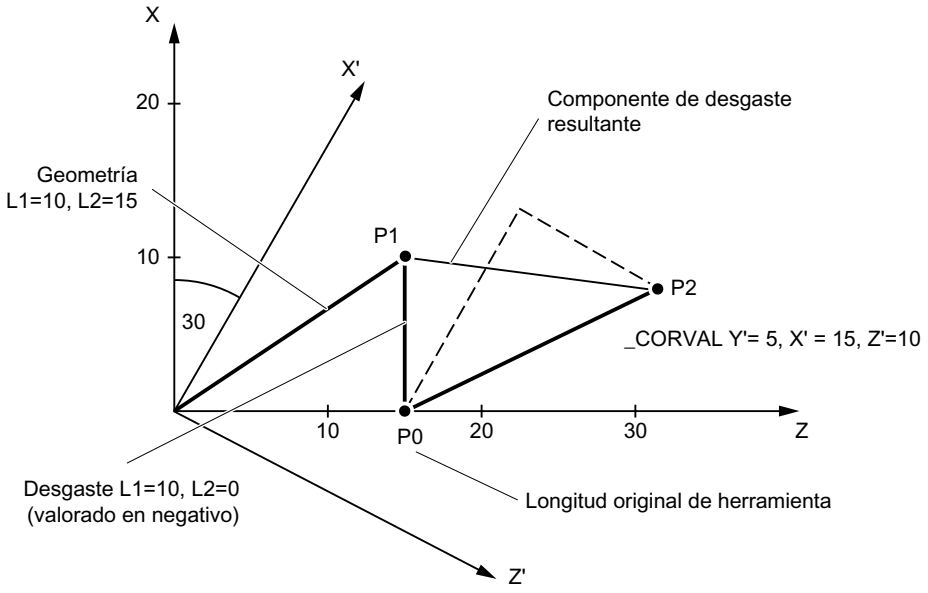
Al contrario que en el ejemplo 6, aquí el parámetro <CorComp> = 3, por tanto, puede eliminarse la indicación del parámetro <EjeGeo>. El valor incluido en _CORVAL[0] actúa

entonces de forma inmediata sobre el componente de herramienta L1, el giro en N80 no influye en el resultado, los componentes de desgaste en $TC_DP12 se aceptan junto con _CORVAL[0] en la parte geométrica, de forma que tras la primera llamada a SETTCOR en N100 la longitud total de herramienta está en la parte geométrica debido a $TC_DP13.

Ejemplo 8

```
Código de programa                          Comentarios
N10 DEF REAL _CORVAL[3]
N20 $TC_DP1[1,1]=500                        ; Herramienta de torneado
N30 $TC_DP3[1,1]=10.0                       ; Geometría L1
N40 $TC_DP4[1,1]=15.0                       ; Geometría L2
N50 $TC_DP5[1,1]=20.0                       ; Geometría L3
N60 $TC_DP12[1,1]=10.0                      ; Desgaste L1
N70 $TC_DP13[1,1]=0.0                       ; Desgaste L2
N80 $TC_DP14[1,1]=0.0                       ; Desgaste 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                         ; ==> Posición MKS X7,990 Y25,000 Z31,160
N170 M30
```

En N90 se establece el dato de operador SD42930 $SC_WEAR_SIGN, es decir, el desgaste debe valorarse con signo negativo. La corrección es vectorial (<CorComp> = 1) y el vector de corrección debe añadirse al desgaste (<CorModo> = 1). Los comportamientos geométricos en el plano Z-X se representan en la siguiente imagen:



Debido a <CorModo> = 1 permanece sin modificar la parte geométrica de la herramienta. El vector de corrección definido en el WKS (giro alrededor del eje y) debe aceptarse en la parte de desgaste de forma que la longitud total de herramienta en la imagen 3 remita al punto P_2. Por tanto, la parte de desgaste resultante de la herramienta se indica mediante la distancia entre ambos puntos P_1 y P_2.
El desgaste se valora de forma negativa debido al dato de operador SD42930 $SC_WEAR_SIGN, por lo que la corrección determinada debe introducirse en la memoria de corrección con signo negativo. El contenido de los parámetros de herramienta relevantes al final del programa es:

$TC_DP3[1,1] : 10.000 ; Geometría L1 (no modificado)

$TC_DP4[1,1] : 15.000 ; Geometría L2 (no modificado)

$TC_DP5[1,1] : 10.000 ; Geometría L3 (no modificado)

$TC_DP12[1,1] : 2.010 ; Desgaste L1 (= 10 - 15 * cos(30) + 10 * sin(30))

$TC_DP13[1,1] : -16.160 ; Desgaste L2 (= -15 * sin(30) - 10 * cos(30))

$TC_DP14[1,1] : -5.000 ; Desgaste L3

Mediante el giro de frame puede reconocerse en el componente L3 en dirección Y el efecto del dato de operador SD42930 $SC_WEAR_SIGN sin la complicación adicional.

información adicional

Herramientas de torneado/rectificado: Cálculo de la longitud de herramienta según MD20360 $MC_TOOL_PARAMETER_DEF_MASK

La forma en la que, para herramientas de torneado/rectificado, deben evaluarse el desgaste o la longitud de herramienta en un posible eje de diámetro se determina en el dato de máquina:

MD20360 $MC_TOOL_PARAMETER_DEF_MASK.<Bit> = <Wert>

<Bit>	<Valor>	Descripción
0	0	En las herramientas de torneado/rectificado, el **parámetro de desgaste** del eje de refrentado se calcula en el **valor de radio**:
	1	En las herramientas de torneado/rectificado, el **parámetro de desgaste** del eje de refrentado se calcula como **valor de diámetro**:
1	0	En las herramientas de torneado/rectificado, el **componente longitudinal de herramienta** del eje de refrentado se calcula como **valor de radio**:
	1	En las herramientas de torneado/rectificado, el **componente longitudinal de herramienta** del eje de refrentado se calcula como **valor de diámetro**:

Si están seteados los bits correspondientes, se valora la entrada con el factor 0,5. La corrección mediante SETTCOR se ejecuta de forma que la modificación de longitud de herramienta total efectiva sea igual al valor transferido a <CorVal>. Si, a causa del dato de máquina MD20360 $MC_TOOL_PARAMETER_DEF_MASK, la longitud se valora con el factor 0,5 durante el cálculo de longitud, la corrección de este componente debe efectuarse con el doble del valor transferido.

Ejemplo

MD20360 $MC_TOOL_PARAMETER_DEF_MASK = 2 (la longitud de herramienta en el eje de diámetro debe valorarse con el factor 0,5)

Eje X es el eje de diámetro.

```
Código de programa                          Comentarios
N10 DEF REAL _LEN[11]
N20 DEF REAL _CORVAL[3]
N30 $TC_DP1[1,1]=500                        ; Tipo de herramienta
N40 $TC_DP2[1,1]=2                          ; Posición del filo
N50 $TC_DP3[1,1]=3.                         ; Geometría - Longitud 1
N60 $TC_DP4[1,1]=4.                         ; Geometría - Longitud 2
N70 $TC_DP5[1,1]=5.                         ; Geometría - Longitud 3
N80 _CORVAL[0]=1.
N90 _CORVAL[1]=1.
N100 _CORVAL[2]=1.
N110 T1 D1 G18 G0 X0 Y0 Z0                  ; ==> Posición MKS X1,5 Y5 Z4
N120 R1=SETTCOR(_CORVAL,"G",1,1)
N130 T1 D1 X0 Y0 Z0                         ; ==> Posición MKS X2,5 Y6 Z5
N140 R3=$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 corrección de la longitud de herramienta debe ser de 1 mm en cada eje (N80 a N100). Por esa razón, en las longitudes L2 y L3 se añade respectivamente 1 mm a la longitud original. Por el contrario, a la longitud de herramienta original en L1 se añade el doble del valor de corrección (2 mm), para que la longitud de herramienta total se modifique 1 mm como se requiere. En comparación con las posiciones de aproximación en las secuencias N110 y N130, se reconoce que cada posición del eje se ha modificado 1 mm.

3.13.14 Leer la asignación de las longitudes de herramienta L1, L2, L3 a los ejes de coordenadas (LENTOAX)

La función LENTOAX proporciona información acerca de la asignación de las longitudes L1, L2 y L3 de la herramienta **activa** a abscisa, ordenada y eje de aplicadas. La asignación de abscisa, ordenada y eje de aplicadas a los ejes geométricos es influida por frames y el plano activo (G17-G19).

Solo se tiene en cuenta la parte geométrica de una herramienta ($TC_DP3[<t>,<d>] a $TC_DP5[<t>,<d>]), es decir, en el resultado no influye ninguna asignación de eje diferente de otros componentes (p. ej., desgaste).

Sintaxis

```
<Estado> = LENTOAX(<IndEje>, <Matriz>[, <Coord>])
```

Principio

LENTOAX(...):	Función predefinida para la lectura de la asignación de las longitudes de herramienta L1, L2 y L3 de la herramienta activa a los ejes de coordenadas		
	Único elemento de la secuencia:		sí
<Estado>:	Valor de retorno de la función. Los valores negativos indican estados de error.		
	Tipo de dato:		INT
	Valor:	0	Función OK La información en <IndEj> es suficiente para la descripción (todos los componentes longitudinales de herramienta son paralelos a los ejes geométricos).
		1	La función es OK, pero para una descripción correcta debe evaluarse el contenido de <Matriz> (los componentes longitudinales de herramienta no son paralelos a los ejes geométricos).
		-1	Cadena de caracteres no válida en el parámetro <Coord>.
		-2	Sin herramienta activa.
Parámetros			
1 <IndEj>:	Si los componentes longitudinales de herramienta son paralelos a los ejes geométricos, los índices axiales que están asignados a los componentes longitudinales L1 a L3 se devuelven en el campo <IndEj>: • <IndEj> [0]: Abscisa • <IndEj> [1]: Ordenada • <IndEj> [2]: Eje de aplicadas		
	Tipo de dato:		INT[3]
	Valor:	0	No existe ninguna asignación (no existe ningún eje)
		1 ... 3 o bien -1 ... -3	Número de la longitud que actúa sobre el eje de coordenadas correspondiente. El signo es negativo cuando el componente longitudinal de herramienta señala en dirección de coordenadas negativa.
	Si no todos los componentes longitudinales son paralelos o antiparalelos a los ejes geométricos, se devuelve en <IndEj> en cada caso el índice de eje que incluye la mayor parte de un componente longitudinal de herramienta. En ese caso (si la función no devuelve un error por otro motivo), el valor de retorno es <Estado> = 1. La representación de los componentes longitudinales de herramienta L1 a L3 en los ejes geométricos 1 a 3 se describe totalmente en el contenido del 2.º parámetro <Matriz>.		
2 <Matriz>:	Matriz que representa el vector de las longitudes de herramienta (L1=1, L2=1, L3=1) en el vector de los ejes de coordenadas (abscisa, ordenada, eje de aplicadas), es decir, a las columnas se les asignan los componentes longitudinales de herramienta en la secuencia L1, L2, L3, y a las líneas, los ejes en la secuencia abscisa, ordenada, eje de aplicadas.		
	Tipo de dato:		REAL
	En la matriz todos los elementos son siempre válidos, incluso cuando no existe el eje geométrico perteneciente a un eje de coordenadas, es decir, cuando la entrada correspondiente en <IndEj> es cero.		

3	<Coord>:	Sistema de coordenadas en el que se aplica la asignación (opcional)		
		Tipo de dato:		STRING
		Carácter:	MCS M	Representación de la longitud de herramienta en el sistema de coordenadas de máquina
			BCS B	Representación de la longitud de herramienta en el sistema de coordenadas básico
			WCS W	Representación de la longitud de herramienta en el sistema de coordenadas de pieza (predeterminado)
			KCS K	Representación de la longitud de herramienta en el sistema de coordenadas de herramienta de la transformada cinemática
			TCS T	Representación de la longitud de herramienta en el sistema de coordenadas de herramienta
		La notación de los caracteres en la cadena (mayúscula o minúscula) es indiferente. Si no se indica el parámetro <Coord>, se utiliza el WKS (predeterminado).		

Nota

En el TCS todos los componentes longitudinales de herramienta son paralelos o antiparalelos a los ejes.

Los componentes solo pueden ser antiparalelos si está activa la simetría y está ajustado el siguiente dato de operador:

SD42900 $SC_MIRROR_TOOL_LENGTH (cambio de signo de longitud de herramienta en la simetría)

Ejemplo

Caso estándar fresa con G17.

L1 actúa en Z (eje de aplicadas), L2 actúa en Y (ordenada), L3 actúa en X (abscisa).

Llamada de la función en el formato:
`<Estado>=LENTOAX(<IndEj>,<Matriz>,"WCS")`

El parámetro de resultado <IndEj> incluye entonces los valores:

<IndEj>[0] = 3

<IndEj>[1] = 2

<IndEj>[2] = 1

o abreviadamente: (3, 2, 1)

En este caso la matriz correspondiente (<Matriz>) es:

$$\text{<Matrix>} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

Un cambio de G17 a G18 o G19 no modifica el resultado, ya que la asignación de los componentes longitudinales a los ejes geométricos se modifica de la misma manera que la asignación de abscisa, ordenada y eje de aplicadas.

Ahora con G17 activo se programa un giro de frame alrededor de Z de 60 grados, p. ej.:
`ROT Z60`

La dirección del eje de aplicadas (dirección Z) no se modifica, el componente principal de L2 está ahora en dirección del nuevo eje X, el componente principal de L1 en dirección del eje negativo Y. El valor de retorno (<Estado>) es, por tanto, "1", <IndEj> incluye los valores (2, -3, 1).

En este caso la matriz correspondiente (<Matriz>) es:

$$\text{<Matrix>} = \begin{pmatrix} 0 & \sin 60° & \cos 60° \\ 0 & \cos 60° & -\sin 60° \\ 1 & 0 & 0 \end{pmatrix}$$

3.14 Comportamiento de contorneado

3.14.1 Variación del avance (FNORM, FLIN, FCUB, FPO)

Para poder flexibilizar la introducción de la ley de variación del avance, se ha ampliado la programación de este (siguiendo la norma DIN 66025) mediante leyes de variación lineal y cúbica.

La ley de variación cúbica se puede programar directamente, o mediante un spline interpolatorio. De esta forma se pueden conseguir unas velocidades de avance bastante uniformes cuando se tienen que mecanizar contornos muy angulosos.

Estas leyes de variación de velocidad permiten realizar cambios de aceleración evitando sacudidas bruscas, consiguiendo por lo tanto un acabado superficial más uniforme.

Sintaxis

```
F… FNORM
F… FLIN
F… FCUB
F=FPO(…,…,…)
```

Descripción

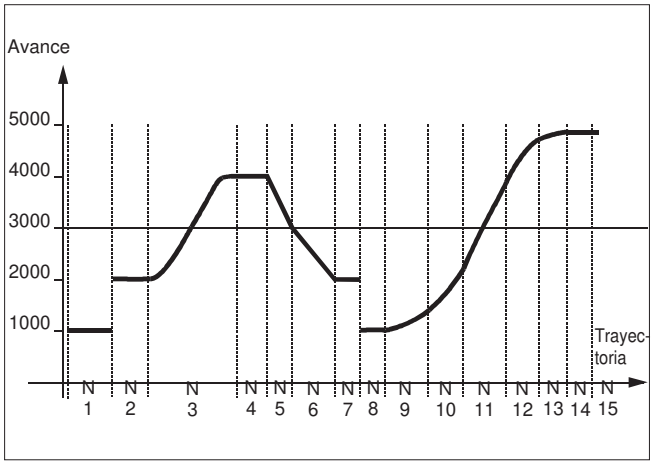
FNORM:	Ajuste básico. El valor para el avance se define conjuntamente con el desplazamiento, generalmente en la misma secuencia, y después de esto es válido de forma modal.
FLIN:	Perfil de velocidad de contorneado **lineal:** El valor del avance varía de forma lineal desde la posición inicial de la secuencia hasta la posición final indicada en la secuencia y después de esto es válido de forma modal. Este comportamiento se puede combinar con G93 y G94.
FCUB:	Perfil de velocidad de contorneado **cúbico:** Los valores F programados secuencia a secuencia se asocian, con referencia al punto final de la secuencia, mediante una interpolación polinómica de tercer orden (spline). El spline se inicia y finaliza tangencialmente al dato de avance anterior y posterior y actúa con G93 y G94. Si se omite la dirección F en una secuencia, entonces se toma el último valor F programado.
F=FPO…:	Perfil de velocidad de contorneado **mediante un polinomio:** La dirección F describe la variación del avance siguiendo una ley polinómica desde el valor actual de avance hasta el final de la secuencia. A continuación, el valor final se toma como modal.

Optimización del avance en tramos de contornos muy angulosos

El avance polinómico `F=FPO` y el avance spline `FCUB` deben ejecutarse siempre con velocidad de corte constante `CFC`. De esta forma se consigue un perfil de velocidades bastante uniforme y prácticamente sin aceleraciones bruscas.

Ejemplo: Diferentes perfiles de avance

En este ejemplo se puede ver la programación de los diferentes perfiles para el avance así como su representación gráfica.

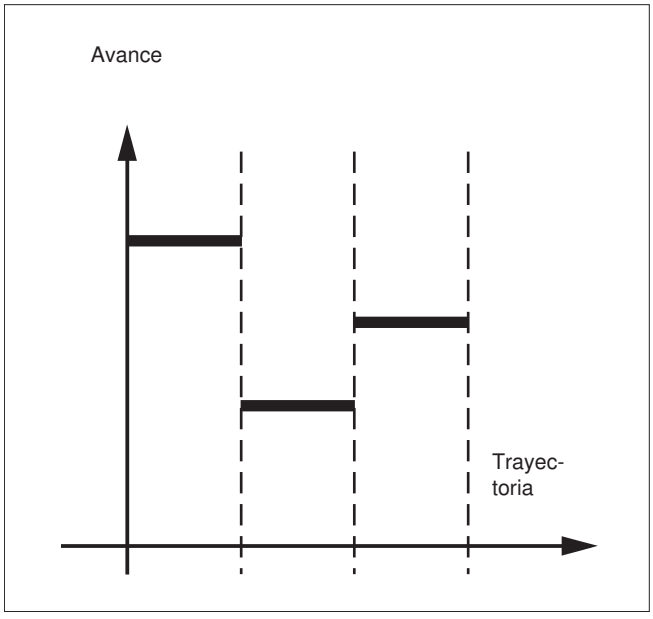


Código de programa	Comentarios
N1 F1000 FNORM G1 X8 G91 G64	; Perfil de avance constante, acotado incremental
N2 F2000 X7	; Modificación brusca de la velocidad de consigna
N3 F=FPO(4000, 6000, -4000)	; Perfil de avance mediante polinomio con avance 4000 al final de la secuencia
N4 X6	; El avance por polinomio 4000 vale como valor modal.
N5 F3000 FLIN X5	; Perfil de avance lineal
N6 F2000 X8	; Perfil de avance lineal
N7 X5	; Perfil de avance lineal, vale como valor modal
N8 F1000 FNORM X5	; Perfil de avance constante con modificación brusca de la aceleración.
N9 F1400 FCUB X8	; Los siguientes valores F programados por secuencias se unirán con splines.
N10 F2200 X6	
N11 F3900 X7	
N12 F4600 X7	
N13 F4900 X5	; Desactivar perfil de spline.
N14 FNORM X5	
N15 X20	

Información adicional

FNORM

La dirección F describe el valor para avance constante según la norma DIN 66025.

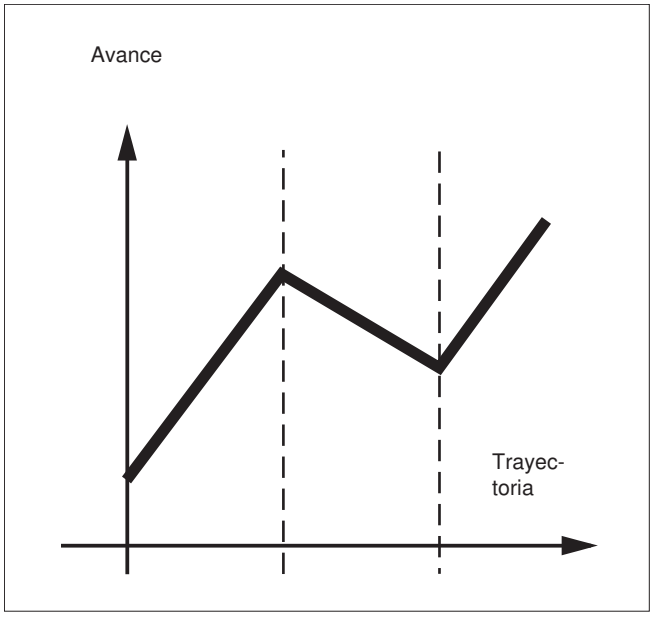


FLIN

El avance varía de forma lineal a lo largo de la trayectoria desde el valor del avance actual hasta el valor F programado.

Ejemplo:

```
N30 F1400 FLIN X50
```



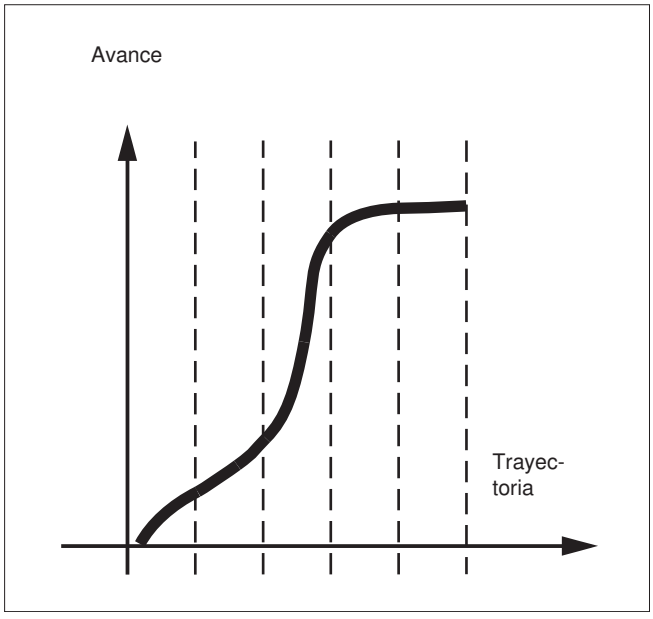
FCUB

El avance varía de forma cúbica a lo largo de la trayectoria programada desde el valor inicial de avance hasta el valor final programado. El control numérico une mediante splines los

valores de avance programados en la secuencias activas con FCUB. Los valores de avance se toman como puntos de interpolación para el cálculo de los splines.

Ejemplo:

```
N50 F1400 FCUB X50
N60 F2000 X47
N70 F3800 X52
```



F=FPO(...,...,...)

La variación del avance se programa directamente mediante un polinomio. La introducción de los coeficientes polinómicos se realiza de forma análoga a la interpolación polinómica.

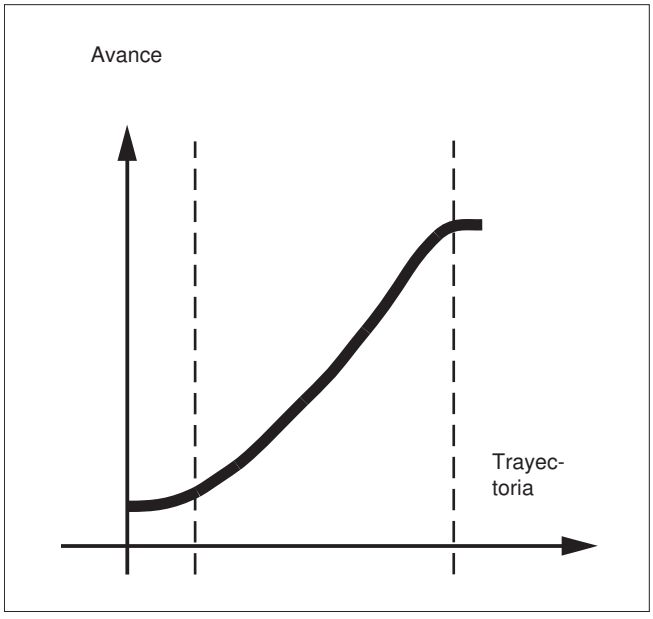
Ejemplo:

```
F=FPO(endfeed, quadf, cubf)
```

endfeed, quadf y cubf son variables con valores previamente definidos.

endfeed:	Avance al final de la secuencia
quadf:	Coeficiente polinómico cuadrado
cubf:	Coeficiente polinómico cúbico

Cuando se tiene la función FCUB activada, el spline que une los avances al principio y al final de la secuencia se concatena de forma tangencial al recorrido definido con FPO.



Condiciones

- Independientemente de la variación del avance programado, son válidas las funciones para la programación de la evolución de la trayectoria.
- La variación de avance programable es válida de forma absoluta independientemente de G90 o G91.
- La variación del avance FLIN y FCUB actúa con G93 y G94, **no** con G95, G96/G961 ni G97/G971.
- Si está activo el compresor COMP... y se juntan varias secuencias formando un segmento spline, se aplica lo siguiente:

FNORM:	Para el segmento spline es válida la palabra F de la última secuencia asociada.
FLIN:	Para el segmento spline es válida la palabra F de la última secuencia asociada. El valor F programado es válido hasta el final del segmento spline y evoluciona de forma lineal.
FCUB:	El spline de avances generado se desvía del valor final programado como máximo en la cantidad indicada en el dato de máquina MD20172 $MC_COMPRESS_VELO_TOL.
F=FPO(…,…,…):	Estas secuencias no se comprimen.

3.14.2 Comportamiento en aceleración

3.14.2.1 Modo de aceleración (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA)

Para la programación del modo de aceleración se dispone de los siguientes comandos del programa de pieza:

- "BRISK, BRISKA"
  Los ejes individuales o de contorneado se desplazan con una aceleración máxima hasta alcanzar la velocidad de avance programada (**aceleración sin limitación de tirones**).
- "SOFT, SOFTA"
  Los ejes individuales o de contorneado se desplazan con una aceleración constante hasta alcanzar la velocidad de avance programada (**aceleración con limitación de tirones**).
- "DRIVE, DRIVEA"
  Los ejes individuales o de contorneado se desplazan con una aceleración máxima hasta un límite de velocidad configurado (ajuste DM). A partir de ese momento se reduce la aceleración (ajuste de DM) hasta alcanzar la velocidad de avance programada.

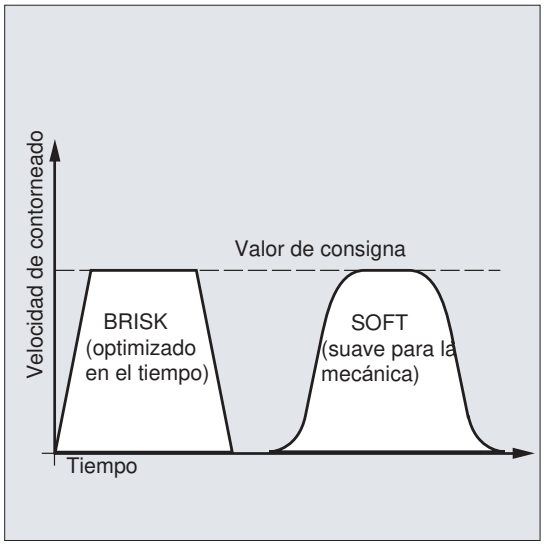


Figura 3-12 Evolución de la velocidad de contorneado en `BRISK` y `SOFT`

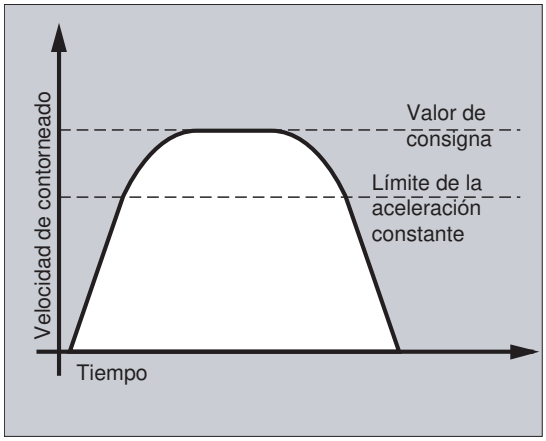


Figura 3-13 Evolución de la velocidad de contorneado en DRIVE

Sintaxis

```
BRISK
BRISKA(<Eje1>,<Eje2>…)
SOFT
SOFTA(<Eje1>,<Eje2>…)
DRIVE
DRIVEA(<Eje1>,<Eje2>…)
```

Descripción

BRISK:	Comando para la activación de la "aceleración sin limitación de tirones" para los ejes de contorneado.
BRISKA:	Comando para la activación de la "aceleración sin limitación de tirones" para movimientos de eje individual (JOG, JOG/INC, eje de vaivén, eje oscilante, etc.).
SOFT:	Comando para la activación de la "aceleración con limitación de tirones" para los ejes de contorneado.
SOFTA:	Comando para la activación de la "aceleración con limitación de tirones" para movimientos de eje individual (JOG, JOG/INC, eje de vaivén, eje oscilante, etc.).
DRIVE:	Comando para la activación de la aceleración reducida por encima de un límite de velocidad configurado (MD35220 $MA_ACCEL_REDUCTION_SPEED_POINT) para los ejes de contorneado.
DRIVEA:	Comando para la activación de la aceleración reducida por encima de un límite de velocidad configurado (MD35220 $MA_ACCEL_REDUCTION_SPEED_POINT) para movimientos de eje individual (JOG, JOG/INC, eje de posicionado, eje de vaivén, etc.).
(<Eje1>,<Eje2>,…):	Ejes individuales a los que debe aplicarse el modo de aceleración llamado.

Condiciones

Cambio del modo de aceleración durante el mecanizado

Cuando se cambia el modo de aceleración en un programa de pieza durante el mecanizado (BRISK ↔ SOFT), en el modo de contorneado se produce también en la transición un cambio de secuencia con parada precisa al final de la secuencia.

Ejemplos

Ejemplo 1: SOFT y BRISKA

Código de programa
N10 G1 X… Y… F900 SOFT
N20 BRISKA(AX5,AX6)

Código de programa
...

Ejemplo 2: DRIVE y DRIVEA

Código de programa
N05 DRIVE
N10 G1 X… Y… F1000
N20 DRIVEA (AX4, AX6)
...

3.14.2.2 Influencia en la aceleración en ejes de seguimiento (VELOLIMA, ACCLIMA, JERKLIMA)

En acoplamientos de ejes (Página 921) (seguimiento tangencial, arrastre de ejes, acoplamiento de valores maestros y reductor electrónico), se desplazan ejes o cabezales esclavos en función de uno o varios ejes o cabezales maestros.

Las limitaciones de la dinámica de los ejes/cabezales esclavos pueden influirse con las funciones VELOLIMA, ACCLIMA y JERKLIMA del programa de pieza o de las acciones síncronas, también con un acoplamiento de ejes ya activo.

Nota

La función JERKLIMA no está disponible para todos los modos de acoplamiento.

Sintaxis

```
VELOLIMA[<Eje>]=<Valor>
ACCLIMA(<Eje>)=<Valor>
JERKLIMA(<Eje>)=<Valor>
```

Descripción

VELOLIMA:	Comando para la corrección de la **velocidad** máxima parametrizada
ACCLIMA:	Comando para la corrección de la **aceleración** máxima parametrizada
JERKLIMA:	Comando para la corrección del **jerk** máximo parametrizado
<Eje>:	Eje esclavo cuyas limitaciones de la dinámica deben corregirse
<Valor>:	Valor porcentual de corrección

Ejemplos

Ejemplo 1: corrección de las limitaciones de la dinámica para un eje esclavo (AX4)

Código de programa	**Comentarios**
...	
VELOLIMA[AX4]=75	; Corrección de la limitación al 75% de la velocidad máxima del eje consignada en el dato de máquina.

Código de programa	Comentarios
ACCLIMA[AX4]=50	; Corrección de la limitación al 50% de la aceleración máxima del eje consignada en el dato de máquina.
JERKLIMA[AX4]=50	; Corrección de la limitación al 50% del tirón máximo del eje consignado en el dato de máquina en el desplazamiento de trayectoria.
...	

Ejemplo 2: reductor electrónico

El eje 4 se acopla al eje X a través de un acoplamiento "Reductor electrónico". La capacidad de aceleración del eje esclavo se limita al 70% de la aceleración máxima. La máxima velocidad admisible se limita al 50% de la velocidad máxima. Tras la activación del acoplamiento, la máxima velocidad admisible se vuelve a ajustar al 100%.

Código de programa	Comentarios
...	
N120 ACCLIMA[AX4]=70	; Aceleración máxima reducida.
N130 VELOLIMA[AX4]=50	; Velocidad máxima reducida.
...	
N150 EGON(AX4,"FINE",X,1,2)	; Activación del acoplamiento de reductor electrónico.
...	
N200 VELOLIMA[AX4]=100	; Velocidad máxima plena.
...	

Ejemplo 3: influencia por acción síncrona estática en el acoplamiento de valores maestros

El eje 4 se acopla a X mediante acoplamiento de valores de control. Las características de aceleración se limitan por cada acción síncrona estática 2 a partir de la posición 100 al 80%.

Código de programa	Comentarios
...	
N120 IDS=2 WHENEVER $AA_IM[AX4] > 100 DO ACCLIMA[AX4]=80	; Acción síncrona
N130 LEADON(AX4, X, 2)	; Acoplamiento de valores maestros activado
...	

3.14.2.3 Activación de los valores de dinámica específicos de tecnología (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH, DYNPREC)

Con los comandos del grupo G 59 "Modo dinámico para interpolación de la trayectoria" se puede activar la dinámica adecuada para diferentes pasos de mecanizado tecnológicos.

Los valores de dinámica y los comandos G son configurables, con lo cual dependen de los ajustes de datos de máquina.

Información adicional: Manual de funciones Funciones básicas

Sintaxis

Activar valores dinámicos:
```
DYNNORM/DYNPOS/DYNROUGH/DYNSEMIFIN/DYNFINISH/DYNPREC
```

Nota

Los valores de dinámica actúan ya en la secuencia en la cual se programa el correspondiente comando G. No sigue ninguna parada del mecanizado.

Escribir o leer un determinado elemento de matriz:
```
R<m>=$MA...[n,X]
$MA...[n,X]=<valor>
```

Descripción

DYNNORM:	Activar **dinámica normal**	
DYNPOS:	Activar dinámica para **modo de posicionado, roscado de taladros**	
DYNROUGH:	Activar dinámica para **desbaste**	
DYNSEMIFIN:	Activar dinámica para **preacabado**	
DYNFINISH:	Activar dinámica para **acabado**	
DYNPREC:	Activar dinámica para **acabado fino**	
R<m>:	Parámetro de cálculo con número <m>	
$MA...[n,X]:	Dato de máquina con elemento de matriz determinante de la dinámica	
<n>:	Índice de matriz	
	Rango de valores:	0 ... 5
	0	Dinámica normal (DYNNORM)
	1	Dinámica para el modo de posicionado (DYNPOS)
	2	Dinámica para desbaste (DYNROUGH)
	3	Dinámica para preacabado (DYNSEMIFIN)
	4	Dinámica para acabado (DYNFINISH)
	5	Dinámica para acabado fino (DYNPREC)
<X>:	Dirección de eje	
<Valor>:	Valor de dinámica	

Ejemplos

Ejemplo 1: activar valores de dinámica

Código de programa	**Comentarios**
DYNNORM G1 X10	; Posición básica
DYNPOS G1 X10 Y20 Z30 F…	; Modo de posicionado, roscado de taladros
DYNROUGH G1 X10 Y20 Z30 F10000	; Desbaste
DYNSEMIFIN G1 X10 Y20 Z30 F2000	; Preacabado

Código de programa	Comentarios
DYNFINISH G1 X10 Y20 Z30 F1000	; Acabado
DYNPREC G1 X10 Y20 Z30 F600	; Acabado fino

Ejemplo 2: escribir o leer un determinado elemento de matriz

Aceleración máxima para desbaste, eje X.

Código de programa	Comentarios
R1=$MA_MAX_AX_ACCEL[2,X]	; Lectura
$MA_MAX_AX_ACCEL[2,X]=5	; Escritura

3.14.3 Desplazamiento con mando anticipativo (FFWON, FFWOF)

Mediante el desplazamiento con mando anticipativo se reduce hasta un valor cercano a cero el error de seguimiento generado por el desplazamiento de contorneado. Desplazamientos con mando anticipativo permiten mayor precisión y por lo tanto mejores acabados.

Sintaxis

```
FFWON
FFWOF
```

Descripción

FFWON:	Comando para la **activación** del mando anticipativo
FFWOF:	Comando para la **desactivación** del mando anticipativo

Nota

Mediante datos de máquina se define el tipo de mando anticipativo y qué ejes de contorneado deberán moverse de este modo.

Estándar: Mando anticipativo dependiente de la velocidad

Opción: Mando anticipativo dependiente de la aceleración

Ejemplo

Código de programa
N10 FFWON
N20 G1 X… Y… F900 SOFT

3.14.4 Precisión del contorno programable (CPRECON, CPRECOF)

La función "Precisión de contorno programable" reduce el error de contorneado en contornos con curvatura mediante la adaptación automática de la velocidad.

La precisión de contorno que debe respetarse se especifica, en función de la configuración de la máquina (MD20470 $MC_MC_CPREC_WITH_FFW; ver datos del fabricante de la máquina), mediante el dato de operador $SC_CONTPREC o mediante la tolerancia de contorno programada CTOL. Cuanto menor sea el valor y menor sea el factor K_V de los ejes geométricos, más se reducirá el avance de contorneado en contornos con curvatura.

La función "Precisión de contorno programable" se activa y desactiva mediante las instrucciones `CPRECON` y `CPRECOF` en el programa CN.

Sintaxis

```
CPRECON
...
CPRECOF
```

Descripción

`CPRECON`:	Llamada a comando G: **activar** "precisión de contorno programable"	
	Efecto:	modal
`CPRECOF`:	Llamada a comando G: **desactivar** "precisión de contorno programable"	
	Efecto:	modal

CPRECON y CPRECOF conforman el grupo de funciones G 39 (precisión de contorno programable).

Nota

Mediante el dato de operador $SC_MINFEED (avance mínimo con CPRECON), el usuario puede especificar una velocidad de avance de contorneado mínima.

El avance no se limita debajo de este valor salvo que se haya programado un valor F más bajo o que las limitaciones dinámicas de los ejes obliguen a fijar una velocidad de contorneado más baja.

Nota

La función "Precisión de contorno programable" considera solo los ejes geométricos de la trayectoria. No influye en las velocidades de los ejes de posicionado.

Ejemplo

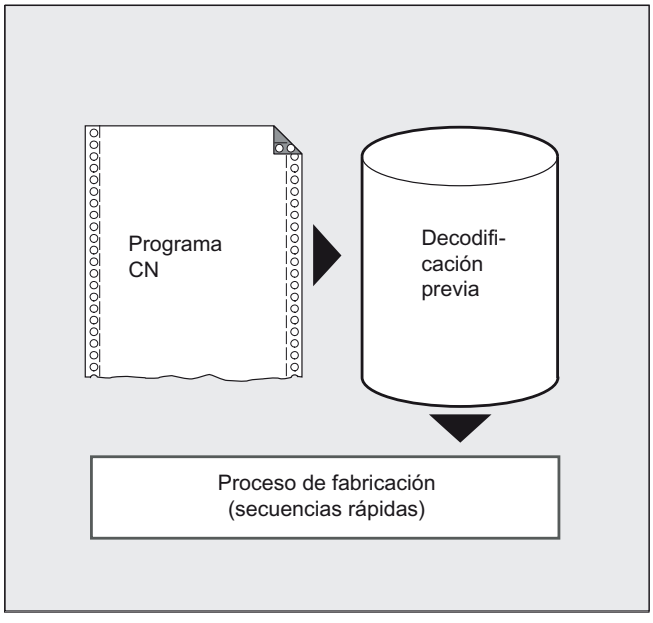
Código de programa	Comentarios
N10 G0 X0 Y0	
N20 CPRECON	; Activación de la "Precisión de contorno programable".
N30 G1 G64 X100 F10000	; Mecanizado con 10 m/min en modo de contorneado.
N40 G3 Y20 J10	; Limitación automática del avance en la secuencia circular.
N50 G1 X0	; Nuevo avance sin limitación (10 m/min).
...	
N100 CPRECOF	; Desactivación de la "Precisión de contorno programable".
N110 G0 ...	

Consulte también

Programar tolerancia de contorno/orientación (CTOL, OTOL, ATOL) (Página 893)

3.14.5 Ejecución del programa con memoria de pretratamiento (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE)

Dependiendo de su configuración, el control numérico dispone de una memoria (búfer) de pretratamiento (con decodificación previa) de una cierta capacidad. En ella se memorizan - antes de la ejecución del programa - secuencias ya preparadas. De allí salen como secuencias rápidas para el proceso de fabricación. Esto permite recorrer trayectos cortos a altas velocidades. Siempre que el tiempo residual del control numérico así lo permita, se llena la memoria de pretratamiento.



Identificación del segmento de ejecución

El segmento de ejecución que debe guardarse en la memoria de pretratamiento se identifica al principio con "STOPFIFO" y al final con "STARTFIFO" en el programa de pieza. La ejecución de las secuencias preparadas y guardadas empieza solo después del comando "STARTFIFO" o cuando la memoria de pretratamiento está llena.

Control automático de la memoria de pretratamiento

El control automático de la memoria de pretratamiento se llama con el comando "FIFOCTRL". Inicialmente, "FIFOCTRL" actúa igual que "STOPFIFO". En cada programación se espera hasta que la memoria de pretratamiento está llena, y entonces empieza la ejecución. Pero en este caso, el comportamiento es diferente al vaciar la memoria de pretratamiento: con "FIFOCTRL", a partir de un nivel de llenado de 2/3 la velocidad de contorneado se reduce cada vez más para impedir un vaciado completo y un frenado hasta la parada.

Parada de decodificación previa

El procesamiento y el almacenamiento intermedio de secuencias se detienen si el comando "STOPRE" está programado en la secuencia. La secuencia subsiguiente sólo se ejecuta cuando se hayan ejecutado completamente todas las secuencias previamente preparadas y memorizadas. La secuencia anterior se detiene en la parada precisa (como G9).

ATENCIÓN
Interrupción del programa
Cuando está activa la corrección de herramienta o en caso de interpolaciones spline no se debería programar "STOPRE", ya que en este caso se interrumpirían series de secuencias conexas.

Sintaxis

Tabla 3-3 Identificar el segmento de ejecución:

```
STOPFIFO
...
STARTFIFO
```

Tabla 3-4 Control automático de la memoria de pretratamiento:

```
...
FIFOCTRL
...
```

Tabla 3-5 Parada de decodificación previa:

```
...
STOPRE
...
```

Nota

Los comandos "STOPFIFO", "STARTFIFO", "FIFOCTRL" y "STOPRE" deben programarse en una secuencia propia.

Descripción

STOPFIFO:	"STOPFIFO" identifica el principio de un segmento de ejecución que debe guardarse en la memoria de pretratamiento. Con "STOPFIFO" se detiene la ejecución y se llena la memoria de pretratamiento hasta que: • se detecta "STARTFIFO" o "STOPRE" o bien • la memoria de pretratamiento está llena o bien • se alcanza el final del programa.
STARTFIFO:	Con "STARTFIFO" empieza la ejecución rápida del segmento de ejecución, y paralelamente se llena la memoria de pretratamiento
FIFOCTRL:	Activación del control automático de la memoria de pretratamiento
STOPRE:	Parar decodificación previa

Nota

El llenado de la memoria de pretratamiento no se ejecuta o se interrumpe si el segmento de ejecución contiene comandos que obligan a un servicio sin respaldo de búfer (búsquedas de punto de referencia, funciones de medida...).

Nota

Cuando se accede a los datos de estado de la máquina ($SA...), el control genera una parada interna de decodificación previa.

Ejemplo: Parar decodificación previa

```
Código de programa                          Comentarios
...
N30 MEAW=1 G1 F1000 X100 Y100 Z50           ; Secuencia de medición con palpador
                                            por la primera entrada e interpolación
                                            lineal.
N40 STOPRE                                  ; Parada de decodificación previa.
...
```

3.14.6 Definir las áreas Stop-Delay (DELAYFSTON, DELAYFSTOF)

Los procedimientos predefinidos DELAYFSTON y DELAYFSTOF sirven para definir un área Stop-Delay con posibilidad de interrupción condicionada en el programa de pieza.

Nota

DELAYFSTON y DELAYFSTOF **no** son admisibles en acciones síncronas.

Sintaxis

```
DELAYFSTON
...
DELAYFSTOF
```

Descripción

DELAYFSTON:	Definir el inicio de un área Stop-Delay	
	Único elemento de la secuencia:	Sí
DELAYFSTOF:	Definición del fin de un área Stop-Delay	
	Único elemento de la secuencia:	Sí

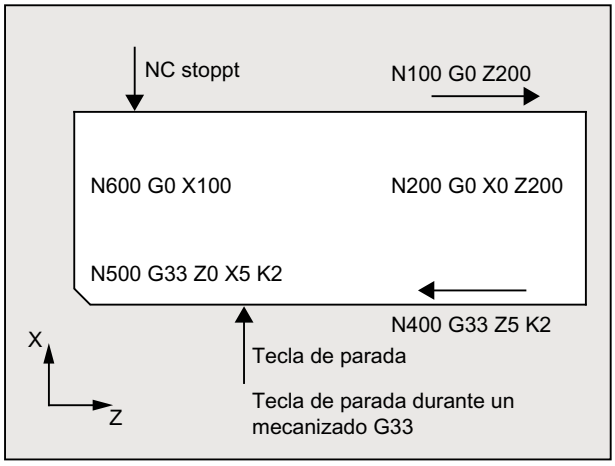
Ejemplo de programación

En un bucle se repite el siguiente bloque de programa:

Código de programa

```
...
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
...
```

En la siguiente figura se puede ver que el usuario pulsa "Parada" en el área Stop-Delay; el CN inicia el frenado fuera del área Stop-Delay, es decir, en la secuencia N100. De este modo, el CN se detiene en el área delantera de N100.



Información adicional

Fin de subprograma

Al final del subprograma en el cual se llamó a DELAYFSTON, se activa implícitamente DELAYFSTOF.

Imbricación

Si el subprograma 1 llama al subprograma 2 en un área Stop-Delay, el subprograma 2 completo es un área Stop-Delay. En particular, DELAYFSTOF queda sin efecto en el subprograma 2.

Ejemplo:

```
Código de programa          Comentarios
N10010 DELAYFSTON           ; Secuencias con N10xxx nivel de programa 1.
N10020 R1 = R1 + 1
N10030 G4 F1                ; Empieza el área Stop-Delay.
...
N10040 Subprograma 2
...
...                         ; Interpretación del subprograma 2.
N20010 DELAYFSTON           ; Sin efecto, inicio repetido, 2.° nivel.
...
N20020 DELAYFSTOF           ; Sin efecto, fin en otro nivel.
N20030 RET
N10050 DELAYFSTOF           ; Fin del área Stop-Delay en el mismo nivel.
...
N10060 R2 = R2 + 2
N10070 G4 F1                ; Finaliza el área Stop-Delay. A partir de ahora, las
                            paradas tienen efecto inmediato.
```

Variables del sistema

Con las siguientes variables de sistema se realiza una consulta para averiguar si la ejecución del programa de pieza se encuentra en un área Stop-Delay en estos momentos:

- En el programa de pieza con $P_DELAYFST
- En acciones síncronas con $AC_DELAYFST

Valor	Descripción
0	Área Stop-Delay no activa
1	Área Stop-Delay activa

3.14.7 Evitar lugar del programa para SERUPRO (IPTRLOCK, IPTRUNLOCK)

Para determinadas situaciones mecánicas complicadas en la máquina es necesario evitar la búsqueda de secuencia con SERUPRO.

Con un puntero de interrupción programable existe una posibilidad de intervención, posicionándose en la "búsqueda en el punto de interrupción" delante del punto no apto para la búsqueda.

También se pueden definir zonas no aptas para la búsqueda en áreas del programa de pieza donde el CN no puede entrar todavía. Con la interrupción de programa, el CN registra la última secuencia procesada en la que puede realizarse una búsqueda mediante la interfaz de usuario HMI.

Sintaxis

```
IPTRLOCK
IPTRUNLOCK
```

Los comandos están solos en una línea del programa de pieza y permiten un puntero de interrupción programable

Descripción

`IPTRLOCK`:	Inicio de la sección de programa no apta para la búsqueda
`IPTRUNLOCK`:	Fin de la sección de programa no apta para la búsqueda

Ambos comandos se admiten únicamente en programas de pieza pero **no** en acciones síncronas.

Ejemplo

Imbricación de secciones de programa no aptas para la búsqueda en dos niveles de programa con "IPTRUNLOCK" implícito. El "IPTRUNLOCK" implícito en el subprograma 1 termina el área no apta para la búsqueda.

Código de programa	**Comentarios**
`N10010 IPTRLOCK()`	

Código de programa	Comentarios
N10020 R1 = R1 + 1	
N10030 G4 F1	; Secuencia de parada, empieza la sección de programa no apta para la búsqueda.
...	
N10040 Subprograma 2	
...	; Interpretación del subprograma 2.
N20010 IPTRLOCK ()	; Sin efecto, inicio repetido.
...	
N20020 IPTRUNLOCK ()	; Sin efecto, fin en otro nivel.
N20030 RET	
...	
N10060 R2 = R2 + 2	
N10070 RET	; Fin de la sección de programa no apta para la búsqueda.
N100 G4 F2	; Se prosigue con el programa principal.

Entonces, una interrupción en 100 vuelve a suministrar el puntero de interrupción.

Información adicional

Registrar y buscar áreas no aptas para la búsqueda

Las secciones de programa no aptas para la búsqueda se identifican con las instrucciones de programación "IPTRLOCK" e "IPTRUNLOCK".

El comando "IPTRLOCK" congela el puntero de interrupción en una secuencia individual ejecutable en el proceso principal (SB1). En lo sucesivo, esta secuencia se denomina secuencia de parada. Si se produce una cancelación del programa después de "IPTRLOCK", se puede buscar esta secuencia, denominada de parada, en la interfaz hombre-máquina (HMI).

Reposicionar en la secuencia actual

El puntero de interrupción se posiciona con "IPTRUNLOCK", para la sección de programa posterior, en la secuencia actual respecto al punto de interrupción.

Una vez que se haya encontrado una meta de búsqueda, se puede repetir una nueva meta de búsqueda con la misma secuencia de parada.

Un puntero de interrupción editado por el usuario se tiene que volver a quitar vía HMI.

Reglas en imbricaciones

Los siguientes puntos regulan la coordinación de las instrucciones de programación "IPTRLOCK" e "IPTRUNLOCK" con imbricaciones y el fin del subprograma:

1. Al final del subprograma en el que se llamó a "IPTRLOCK", se activa implícitamente "IPTRUNLOCK".
2. "IPTRLOCK" en una zona no apta para la búsqueda queda sin efecto.
3. Si el subprograma 1 llama el subprograma 2 en una zona no apta para la búsqueda, el subprograma 2 completo queda no apto para la búsqueda. En particular, "IPTRUNLOCK" queda sin efecto en el subprograma 2.

Información adicional: Manual de funciones Funciones básicas

Variable de sistema

Un área no apta para la búsqueda se puede detectar con "$P_IPTRLOCK" en el programa de pieza.

Puntero de interrupción automático

La función Puntero de interrupción automático define automáticamente un tipo de acoplamiento previamente determinado como no apto para la búsqueda. Mediante un dato de máquina, se activa para

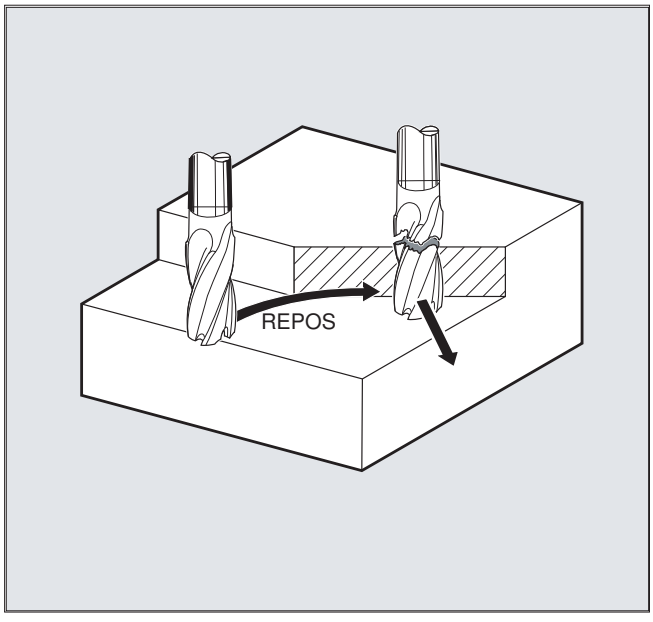
- Reductor electrónico con "EGON"
- Acoplamiento entre ejes de valores maestros con "LEADON"

el puntero de interrupción automático. Si se solapan los punteros de interrupción programado y automático (activable a través de un dato de máquina), se forma el área no apta para la búsqueda más amplia posible.

3.14.8 Reposicionamiento a contorno (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)

Si el programa en curso se interrumpe durante el mecanizado y se retira la herramienta (p. ej., debido a una rotura de herramienta o para medir una pieza), a continuación es posible repetir el posicionado en un punto seleccionable del contorno de forma controlada por el programa.

El comando `REPOS` tiene en un ASUP el mismo efecto que un salto atrás a subprograma (p. ej. M17). Las secuencias subsiguientes ya no se ejecutan. Sobre la interrupción de la secuencia de programa, ver también "Rutina de interrupción (ASUP) (Página 555)".



Sintaxis

```
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=…
```

Descripción

Selección del trayecto de posicionado

REPOSA:	Reposicionamiento en el contorno con los ejes geométricos en una línea recta. Los demás ejes de canal también se reposicionan.
REPOSL:	Reposicionamiento en el contorno con los ejes geométricos en una línea recta. Los demás ejes de canal deben programarse explícitamente.
REPOSQ DISR=… :	Reposicionamiento en el contorno con los ejes geométricos en un cuadrante con radio DISR. Los demás ejes de canal deben programarse explícitamente.
REPOSQA DISR=…:	Reposicionamiento en el contorno con los ejes geométricos en un cuadrante con radio DISR. Los demás ejes de canal también se reposicionan.
REPOSH DISR=… :	Reposicionamiento en el contorno con los ejes geométricos en una semicircunferencia con diámetro DISR. Los demás ejes de canal deben programarse explícitamente.
REPOSHA DISR=…:	Reposicionamiento en el contorno con los ejes geométricos en un semicírculo con radio DISR. Los demás ejes de canal también se reposicionan.

Selección del punto de reposicionado

RMIBL:	Posicionado en el punto de interrupción
RMIBL DISPR=…:	Punto de entrada a una distancia DISR en mm/pulg antes del punto de interrupción
RMBBL:	Posicionado en el inicio de secuencia
RMEBL:	Posicionado en el fin de secuencia
RMEBL DISPR=… :	Posicionado en el fin de secuencia a una distancia DISPR del punto final

RMNBL:	Posicionado en el punto de trayectoria más cercano
A0 B0 C0 :	Ejes en los cuales hay que efectuar el posicionado

Nota

Compatibilidad

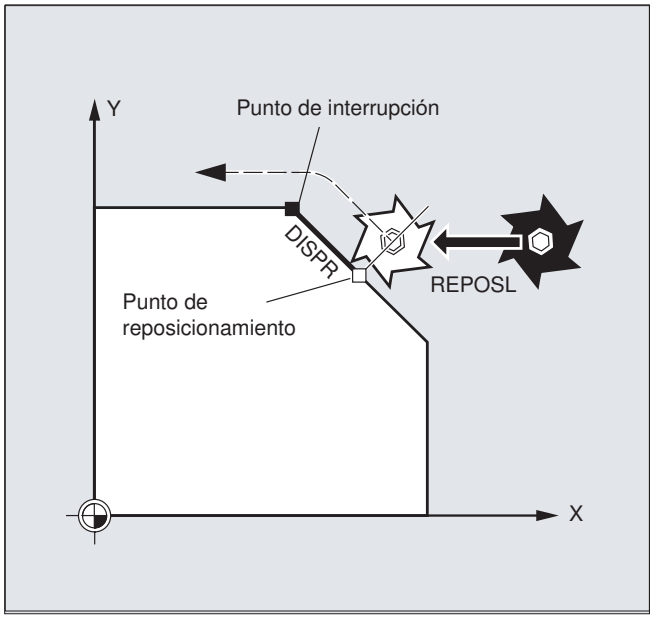
Para mantener la compatibilidad con versiones anteriores del software, es posible seguir programando el modo de aproximación REPOS mediante los comandos modales RMI, RMB, RME y RMN. En caso de utilizarse dentro de un ASUP, este debe tener el atributo SAVE en la instrucción PROC. De lo contrario, el modo de aproximación modal REPOS utilizado en el ASUP actúa también en los posteriores procesos REPOS si no coincide con el ajuste predeterminado RMI.

Reposicionamiento en el contorno en una recta, REPOSA, REPOSL

La herramienta se aproxima al punto de reposi cionamiento siguiendo directamente una línea recta.

Ejemplo

```
REPOSL RMIBL DISPR=6 F400
```

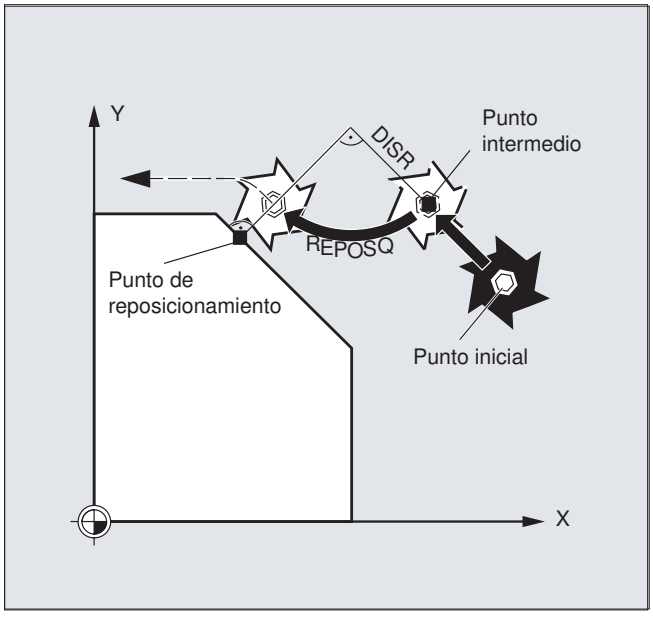


Reposicionamiento en el contorno en cuadrante, REPOSQ, REPOSQA

La herramienta se aproxima al punto de reposicionamiento siguiendo un cuadrante de radioDISR=. . . . El control calcula automáticamente el punto intermedio requerido entre los puntos de arranque y reposicionamiento.

Ejemplo

```
REPOSQ RMIBL DISR=10 F400
```

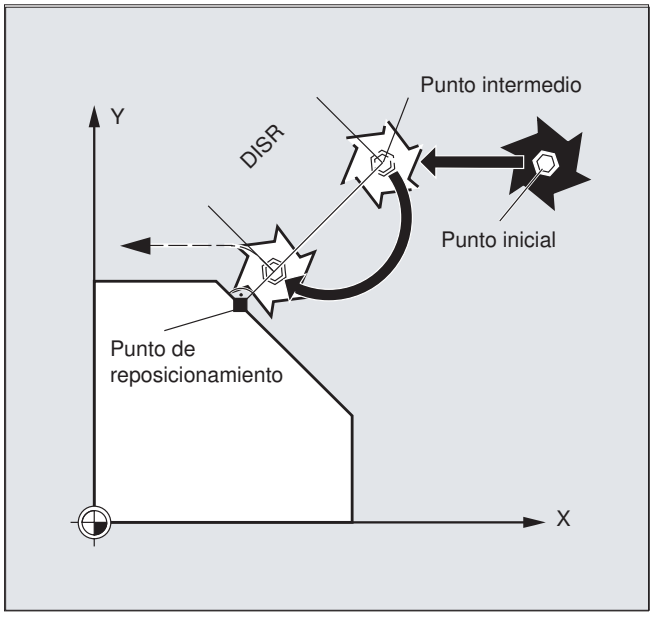


Reposicionamiento en el contorno en semicírculo, REPOSH, REPOSHA

La herramienta se aproxima al punto de reposicionamiento siguiendo un semicircunferencia `DISR=...`. El control calcula automáticamente el punto intermedio requerido entre los puntos de arranque y reposicionamiento.

Ejemplo

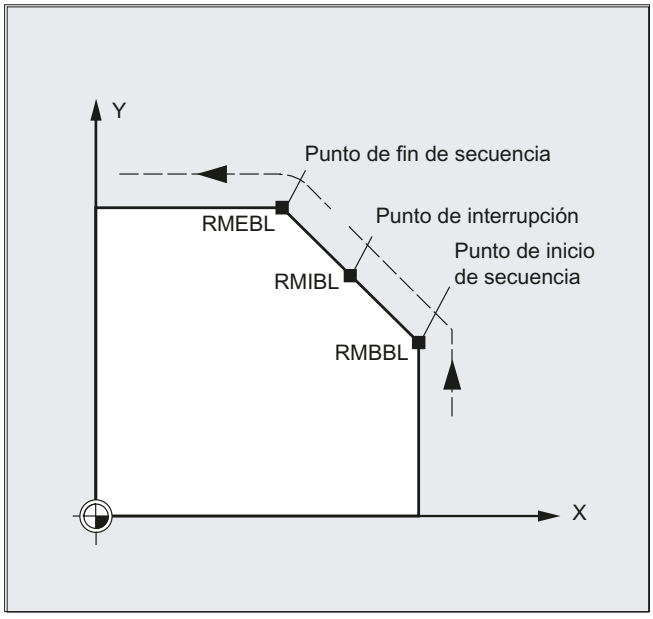
```
REPOSH RMIBL DISR=20 F400
```



Definir punto de reposicionamiento (no para aproximación SERUPRO con RMNBL)

Con respecto a la secuencia de CN en la que se interrumpió la ejecución del programa, se puede elegir entre tres puntos de reposicionamiento:

- RMIBL, punto de interrupción
- RMBBL, punto de inicio de secuencia o bien último punto final
- RMEBL, punto final de secuencia



Con RMIBL DISPR=... o con RMEBL DISPR=... se define un punto de reposicionamiento que se encuentra delante del punto de interrupción o del punto final de la secuencia.

Con DISPR=... se define el trayecto de contorno en mm/pulg. que separa el punto de reposicionamiento del punto de interrupción o del punto final. Como máximo, este punto puede estar localizado - también para valores mayores - en el punto de inicio de secuencia.

Si no se define ningún DISPR=..., se aplica DISPR=0 y, por tanto, el punto de la interrupción (con RMIBL) o el punto final de la secuencia (con RMEBL).

Signo DISPR

Se evalúa el signo de DISPR. En caso de signo positivo, el comportamiento es como hasta ahora.

Si el signo es negativo, se reposiciona detrás del punto de interrupción o, con RMBBL, detrás del punto inicial.

La distancia entre punto de la interrupción-punto de reposicionamiento resulta del valor absoluto de DISPR. Como máximo, este punto puede estar localizado - también para valores de magnitud mayor - en el punto final de secuencia.

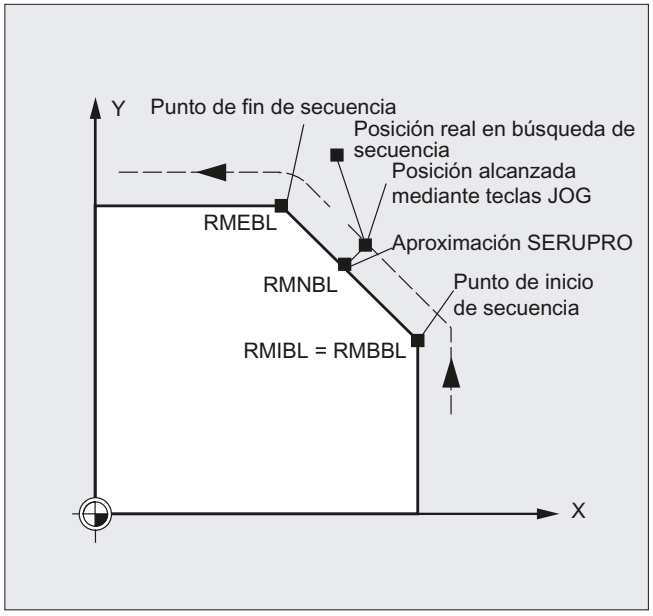
Ejemplo de aplicación:

Con un sensor se detecta la aproximación a una garra de sujeción. Se dispara un ASUP con el que se esquiva la garra de sujeción.

A continuación se reposiciona con el DISPR negativo en un punto detrás de la garra de sujeción y se continúa el programa.

Aproximación SERUPRO con RMNBL

Si se fuerza una interrupción en cualquier punto del mecanizado, se posiciona con aproximación SERUPRO y RMNBL por la vía más corta en el punto de interrupción para ejecutar después solo el recorrido restante. Para este fin, el usuario inicia un proceso SERUPRO con respecto a la secuencia de interrupción y posiciona con las teclas JOG delante del punto defectuoso de la secuencia de destino.



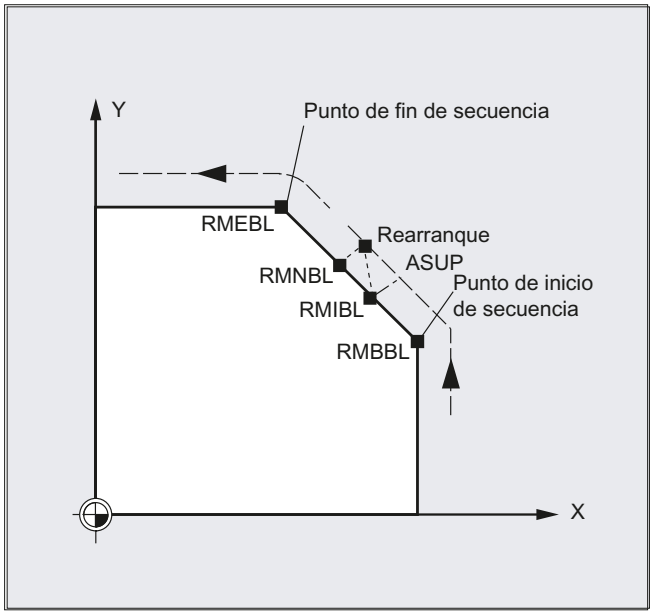
Nota

SERUPRO

Para SERUPRO, RMIBL y RMBBL son idénticos. RMNBL no se limita a SERUPRO, sino que tiene validez general.

Posicionado desde el punto de trayectoria más cercano RMNBL

En el momento de interpretación de REPOSA, la secuencia de reposicionamiento después de una interrupción no se vuelve a iniciar por completo con RMNBL, sino que se ejecuta solo el recorrido restante. Se posiciona en el punto de trayectoria mas cercano de la secuencia interrumpida.



Estado para el modo REPOS válido

El modo REPOS válido de la secuencia interrumpida se puede leer a través de acciones síncronas con la variable $AC_REPOS_PATH_MODE:

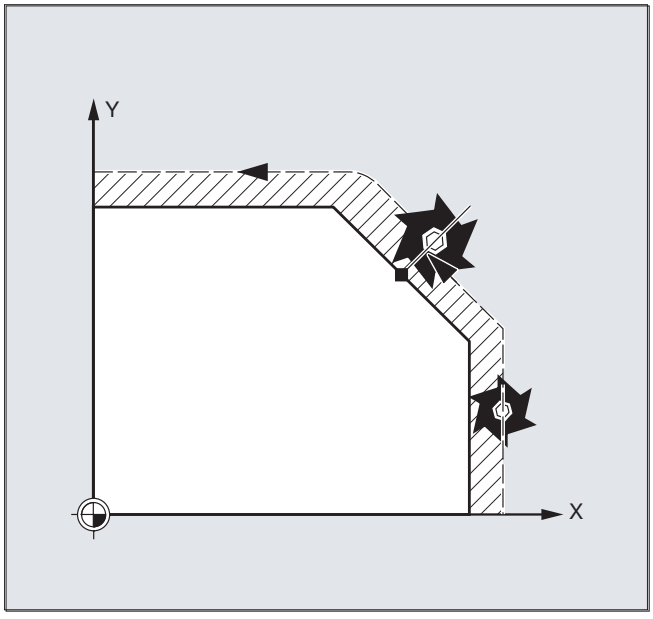
0 Posicionado no definido
1 RMBBL: Posicionado al principio
2 RMIBL: Posicionado en el punto de interrupción
3 RMEBL: Posicionado al final de la secuencia
4 RMNBL: Posicionado en el punto de trayectoria más cercano de la secuencia interrumpida

Aproximación con nueva herramienta

Si la ejecución del programa fue interrumpida por rotura de herramienta:

La programación de un nuevo número D permite continuar el programa a partir del punto de reposicionamiento, utilizando los valores de corrección de herramienta modificados.

Si se modifican los valores de corrección de herramienta es probable que no se pueda reposicionar en el punto de interrupción. En este caso se reposiciona en el punto más cercano a dicho punto en el nuevo contorno (eventualmente modificado con DISPR).



Reposicionamiento en el contorno

El movimiento con el cual la herramienta se aproxima al contorno se puede programar. Las direcciones de los ejes a desplazar se definirán con el valor cero.

Con los comandos REPOSA, REPOSQA y REPOSHA se reposicionan automáticamente todos los ejes. sin necesidad de indicarlos.

Si se programan REPOSL, REPOSQ y REPOSH, todos los ejes geométricos arrancan automáticamente, aunque no se hayan indicado en el comando. Todos los demás ejes se deben definir en el comando.

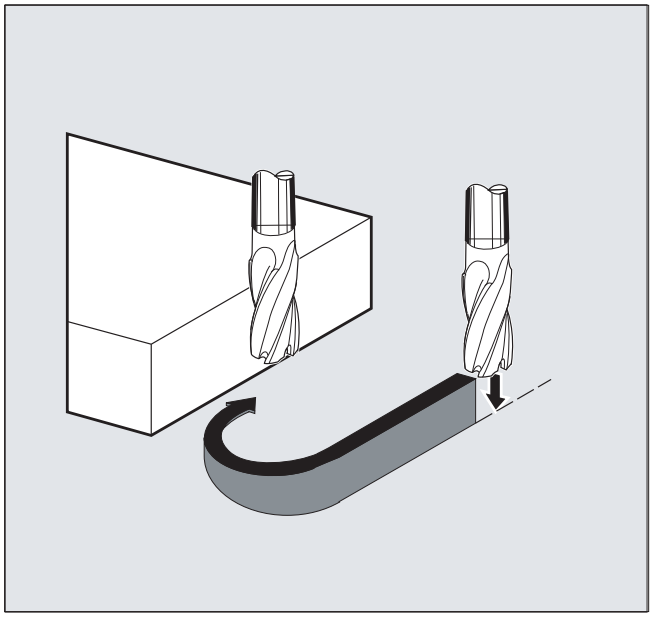
Para los movimientos circulares REPOSH y REPOSQ se aplica:

El arco se recorre en el plano de mecanizado indicado G17 a G19.

Si en la secuencia de posicionado se indica el tercer eje geométrico (sentido de penetración), entonces el punto de reposicionamiento se posiciona en una espiral, en caso de que la posición de herramienta no coincida con la posición programada en el sentido de penetración.

En los siguientes casos se conmuta automáticamente a posicionamiento lineal REPOSL:

- No se indicó ningún valor para DISR.
- No está definido el sentido de aproximación (interrupción de programa en una secuencia sin información de desplazamiento).
- Sentido de aproximación perpendicular al plano actual de trabajo.



3.14.9 Influencia en el guiado de movimiento

3.14.9.1 Adaptación de tirón por eje máximo (JERKLIM)

El comando JERKLIM permite reducir o aumentar el tirón (jerk o sobreaceleración) máximo posible de un eje, ajustado por dato de máquina, en el programa de pieza con desplazamiento de trayectoria en secciones críticas del programa.

Requisitos

El modo de aceleración SOFT tiene que estar activo.

Efecto

JERKLIM surte efecto:

- en los modos de operación AUTOMÁTICO
- solo en ejes de contorneado

Sintaxis

```
JERKLIM[<Ax>]=<Value>
```

Descripción

JERKLIM:	Adaptación del valor límite de tirón
<Ax>:	Eje de máquina cuyo valor límite de tirón debe ajustarse
<Valor>:	Valor porcentual de corrección Valor de corrección referido al tirón por eje máximo configurado con desplazamiento de trayectoria (MD32431 $MA_MAX_AX_JERK)
	Rango de valores: 1 ... 200
	El valor 100 no tiene ninguna influencia en el tirón.

Nota

Comportamiento con fin de programa de pieza y reset de canal

El comportamiento de JERKLIM con fin del programa de pieza y reset de canal depende del ajuste del bit 0 en el dato de máquina MD32320 $MA_DYN_LIMIT_RESET_MASK.

Ejemplo

Código de programa	**Comentarios**
N1000 G0 X0 Y0 F10000 SOFT G64	
N1100 G1 X20 RNDM=5 ACC[X]=20 ACC[Y]=30	
N1200 G1 Y20 JERKLIM[Y]=200	; El desplazamiento del carro en la dirección Y puede acelerarse/retrasarse con máx. el 200% del tirón permitido para el eje.
N1300 G1 X0 JERKLIM[X]=2	; El desplazamiento del carro en la dirección X debe acelerarse/retrasarse con máx. el 2% del tirón permitido para el eje.
N1400 G1 Y0	
M30	

3.14.9.2 Adaptación de la velocidad máxima de eje o cabezal (VELOLIM)

El comando VELOLIM permite reducir, en el programa de pieza, la velocidad máxima de un eje ajustada a través del dato de máquina o la velocidad de giro máxima, dependiente del escalón de reductor, de un cabezal.

Efecto

VELOLIM surte efecto:

- En los modos de operación AUTOMÁTICO
- En ejes de contorneado y de posicionado
- En cabezales en modo Cabezal/Eje

Sintaxis

```
VELOLIM[<Ax>]=<Value>
```

Descripción

VELOLIM:	Adaptación del valor límite de velocidad lineal o de giro
<Ax>:	Eje o cabezal cuyo valor límite de velocidad lineal o de giro debe ajustarse. Mediante MD30455 $MA_MISC_FUNCTION_MASK, bit 6, puede ajustarse si la función VELOLIM debe actuar independientemente de si se está utilizando en ese momento como cabezal o como eje (bit 6 = 1) o si debe poder programarse por separado para cada modo de operación (bit 6 = 0). Si se ha configurado que deben actuar por separado, la selección se realiza mediante el identificador durante la programación: • Identificador de cabezal para modos de operación de cabezal: S<n> • Identificador de eje para el modo Eje, p. ej.: C
<Value>:	Valor porcentual de corrección En caso de ejes o cabezales en modo Eje con el ajuste MD30455, bit 6 = 0, el valor de corrección se refiere a la máxima velocidad de eje configurada (MD32000 $MA_MAX_AX_VELO). En caso de cabezales en modo Eje/Cabezal y el ajuste MD30455, bit 6 = 1, el valor de corrección se refiere a la máxima velocidad de giro del escalón de reductor activo (MD35130 $MA_GEAR_STEP_MAX_VELO_LIMIT[<n>]).
	Rango de valores: 1 ... 100
	El valor 100 no tiene ninguna influencia en la velocidad lineal o de giro.

Nota

Comportamiento con fin de programa de pieza y reset de canal

El comportamiento de VELOLIM con fin del programa de pieza y reset de canal depende del ajuste del bit 0 en el dato de máquina MD32320 $MA_DYN_LIMIT_RESET_MASK.

Nota

Limitación de la velocidad de giro en modo Cabezal

La limitación de la velocidad de giro mediante VELOLIM (inferior al 100%) en el modo Cabezal puede detectarse con las siguientes variables de sistema:

- $AC_SMAXVELO (velocidad de giro máxima posible del cabezal)
- $AC_SMAXVELO_INFO (identificación de la causa que limita la velocidad de giro)

Ejemplos

Ejemplo 1: limitación de la velocidad del eje de máquina

Código de programa	Comentarios
...	
N70 VELOLIM[X]=80	; El desplazamiento del carro en la dirección X debe realizarse con solo un 80% de la velocidad permitida para el eje como máximo.
...	

Ejemplo 2: Limitación de la velocidad de giro del cabezal 1 (AX5)

Configuración:

- MD35130 $MA_GEAR_STEP_MAX_VELO_LIMIT[1, AX5] = 1000
  (velocidad de giro máxima del escalón de reductor 1 = 1000 r/min)
- MD30455 $MA_MISC_FUNCTION_MASK[AX5], bit 6 = 1
  (la programación de VELOLIM tiene efecto conjuntamente para modo Cabezal y modo Eje, independientemente del identificador programado)

Programación:

Código de programa	Comentarios
N05 VELOLIM[S1]=90	; Limitación de la velocidad de giro máxima del cabezal 1 al 90% de 1000 r/min.
...	
N50 VELOLIM[C]=45	; Limitación de la velocidad de giro máxima del cabezal 1 al 45% de 1000 r/min, siendo C el identificador de eje de S1.
...	

3.14.10 Programar tolerancia de contorno/orientación (CTOL, OTOL, ATOL)

Con las direcciones CTOL, OTOL y ATOL pueden adaptarse en el programa de pieza las tolerancias de mecanizado parametrizadas mediante datos de máquina y de operador para funciones de compresor, matado de esquinas y alisado de la orientación.

Los valores de tolerancia programados se aplican hasta que se reprograman o se borran mediante la asignación de un valor negativo. Además se borran al final del programa o con un reset. Después del borrado tienen efecto nuevamente los valores de tolerancia parametrizados.

Sintaxis

```
CTOL=<Value>
OTOL=<Value>
ATOL[<Axis>]=<Value>
```

Descripción

CTOL:	Dirección para programar la **tolerancia de contorno**			
	Campo de aplicación:	• todas las funciones de compresor • todos los tipos de matado de esquinas excepto G641 y G644		
	Parada de decodificación previa:	No		
	Efecto:	modal		
	<Value>:	El valor para la tolerancia de contorno es una indicación de longitud.		
		Tipo:	REAL	
		Unidad:	Pulgadas/mm (en función del ajuste actual del acotado)	
		Rango de valores:	≥ 0:	Valor de tolerancia
			< 0:	Borrado del valor de tolerancia programado ⇒ Se aplica nuevamente el valor de tolerancia parametrizado en el dato de máquina o de operador.
OTOL:	Dirección para programar la **tolerancia de orientación**			
	Campo de aplicación:	• todas las funciones de compresor • el alisado de la orientación ORISON • todos los tipos de matado de esquinas excepto G641, G644 y OSD		
	Parada de decodificación previa:	No		
	Efecto:	modal		
	<Value>:	El valor para la tolerancia de la orientación es una especificación de ángulo.		
		Tipo:	REAL	
		Unidad:	grados	
		Rango de valores:	≥ 0:	Valor de tolerancia
			< 0:	Borrado del valor de tolerancia programado ⇒ Se aplica nuevamente el valor de tolerancia parametrizado en el dato de máquina o de operador.

ATOL:	Dirección para programar una **tolerancia específica de eje**			
	Campo de aplicación:	• todas las funciones de compresor • el alisado de la orientación ORISON • todos los tipos de matado de esquinas excepto G641, G644 y OSD		
	Parada de decodificación previa:	No		
	Efecto:	modal		
	<Axis>:	Nombre del eje de canal al que ha de aplicarse la tolerancia programada		
	<Value>:	El valor de la tolerancia de eje es una indicación de longitud o una especificación de ángulo, en función del tipo de eje (lineal o giratorio).		
		Tipo:	REAL	
		Unidad:	para ejes lineales:	Pulgadas/mm (en función del ajuste actual del acotado)
			para ejes giratorios:	Grados
		Rango de valores:	≥ 0:	Valor de tolerancia
			< 0:	Borrado del valor de tolerancia programado ⇒ Se aplica nuevamente el valor de tolerancia parametrizado en el dato de máquina o de operador.

Nota

Los valores de tolerancia específicos de canal programados con CTOL y OTOL tienen mayor prioridad que los valores de tolerancia específicos de eje programados con ATOL.

Nota

Frames de escala

Los frames de escala actúan sobre las tolerancias programadas de la misma forma que sobre las posiciones de eje, es decir, la tolerancia relativa se mantiene igual.

Ejemplo

```
Código de programa                  Comentarios
COMPCAD G645 G1 F10000              ; Activar la función de compresor COMPCAD.
X... Y... Z...                      ; Aquí actúan los datos de máquina y de operador.
X... Y... Z...
X... Y... Z...
CTOL=0.02                           ; A partir de aquí actúa una tolerancia de contorno
                                    de 0,02 mm.
X... Y... Z...
```

Código de programa	Comentarios
X... Y... Z...	
X... Y... Z...	
ASCALE X0.25 Y0.25 Z0.25	; A partir de aquí actúa una tolerancia de contorno de 0,005 mm.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
CTOL=-1	; A partir de aquí vuelven a actuar los datos de máquina y de operador.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

Variables del sistema

Lectura con parada de decodificación previa

Las tolerancias actuales activas en el programa de pieza y en la acción síncrona pueden leerse mediante las siguientes variables del sistema:

- $AC_CTOL
  Tolerancia de contorno específica de canal que estaba activa en el momento de preparar la secuencia del proceso principal actual.
  Si no hay activa ninguna tolerancia de contorno, $AC_CTOL suministra la raíz de la suma de los cuadrados de las tolerancias de los ejes geométricos.
- $AC_OTOL
  Tolerancia de orientación específica de canal que estaba activa en el momento de preparar la secuencia del proceso principal actual
  Si no hay activa ninguna tolerancia de orientación, $AC_OTOL suministra la raíz de la suma de los cuadrados de las tolerancias de los ejes de orientación durante una transformada de orientación activa; de lo contrario, suministra el valor "-1".
- $AA_ATOL[<Eje>]
  Tolerancia de contorno específica de eje que estaba activa en el momento de preparar la secuencia del proceso principal actual
  Si hay activa una tolerancia de contorno, $AA_ATOL[<Eje geométrico>] suministra la tolerancia de contorno dividida entre la raíz del número de ejes geométricos.
  Si hay una tolerancia de orientación y una transformada de orientación activa, $AA_ATOL[<Eje de orientación>] suministra la tolerancia de orientación dividida entre la raíz del número de ejes de orientación.

Nota

Si no se han programado valores de tolerancia, las variables $A no estarán lo suficientemente diferenciadas para distinguir las tolerancias de cada una de las funciones.

Estos casos pueden darse cuando los datos de máquina y los de operador ajustan diferentes tolerancias para funciones de compresor, matado de esquinas y alisado de orientación. Las variables del sistema suministran entonces el valor mayor que aparece con las funciones que están activas en ese momento. Si está activa, p. ej., una función de compresor con tolerancia de orientación 0,1° y un alisado de orientación ORISON con 1°, $AC_OTOL suministra el valor "1". Si el alisado de orientación se desactiva, $AC_OTOL suministra solo el valor "0,1".

Lectura sin parada de decodificación previa

Las tolerancias actuales activas en el programa de pieza pueden leerse mediante las siguientes variables del sistema:

- $P_CTOL
  Tolerancia de contorno específica de canal activa actualmente.
- $P_OTOL
  Tolerancia de orientación específica de canal activa actualmente.
- $PA_ATOL
  Tolerancia de contorno específica de eje activa actualmente.

Condiciones

Las tolerancias programadas con CTOL, OTOL y ATOL tienen efecto también en funciones que dependen indirectamente de estas tolerancias:

- La limitación del error de contorno en el cálculo de la consigna
- Describe las funciones básicas del modo de superficies de forma libre

La programación de CTOL, OTOL y ATOL **no** influye en las siguientes funciones de matado de esquinas:

- Matado de esquinas de la orientación con OSD
  OSD no utiliza tolerancias sino una distancia para la transición de secuencia.
- Matado de esquinas con G644
  G644 no se utiliza para el mecanizado sino para optimizar cambios de herramienta y otros movimientos sin mecanizado.
- Matado de esquinas con G645
  G645 tiene casi siempre el mismo comportamiento que G642 y utiliza, por tanto, las tolerancias programadas. El valor de tolerancia del dato de máquina MD33120 $MA_PATH_TRANS_POS_TOL se utiliza solo en las transiciones de secuencia con tangente continua y escalón de curvatura, p. ej., una transición tangencial circunferencia-recta. En estos puntos, el recorrido de matado de esquinas puede estar situado en el lado exterior del contorno programado, donde la tolerancia de muchas aplicaciones es más pequeña. Además, para compensar los cambios de curvatura es suficiente generalmente una tolerancia pequeña y fija de la que el programador CN no tenga que preocuparse más.

3.14.11 Comportamiento en cambio de secuencia con acoplamiento activo (CPBC)

El comando CPBC especifica el criterio de cambio se secuencia que debe cumplirse para que se realice en cambio de secuencia en el programa de pieza cuando está activo el acoplamiento.

Sintaxis

```
CPBC[<Eje esclavo>] = <Criterio>
```

Descripción

CPBC:	Criterio de cambio de secuencia con acoplamiento activo	
<Eje esclavo>:	Identificador del eje esclavo	
<Criterio>:	Criterio de cambio de secuencia	
	Tipo:	STRING
	Valor	**Significado:** Cambio de secuencia realizado
	"NOC"	independiente del estado de acoplamiento
	"IPOSTOP"	con la sincronización del valor de consigna
	"COARSE"	con la sincronización del valor real "basta"
	"FINE"	con la sincronización del valor real "fina"

Ejemplo

Código de programa

```
; cambio de secuencia realizado con:
;  - acoplamiento con eje esclavo X2 == activo
;  - sincronización del valor de consigna == activo
CPBC[X2]="IPOSTOP"
```

3.15 Funciones de ejes

3.15.1 Intercambio de eje, intercambio de cabezales (RELEASE, GET, GETD)

Uno o varios ejes o cabezales se pueden interpolar siempre en un solo canal. Si un eje ha de trabajar alternadamente en dos canales diferentes (p. ej., intercambiador de palets), deberá liberarse primero en el canal actual y ser aceptado luego en el otro canal. El eje se intercambia entre los canales.

Ampliaciones del intercambio de ejes

Un eje/cabezal se puede intercambiar con parada de decodificación previa y sincronización entre el preprocesamiento y la marcha principal o, como alternativa, también sin parada de decodificación. Además, el intercambio de ejes también es posible a través de:

- Giro del contenedor de ejes AXCTSWE o AXCTWED con `GET/GETD` implícito.
- Frame con rotación si el eje en cuestión está enlazado de esta forma con otros ejes.
- Acciones síncronas, ver acciones síncronas de desplazamientos, "Intercambio de ejes `RELEASE`, `GET`".

Fabricante de la máquina

Preste atención a las indicaciones del fabricante de la máquina. Para el intercambio de ejes, un eje tiene que estar definido unívocamente en todos los canales a través de datos de máquina configurables, y el comportamiento de intercambio también puede ajustarse mediante la modificación de datos de máquina.

Sintaxis

```
RELEASE (nombre de eje, nombre de eje, ...) o RELEASE (S1)
GET (nombre de eje, nombre de eje, ...) o GET (S2)
GETD(nombre de eje, nombre de eje, ...) o GETD(S3)
```

Con el comando GETD (GET Directly) se toma directamente un eje desde otro canal. Es decir, para este GETD no hay que programar el correspondiente comando RELEASE en otro canal. No obstante, esto significa que se precisa entonces establecer otra comunicación de canal (p. ej., metas de espera).

Descripción

`RELEASE (nombre eje, nombre eje, …):`	Libera los ejes
`GET (nombre eje, nombre eje, …):`	Aplica los ejes
`GETD (nombre eje, nombre eje, …):`	Aplica directamente los ejes
`Nombre eje:`	Asignación de ejes en el sistema: AX1, AX2, … o indicación de los nombres de eje de máquina
`RELEASE(S1):`	Libera los cabezales S1, S2, ...
`GET(S2):`	Aplica los cabezales S1, S2, ...
`GETD(S3):`	Aplicación directa de los cabezales S1, S2, ...

Solicitud GET sin parada de decodificación previa

Si, después de una solicitud GET **sin** parada de decodificación previa, se vuelve a liberar el eje con `RELEASE(eje)` o `WAITP(eje)`, un `GET` posterior produce un `GET` **con** parada de decodificación.

⚠ PRECAUCIÓN

Asignación de ejes, modificada

Un eje o cabezal aplicado con el comando GET permanece asignado a este canal aún tras un RESET por tecla o por programa.

Al arrancar nuevamente el programa, los ejes o cabezales cambiados se deben asignar a través del programa en caso de que el eje se requiera en su canal básico.

Con POWER ON se asigna el eje al canal ajustado en los datos de máquina.

Ejemplos

Ejemplo 1: Intercambio de ejes entre dos canales

De 6 ejes se utilizan en el canal 1 para el mecanizado: el 1.º, 2.º, 3.º y 4.º eje.
El 5.º y el 6.º eje se utilizan en el canal 2 para el cambio de pieza.

El eje 2 debe poder intercambiarse entre ambos canales y estar asignado al canal 1 tras POWER ON.

Programa "MAIN" en el canal 1:

Código de programa	**Comentarios**
`INIT (2,"CAMBIAR2")`	`; Seleccionar el programa CAMBIAR2 en el canal 2.`
`N… START (2)`	`; Iniciar el programa en el canal 2.`
`N… GET (AX2)`	`; Aceptar el eje AX2.`
`...`	
`N… RELEASE (AX2)`	`; Liberar el eje AX2.`
`N... WAITM(1,1,2)`	`; Esperar la marca WAIT en los canales 1 y 2 para efectuar la sincronización en ambos canales.`
`...`	`; Continuación del proceso tras el cambio de eje.`
`N… M30`	

Programa "CAMBIAR2" en el canal 2:

Programación	**Comentarios**
`N… RELEASE (AX2)`	
`N160 WAITM(1,1,2)`	`; Esperar la marca WAIT en los canales 1 y 2 para efectuar la sincronización en ambos canales.`
`N150 GET(AX2)`	`; Aceptar el eje AX2.`
`...`	`; Continuación del proceso tras el cambio de eje.`

Programación	Comentarios
N… M30	

Ejemplo 2: Intercambio de ejes sin sincronización

Si el eje no tiene que ser sincronizado, GET no genera parada de decodificación previa.

Programación	Comentarios
N01 G0 X0	
N02 RELEASE(AX5)	
N03 G64 X10	
N04 X20	
N05 GET(AX5)	; Si no se requiere una sincronización, esta secuencia no será ejecutable.
N06 G01 F5000	; Secuencia no ejecutable.
N07 X20	; Secuencia no ejecutable, posición X como en N04.
N08 X30	; Primera secuencia ejecutable tras N05.
...	

Ejemplo 3: Activación de un intercambio de ejes sin parada de decodificación previa

Requisitos: El intercambio de ejes sin parada de decodificación previa debe configurarse a través de un dato de máquina.

Programación	Comentarios
N010 M4 S100	
N011 G4 F2	
N020 M5	
N021 SPOS=0	
N022 POS[B]=1	
N023 WAITP(B)	; El eje B pasa a ser el eje neutro.
N030 X1 F10	
N031 X100 F500	
N032 X200	
N040 M3 S500	; El eje no inicia parada de decodificación previa/ REORG.
N041 G4 F2	
N050 M5	
N099 M30	

Si el cabezal o el eje B se desplaza inmediatamente después de la secuencia N023 como **eje PLC**,
p. ej., a 180 grados y de vuelta a 1 grado, este eje se vuelve a convertir en eje neutro y no produce ninguna parada de decodificación previa en la secuencia N40.

Información adicional

Requisitos para el intercambio de ejes

- El eje tiene que estar definido a través de datos de máquina en todos los canales que utilicen el eje.
- A través del dato de máquina específico del **eje**se tiene que definir a qué canal se asignará el eje después de POWER ON.

Descripción

Liberar eje: RELEASE

Al liberar el eje se ha de tener en cuenta lo siguiente:

1. El eje no puede estar activo en ninguna transformada.
2. En los acoplamientos de ejes (control tangencial) se deben desbloquear todos los ejes del conjunto.
3. Un eje de posicionado concurrente no se puede intercambiar en este estado.
4. En caso de un eje maestro Gantry se intercambian también todos los ejes esclavos.
5. En acoplamientos de ejes (arrastre, acoplamiento de valor maestro, reductor electrónico) sólo se puede desbloquear el eje maestro del conjunto.

Aceptar eje: GET

Con este comando tiene lugar el intercambio de ejes propiamente dicho. La responsabilidad sobre el eje reside completamente en el canal donde fue programado el comando.

Efectos de GET:

Intercambio de eje con sincronización:

Un eje tendrá que ser sincronizado siempre que haya sido asignado entretanto a otro canal o al PLC y antes de GET no haya sido efectuada una sincronización a través de "WAITP", G74 o por borrado de trayecto residual.

- Se ejecuta una parada de decodificación previa (como en STOPRE).
- El mecanizado se interrumpe hasta que haya concluido completamente el intercambio.

"GET" automático

Si un eje está disponible por principio en un canal, pero no existe actualmente como "eje de canal", se ejecuta automáticamente el comando GET. Si los ejes ya están sincronizados, no se genera ninguna parada de decodificación previa.

Ajuste modificable del comportamiento de intercambio de ejes

El momento de entrega de ejes se puede ajustar como sigue a través de un dato de máquina:

- El intercambio automático de ejes también se produce entre dos canales si el eje ha sido llevado mediante WAITP a un estado neutro (comportamiento como antes)
- Al hacer la solicitud de giro del contenedor de ejes, todos los ejes del contenedor que pueden asignarse al canal que realiza la ejecución se recogen en el canal mediante un comando implícito GET o GETD. Sólo se permite un intercambio de ejes subsiguiente una vez que ha finalizado el giro del contenedor de ejes.
- Se comprueba después de una secuencia intermedia intercalada en el proceso principal si es necesaria o no una reorganización. Sólo si los estados de ejes de dicha secuencia **no** coinciden con los estados de ejes actuales es necesario proceder a una reorganización.
- En lugar de una secuencia GET con parada de decodificación previa y sincronización entre el preprocesamiento y el proceso principal, un intercambio de ejes también se puede realizar sin parada de decodificación previa. Entonces sólo se genera una secuencia intermedia con la solicitud GET. En el proceso principal se comprueba, en la ejecución de esta secuencia, si los estados del eje en la secuencia coinciden con los estados actuales del eje.

3.15.2 Transferir el eje a otro canal (AXTOCHAN)

Con la instrucción de programación (comando) `AXTOCHAN` se puede solicitar un eje para transferirlo a otro canal. El eje se puede traer al correspondiente canal desde el programa de pieza CN, así como desde una acción síncrona.

Sintaxis

```
AXTOCHAN(nombre de eje,nombre de canal[,nombre de eje,número de
canal[,...]])
```

Descripción

Elemento	Descripción
`AXTOCHAN`:	Solicitar un eje para un determinado canal
`Nombre eje`:	Asignación de ejes en el sistema: X, Y,… o indicación de los nombres de eje de máquina afectados. El canal que realiza la ejecución no necesita ser el canal propio ni tampoco el canal que posee actualmente el derecho de interpolación para el eje
`Número de canal`:	Número del canal al que debe asignarse el eje

Nota

Eje de posicionado concurrente y eje controlado exclusivamente por el PLC

Un eje PLC no puede cambiar de canal como eje de posicionado concurrente. Un eje controlado exclusivamente por el PLC no se puede asignar al programa CN.

Información adicional: Manual de funciones Ejes y cabezales

Ejemplo

AXTOCHAN en el programa CN

Los ejes X e Y son conocidos en el 1.er y en el 2.º canal. Actualmente, el canal 1 tiene el derecho de interpolación y en el canal 1 se inicia el siguiente programa:

Código de programa	Comentarios
N110 AXTOCHAN(Y,2)	; Desplazar el eje Y al 2.° canal.
N111 M0	
N120 AXTOCHAN(Y,1)	; Recuperar el eje Y (neutro).
N121 M0	
N130 AXTOCHAN(Y,2,X,2)	; Desplazar el eje Y y el eje X al 2.° canal (ejes neutros).
N131 M0	
N140 AXTOCHAN(Y,2)	; Desplazar el eje Y al 2.° canal (programa CN).
N141 M0	

Información adicional

AXTOCHAN en el programa CN

En este caso, sólo se ejecuta un `GET` si se realiza una solicitud del eje para el programa CN en el canal propio, esperando así también el cambio de estado efectivo. Si el eje se solicita para otro canal o se tiene que convertir en el eje neutro en el canal propio, entonces sólo se envía la correspondiente solicitud.

AXTOCHAN desde una acción síncrona

Si se solicita un eje para el canal propio, `AXTOCHAN` se refleja en un `GET` de una acción síncrona desde una acción síncrona. En este caso, el eje se convierte en eje neutro en la primera solicitud para el canal propio. Con la segunda solicitud, el eje se asigna al programa CN de forma análoga a la petición GET en el programa CN. Sobre la solicitud GET desde una acción síncrona, ver el apartado "Acciones síncronas a desplazamientos".

3.15.3 Funciones de eje (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)

"AXNAME" se utiliza, p. ej., para la creación de ciclos de validez general, cuando los nombres de los ejes se desconocen.

"AX" se utiliza para la programación indirecta de ejes geométricos y síncronos. El identificador de eje se guarda en una variable del tipo AXIS o se suministra mediante un comando como "AXNAME" o "SPI".

"SPI" se utiliza cuando se programan funciones de eje para un cabezal, p. ej., un cabezal síncrono.

"AXTOSPI" se utiliza para convertir un identificador de eje en un índice de cabezal (función inversa de "SPI").

"AXSTRING" se utiliza para convertir un identificador de eje (tipo de datos AXIS) en un string (función inversa de "AXNAME").

"ISAXIS" se aplica a los ciclos de propósito general para garantizar que se dispone de determinados ejes geométricos y evitar que se interrumpa con error la llamada siguiente de $P_AXNX.

"MODAXVAL" se utiliza para determinar la posición módulo en ejes giratorios con módulo.

Sintaxis

```
AXNAME("String")
AX[AXNAME("String")]
SPI(n)

AXTOSPI(A),AXTOSPI(B) o AXTOSPI(C)
AXSTRING(SPI(n))
ISAXIS(<Número de eje geométrico>)
<Posición módulo>=MODAXVAL(<Eje>,<Posición de eje>)
```

Descripción

AXNAME:	Conversión de una cadena de caracteres en identificador de eje; la cadena de caracteres debe estar constituida por un nombre de un eje válido.
AX:	Identificador de eje variable
SPI:	Conversión de un número de cabezal en identificador de eje; el parámetro de transferencia debe ser un número de cabezal válido.
n:	Número de cabezal
AXTOSPI:	Convierte un identificador de eje en un índice de cabezal del tipo Integer. "AXTOSPI" es la función inversa de "SPI".
X, Y, Z:	Identificador de eje del tipo AXIS como variable o constante
AXSTRING:	Se emite la cadena con el número de cabezal correspondiente.
ISAXIS:	Comprobar si existe el eje geométrico introducido.
MODAXVAL:	Determina la posición módulo en ejes giratorios con módulo; esta posición equivale al resto de módulo respecto a la zona de módulo parametrizada (entre 0 y 360 grados en el ajuste estándar; mediante DM30340 MODULO_RANGE_START y DM30330 $MA_MODULO_RANGE pueden modificarse el inicio y el tamaño de la zona de módulo).

Nota

Ampliaciones SPI

La función de eje SPI(n) también puede utilizarse para la lectura y la escritura de componentes frame. Ello permite, por ejemplo, escribir frames con la sintaxis `$P_PFRAME[SPI(1),TR]=2.22`.

Mediante la programación adicional de posiciones de eje a través de la dirección `AX[SPI(1)]=<Posición de eje>` puede desplazarse un eje. El requisito para ello es que el cabezal se encuentre en modo de posicionamiento o de eje.

Ejemplos

Ejemplo 1: AXNAME, AX, ISAXIS

Código de programa	Comentarios
OVRA[AXNAME("eje refrentado")]=10	; Corrección para el eje de refrentado
AX[AXNAME("eje refrentado")]=50.2	; Posición final para el eje de refrentado
OVRA[SPI(1)]=70	; Corrección para cabezal 1
AX[SPI(1)]=180	; Posición final para cabezal 1
IF ISAXIS(1) == FALSE GOTOF CONTINUAR	; ¿Existe la abscisa?
AX[$P_AXN1]=100	; Desplazar en abscisas
CONTINUAR:	

Ejemplo 2: AXSTRING

En la programación con AXSTRING[SPI(n)] ya no se emite el índice del eje asignado al cabezal como número de cabezal, sino la cadena "Sn".

Código de programa	Comentarios
AXSTRING[SPI(2)]	; Se emite la cadena "S2".

Ejemplo 3: MODAXVAL

Debe determinarse la posición de módulo del eje giratorio A con módulo.

El valor inicial para el cálculo es la posición de eje 372.55.

La zona de módulo parametrizada va de los 0 hasta los 360 grados:

DM30340 MODULO_RANGE_START = 0

DM30330 $MA_MODULO_RANGE = 360

Código de programa	Comentarios
R10=MODAXVAL(A,372.55)	; Posición módulo calculada R10 = 12.55.

Ejemplo 4: MODAXVAL

Cuando el identificador de eje programado no se refiere a un eje giratorio con módulo, entonces se devuelve el valor a convertir (<Posición de eje>) sin modificar.

Código de programa	Comentarios
R11=MODAXVAL(X,372.55)	; X es el eje lineal; R11 = 372.55.

3.15.4 Ejes geométricos conmutables (GEOAX)

Con la función "Ejes geométricos conmutables" se pueden sustituir los ejes geométricos parametrizados mediante datos de máquina por otros ejes de canal.

Sintaxis

```
GEOAX(<n>,<Eje de canal>,<n>,<Eje de canal>,<n>,<Eje de canal>)
```

```
GEOAX()
```

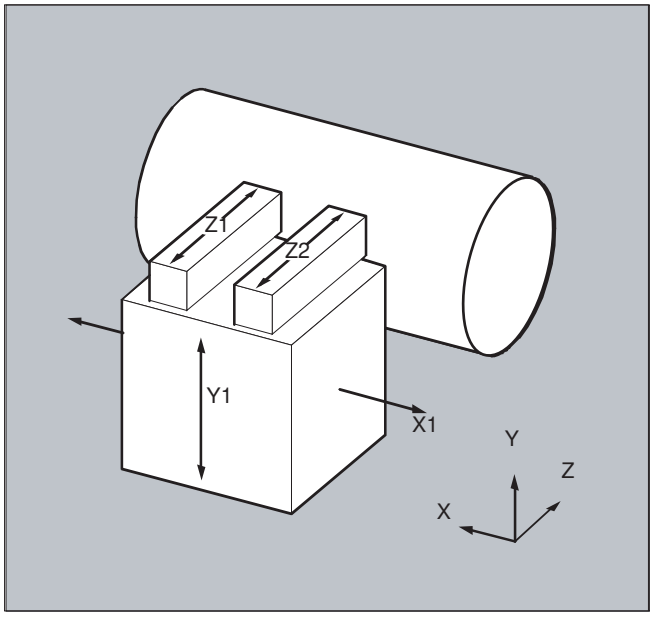
Descripción

`GEOAX(...)`	Función para conmutar ejes geométricos. **Nota:** `GEOAX()` sin indicación de parámetros vuelve a activar la configuración básica de los ejes geométricos parametrizada en los datos de máquina.
`<n>`	Número del eje geométrico que se sustituye por el eje de canal indicado. Rango de valores: 0, 1, 2, 3 **Nota:** 0: el eje de canal indicado se retira del conjunto de ejes geométricos sin ser sustituido 1: 1. eje geométrico ≙ eje de coordenadas X (abscisa) del WKS 2: 2. eje geométrico ≙ eje de coordenadas Y (ordenada) del WKS 3: 3. eje geométrico ≙ eje de coordenadas Z (aplicada) del WKS
`<Eje de canal>`	Nombre del eje de canal que debe adoptarse en el conjunto de ejes geométricos.

Ejemplos

Ejemplo 1: Activar dos ejes alternativamente como eje geométrico

Un carro de herramienta se puede desplazar a través de los ejes de canal X1, Y1, Z1, Z2:



Los ejes geométricos están configurados de tal manera que, tras la activación, primero es efectivo Z1 como 3.er eje geométrico con el nombre de eje geométrico "Z", que forma junto con X1 e Y1 el conjunto de ejes geométricos.

En el programa de pieza deben entrar ahora en acción los ejes Z1 y Z2 alternativamente como eje geométrico Z:

Código de programa	Comentarios
...	
N100 GEOAX(3,Z2)	; Como 3.er eje geométrico (Z) actúa el eje de canal Z2.
N110 G1 ...	
N120 GEOAX(3,Z1)	; Como 3.er eje geométrico (Z) actúa el eje de canal Z1.
...	

Ejemplo 2: Conmutación de los ejes geométricos en 6 ejes de canal

Una máquina tiene 6 ejes de canal con los nombres XX, YY, ZZ, U, V y W.

El ajuste inicial de la configuración de ejes geométricos a través de datos de máquina es el siguiente:

Eje de canal XX = 1er eje geométrico (eje X)

Eje de canal YY = 2º eje geométrico (eje Y)

Eje de canal ZZ = 3er eje geométrico (eje Z)

Código de programa	Comentarios
N10 GEOAX()	; La configuración básica de los ejes geométricos es efectiva.
N20 G0 X0 Y0 Z0 U0 V0 W0	; Todos los ejes en Rápido en la posición 0.
N30 GEOAX(1,U,2,V,3,W)	; El eje de canal U será el primer eje geométrico (X), V el segundo (Y)
	; y W el tercero (Z).
N40 GEOAX(1,XX,3,ZZ)	; El eje de canal XX será el primer eje geométrico (X) y ZZ será el tercer
	; eje geométrico (Z). El eje de canal V se mantiene como segundo
	; eje geométrico (Y).
N50 G17 G2 X20 I10 F1000	; Círculo en el plano X/Y. Se desplazan los
	; ejes de canal XX y V.
N60 GEOAX(2,W)	; El eje de canal W se convierte en segundo eje geométrico (Y).
N80 G17 G2 X20 I10 F1000	; Círculo en el plano X/Y. Se desplazan los
	; ejes de canal XX y W.
N90 GEOAX()	; Vuelta al estado inicial.
N100 GEOAX(1,U,2,V,3,W)	; El eje de canal U será el primer eje geométrico (X), V el segundo
	; (Y) y W el tercero (Z).
N110 G1 X10 Y10 Z10 XX=25	; Los ejes de canal U, V y W se desplazan respectivamente a la
	; posición 10. XX como eje adicional se desplaza a la posición 25.

Código de programa	Comentarios
N120 GEOAX(0,V)	; V se elimina del conjunto de ejes geométricos.
	; U y W siguen siendo el primer (X) y tercer
	; eje geométrico (Z).
	; El segundo eje geométrico (Y) se mantiene sin ocupar.
N130 GEOAX(1,U,2,V,3,W)	; El eje de canal U se mantiene como primer eje geométrico (X), V será el segundo
	; (Y) y W se mantiene como tercero (Z).
N140 GEOAX(3,V)	; V será el tercer eje geométrico (Z), al tiempo que W
	; se sobrescribe, con lo que será retirado del conjunto
	; de ejes geométricos. El segundo eje geométrico (Y)
	; sigue estando desocupado.

Datos de máquina

Configuración de ejes

Asignación de ejes geométricos, adicionales y de máquina a ejes de canal:

- MD10000 $MN_AXCONF_MACHAX_NAME_TAB
- MD20050 $MC_AXCONF_GEOAX_ASIGN_TAB
- MD20060 $MC_AXCONF_GEOAX_NAME_TAB
- MD20070 $MC_AXCONF_MACHAX_USED
- MD20080 $MC_AXCONF_CHANAX_NAME_TAB
- MD35000 $MA_SPIND_ASSIGN_TO_MACHAX

Comportamiento Reset

Comportamiento Reset de asignaciones modificadas de ejes geométricos:

- MD20110 $MC_RESET_MODE_MASK, bit 12
- MD20118 $MC_GEOAX_CHANGE_RESET

Comportamiento de marcha CN

- MD20112 $MC_START_MODE_MASK, bit 12

Notificación al programa de usuario PLC

Posible parametrización del comando M que se emite en la interfaz CN/PLC en caso de conmutación de ejes geométricos:

- MD22532 $MC_GEOAX_CHANGE_M_CODE

Condiciones

No hay conmutación de ejes geométricos

- Si una de las siguientes funciones está activa, no es posible conmutar los ejes geométricos:
  - Transformada
  - Interpolación spline
  - Corrección de radio de herramienta
  - Corrección de herramienta fina
- El eje geométrico y otro eje de canal tienen el mismo nombre.
- Uno de los ejes implicados en la conmutación de ejes geométricos participa en una acción que perdura más allá de los límites de secuencia. P. ej., eje de posicionado modalmente activo o eje conducido de un acoplamiento de ejes.

Ejes giratorios

Los ejes giratorios no pueden pasar a ser ejes geométricos.

Estado del eje tras la sustitución

Un eje sustituido mediante la conmutación en el conjunto de ejes geométricos puede programarse como eje adicional tras la operación de conmutación mediante su nombre de eje de canal.

Frames, zonas protegidas, limitaciones de zona de trabajo

Con la conmutación de los ejes geométricos se suprimen todos los frames, zonas protegidas y limitaciones de zona de trabajo.

Coordenadas polares

Un cambio de los ejes geométricos con `GEOAX`, de forma análoga a un cambio de nivel con `G17-G19`, define las coordenadas polares modales al valor 0.

DRF, DO

Un posible decalaje con volante (DRF) o un decalaje de origen externo (DO) sigue siendo efectivo tras la conmutación.

Configuración básica de los ejes geométricos

El comando `GEOAX()` llama la configuración básica del conjunto de ejes geométricos.

Tras POWER ON y al conmutar al modo de operación "Posicionar referencia" se vuelve automáticamente a la configuración básica.

Corrección longitudinal de herramienta

Una corrección longitudinal de herramienta activa sigue siendo efectiva tras la operación de conmutación. Sin embargo, para los ejes geométricos nuevos o cambiados de posición vale como aún no aplicada. En el primer comando de movimiento para estos ejes geométricos, el trayecto resultante se compone por lo tanto de la suma de la corrección longitudinal de herramienta y del trayecto programado.

Los ejes geométricos que mantienen su posición en el conjunto de ejes tras una conmutación, también mantienen su estado en lo que a la corrección longitudinal de herramienta se refiere.

Configuración de eje geométrico durante una transformada activa

- La configuración de ejes geométricos parametrizada mediante datos de máquina de transformada para una transformada activa no puede modificarse mediante la función "Ejes geométricos conmutables".
- Diferente configuración de ejes geométricos para una transformada: deben parametrizarse diferentes registros en los datos de máquina de la transformada.
- Una configuración de ejes geométricos modificada mediante `GEOAX` se borra al activar una transformada.
- En lo que respecta a los ejes geométricos, las parametrizaciones de ejes geométricos específicas de la transformada en transformadas activas tienen preferencia frente a las parametrizaciones que afectan a la conmutación de ejes geométricos.
  Ejemplo: Una transformada está activa. Según los datos de máquina, la transformada debe mantenerse al realizarse un reset de canal. Sin embargo, al mismo tiempo, en caso de reset de canal debe establecerse la configuración básica de los ejes geométricos. Se mantiene la configuración de los ejes geométricos definida para la transformada.
- Al desactivar una transformada vuelve a ser efectivo el ajuste básico parametrizado de la configuración de los ejes geométricos.

Modo de operación JOG, función de máquina REF

Al conmutar al modo de operación JOG, función de máquina REF (búsqueda del punto de referencia), será efectiva la configuración de ejes geométricos parametrizada en los datos de máquina.

3.15.5 Contenedor de ejes (AXCTSWE, AXCTSWED, AXCTSWEC)

Mediante los comandos "AXCTSWE" y "AXCTSWED" se habilita el giro del contenedor de ejes especificado.

El comando "AXCTSWEC" anula nuevamente una habilitación del giro del contenedor de ejes.

Sintaxis

```
AXCTSWE(<ID>)
AXCTSWED(<ID>)
AXCTSWEC(<ID>)
```

Descripción

AXCTSWE:	Habilitación de giro del contenedor de ejes El procesamiento del programa no se detiene con "AXCTSWE". El giro se produce cuando todos los canales con ejes en el contenedor han enviado la habilitación.	
AXCTSWED:	Habilitación de giro del contenedor de ejes sin inclusión de los otros canales con ejes en el contenedor **Nota** • Variante de comando para simplificar la puesta en marcha del programa de pieza o acción síncrona. • El comportamiento con relación a los otros canales con ejes en el contenedor puede especificarse mediante: MD12760 $MN_ AXCT_FUNCTION_MASK, bit 0	
AXCTSWEC:	Anulación de la habilitación de giro del contenedor de ejes **Nota** La habilitación del giro del contenedor de ejes solo se puede anular si el giro todavía no ha comenzado: $AN_AXCTSWA[<Contenedor de ejes>] == 0 Para la variable de sistema, ver "Contenedor de ejes (AXCTSWE, AXCTSWED, AXCTSWEC) (Página 911)"	
<ID>:	Identificador del contenedor de ejes o de un eje del contenedor:	
	CT<Número>:	Identificador predeterminado de un contenedor de ejes: MD12750 $MN_AXCT_NAME_TAB Ejemplo: "CT1"
	<Contenedor>:	Identificador específico de usuario de un contenedor de ejes: MD12750 $MN_AXCT_NAME_TAB Ejemplo: "CONTENEDOR_1"
	<Eje>:	Identificador de un eje del contenedor conocido en el canal

Nota

Incremento

El incremento de un giro del contenedor de ejes se ajusta mediante el dato de operador:

SD41700 $SN_AXCT_SWWIDTH

Información adicional

Diagnóstico

El estado actual de un contenedor de ejes puede leerse por medio de la siguiente variable de sistema:

Variable del sistema	Tipo	Descripción
$AC_AXCTSWA[<Nombre>]	BOOL	Estado del contenedor de ejes específico de canal
$AN_AXCTSWA[<Contenedor de ejes>]	BOOL	Estado del contenedor de ejes específico de la NCU

Variable del sistema	Tipo	Descripción
$AN_AXCTSWE[<Contenedor de ejes>]	INT	Estado de giro del contenedor de ejes específico de slot La variable de sistema comunica **bit a bit** el estado de los slots del contenedor de ejes. Cada bit corresponde a un slot.
$AN_AXCTAS[<Contenedor de ejes>]	INT	Número de puestos (slots) por los que ha pasado actualmente el contenedor de ejes.

Giro del contenedor de ejes con GET/GETD implícito

A través del siguiente dato de máquina, se ajusta si el comando "AXCTSWE" trae al canal todos los ejes de contenedor del canal mediante "GET/GETD" implícito. El intercambio de eje no es posible hasta que el contenedor ha girado.

MD10722 $MN_AXCHANGE_MASK, bit 1 = 1

Nota

El giro del contenedor de ejes con "GET/GETD" implícito **no** se ejecuta para un eje con estado "Eje de marcha principal" (p. ej., eje PLC) porque el eje debería salir de este estado para permitir el giro del contenedor.

3.15.6 Esperar a posición de eje válida (WAITENC)

Con el comando de lenguaje "WAITENC" se puede esperar en el programa CN hasta que estén disponibles posiciones de eje sincronizadas o restauradas para los ejes configurados con MD34800 $MA_WAIT_ENC_VALID = 1.

En el estado de espera puede tener lugar una interrupción, p. ej. por inicio de un ASUP o por cambio de modo de operación tras JOG. Al reanudar el programa se vuelve a adoptar el estado de espera, dado el caso.

Nota

El estado de espera se indica en la interfaz de usuario con el estado de parada "Esperar sistema de medida".

Sintaxis

"WAITENC" se puede programar en la parte de programación de un programa CN cualquiera.

La programación debe realizarse en una secuencia propia:

```
...
WAITENC
...
```

Ejemplo

"WAITENC" se utiliza, p. ej., en el programa de usuario controlado por sucesos .../_N_CMA_DIR/_N_PROG_EVENT_SPF, como puede verse en el ejemplo siguiente.

Ejemplo de aplicación: retirada de herramienta tras POWER OFF con transformada de orientación

Se ha interrumpido un mecanizado con orientación de herramienta debido a un fallo de la alimentación.
Al volver a arrancar se llamará al programa de usuario controlado por sucesos .../_N_CMA_DIR/_N_PROG_EVENT_SPF.

En el programa de usuario controlado por sucesos se espera con "WAITENC" a disponer de posiciones de eje sincronizadas o restauradas, para poder calcular entonces un frame que oriente el WKS en la dirección de la herramienta.

```
Código de programa                  Comentarios
...
IF $P_PROG_EVENT == 4               ; Arranque.
    IF $P_TRAFO <> 0                ; Se ha seleccionado una transformada.
        WAITENC                     ; Esperar a posiciones válidas de los ejes de orien-
                                      tación.
        TOROTZ                      ; Girar el eje Z del WKS en la dirección del eje de
                                      la herramienta.
    ENDIF
    M17
ENDIF
...
```

Después la herramienta se puede retirar en el modo de operación JOG con un movimiento de retroceso en la dirección del eje de la herramienta.

3.15.7 Conmutación programable entre secuencias de parámetros (SCPARA)

El comando SCPARA permite solicitar la conmutación a una determinada secuencia de parámetros para un eje.

Nota

Sin conmutación de secuencia de parámetros durante mecanizado de la rosca

En el tallado de roscas G33 y roscado G331/G332, el control selecciona una secuencia de parámetros no modificable.

Conmutación de secuencia de parámetros bloqueada

La conmutación de secuencia de parámetros puede solicitarse también a través de la interfaz CN/PLC. Para evitar conflictos de conmutación, es posible bloquear la conmutación de secuencia de parámetros de CN (SCPARA) a través de la interfaz CN/PLC:

DB31, ... DBX9.3 (predefinición de secuencia de parámetros por CN bloqueada)

Nota

Si se solicita una conmutación de secuencia de parámetros mediante SCPARA mientras está bloqueada la conmutación a través de la interfaz CN/PLC, la conmutación se rechaza sin aviso de error.

Sintaxis

```
SCPARA[<Eje>]=<Valor>
```

Descripción

SCPARA:	Comando: conmutar secuencia de parámetros	
<Eje>:	Identificador de eje (eje de canal)	
	Tipo:	AXIS
<Valor>:	Número de secuencia de parámetros: 1, 2, 3... (máx. número de secuencia de parámetros)	

Ejemplo

Código de programa	Comentarios
...	
N110 SCPARA[X]= 3	; Selección: eje X, 3.er Juego de parámetros
...	

Información adicional

Habilitación de la conmutación de secuencia de parámetros

Es preciso habilitar explícitamente la conmutación de secuencia de parámetros del eje:

MD35590 $MA_PARAMSET_CHANGE_ENABLE[<Eje>]

Leer número de secuencia de parámetros

El número de la secuencia de parámetros seleccionada (secuencia de parámetros de consigna) puede leerse a través de la variable de sistema $AA_SCPAR.

3.15.8 Activar/desactivar adaptación (CADAPTON, CADAPTOF)

Los procedimientos predefinidos CADAPTON() y CADAPTOF() permiten activar, actualizar y desactivar, desde el programa de pieza, adaptaciones predefinidas para modificar los parámetros de dinámica o regulación. Se utilizan también en el CYCLE782 (Página 1129).

Sintaxis

```
...
CADAPTON(<Result>,<Axis>,<InVar>[,<InValue>])
...
CADAPTOF(<Result>,<Axis>,<InVar>)
...
```

Descripción

`CADAPTON()`:	Activar la relación de adaptación		
`CADAPTOF()`:	Desactivar la relación de adaptación **Nota:** Las relaciones de adaptación de efecto permanente (MD16501 = 1) omiten `CADAPTOF()`, mientras que los eventuales valores de entrada previamente programados (`<InVal>`) mantienen su efectividad.		
`<Result>`:	Variable de resultado: Valor de retorno para el estado (parámetro call by reference)		
	Tipo de dato:	INT	
	Valor:	0	Sin errores
		1	No hay ninguna tabla de adaptación válida parametrizada
		2	Parámetro `<Axis>` no válido
		3	Parámetro `<InVar>` no válido
		4	Reservado
		5	Parámetro `<InVal>` no válido
`<Axis>`:	Nombre de eje de máquina del eje de entrada de la relación de adaptación		
	Tipo de dato:	AXIS	
	Rango de valores:	Nombres de eje de máquina definidos en el canal	
	Nota: Con este parámetro se direccionan las adaptaciones que han introducido en el MD16504 $MN_CADAPT_INPUT_AX un valor correspondiente al parámetro `<Axis>`. Los restantes parámetros `<InVar>` e `<InVal>` están asignados a dicho eje.		
`<InVar>`:	Magnitud de entrada de la relación de adaptación		
	Tipo de dato:	INT	
	Valor:	1	Inercia del eje
		2	Posición del eje
		3	Velocidad del eje

<InVal>:	Valor de entrada de la relación de adaptación Parámetro opcional para la adaptación inteligente (<InVar>=1).		
	Tipo de dato:	REAL	
	Valor:	> 0	Inercia actual
		= 0	La carga es indefinida o no se conoce. En este caso, las adaptaciones asignadas devuelven como magnitud de salida el valor sustitutivo 1.0.
		< 0	La llamada finaliza con <Status> = 5 (parámetro <InVal> no válido). El valor de entrada activado en último lugar permanece activo.
	Nota: Si no hay un valor de entrada programado, tendrá efectividad el último valor programado o, tras un arranque del control, el valor estándar (= 0).		

Ejemplo

Código de programa	**Comentarios**
DEF INT RESULT	; Definición de las variables de resultado
...	
CADAPTON(RESULT,MX1,2)	; Activar la adaptación
IF RESULT <> 0	; Evaluación de las variables de resultado (necesario tras cada instrucción CADAPTON/CADAPTOF)
MSG("CADAPT-RESULT=" << RESULT)	
STOPRE	
SETAL(61000)	
ENDIF	
...	
CADAPTOF(RESULT,MX1,2)	; Desactivar la adaptación
...	; Evaluación de las variables de resultado
...	

Información adicional

Ejes lincados o Link

La función no puede aplicarse a los ejes lincados (función NCU Link). En ese caso, la variable de resultado devolvería el valor de retorno 2 ("Parámetro <Axis> no válido").

Búsqueda de secuencia

- **Búsqueda de secuencia sin cálculo**
  Las instrucciones CADAPTON/CADAPTOF programadas entre el inicio del programa y la secuencia meta se omiten. El usuario debe escoger la meta de búsqueda de modo que se seleccionen las adaptaciones relevantes para la sección de mecanizado.
- **Búsqueda de secuencia con cálculo en el contorno o en el punto final de secuencia**
  Los comandos de activación o desactivación y los eventuales valores de entrada programados se recopilan y, al final de la búsqueda de secuencia, se emite el estado actual con las secuencias de acciones de búsqueda de secuencia.
- **Búsqueda de secuencia con cálculo en modo de prueba del programa (SERUPRO)**
  Al desactivar SERUPRO, se transfiere el estado de las adaptaciones alcanzado por medio de las instrucciones CADAPTON/CADAPTOF al servicio real.

3.15.9 Adaptación del filtro antitirones FIR al modo dinámico (CALCFIR)

Después de cambiar el modo dinámico para todos los ejes del canal para los que hay activo un filtro antitirones de paso bajo FIR, para alcanzar un efecto de atenuación y una precisión de contorno idénticos, los ajustes de filtro FIR dependientes de la dinámica del eje lineal limitante del conjunto de interpolación pueden transferirse con el procedimiento predefinido CALCFIR a todos los demás ejes de canal con filtro antitirones de paso bajo FIR activo, y hacerse efectivos.

Aplicación

Se utiliza CALCFIR en el ciclo de fabricante CUST_832.

Proceso estándar

Al principio de un programa CN para el mecanizado de superficies de forma libre se programa CYCLE832. Durante la ejecución del programa, CYCLE832 llama el ciclo de fabricante CUST_832 y se ejecuta el procedimiento predefinido CALCFIR insertado por el fabricante de la máquina en el CUST_832.

Efecto

Los ajustes de filtro sobrescritos por CALCFIR permanecen activos hasta que se vuelven a sobrescribir con la próxima llamada de CALCFIR. Por lo tanto, esto siempre es necesario cuando hay programada una modificación del modo dinámico actual por un comando del grupo G 59 (modo dinámico para interpolación de la trayectoria).

Comportamiento en caso de arranque del control / reset de canal / reset de fin de programa

En el arranque del control y en caso de reset de canal / de fin de programa se activan automáticamente ajustes de filtro FIR adecuados a la posición de borrado del grupo G 59.

Requisitos

- La opción "Top Speed" está activada.

  Nota

  Para obtener resultados óptimos con "Top Speed", se recomienda además utilizar la opción "Top Surface".

- Para los ejes del conjunto de interpolación deben cumplirse las siguientes condiciones:
  - El filtro antitirones de paso bajo FIR está activo.
  - Los ajustes de filtro FIR dependientes de la dinámica están configurados.
  - Está habilitada la sobrescritura de los ajustes de filtro por parte de CALCFIR.
- Está activa la limitación de tirones SOFT/SOFTA.

Sintaxis

CALCFIR se programa en una secuencia propia. La llamada en el ciclo de fabricante CUST_832 se realiza directamente después de la programación del modo dinámico:

```
...
DYN...
CALCFIR
...
```

Descripción

DYN...:	Comando del grupo G 59 para seleccionar el modo dinámico	
	DYNNORM:	Activar **dinámica normal**
	DYNPOS:	Activar dinámica para **modo de posicionado, roscado de taladros**
	DYNROUGH:	Activar dinámica para **desbaste**
	DYNSEMIFIN:	Activar dinámica para **preacabado**
	DYNFINISH:	Activar dinámica para **acabado**
	DYNPREC:	Activar dinámica para **acabado fino**
CALCFIR:	Procedimiento predefinido para la adaptación dinámica de filtros antitirones de paso bajo FIR	

Nota

CALCFIR ejecuta una NEWCONFIG implícita para hacer efectivo el resultado del cálculo del filtro FIR, tanto en los datos de máquina específicos de eje para parametrizar el filtro antitirones de paso bajo FIR, como también en CPRECON (Página 874). La NEWCONFIG implícita hace que puedan activarse otros datos de máquina que son efectivos con la NEWCONFIG.

Condiciones

Uso en acciones síncronas

CALCFIR **no** se puede programar en acciones síncronas.

3.16 Acoplamientos de ejes

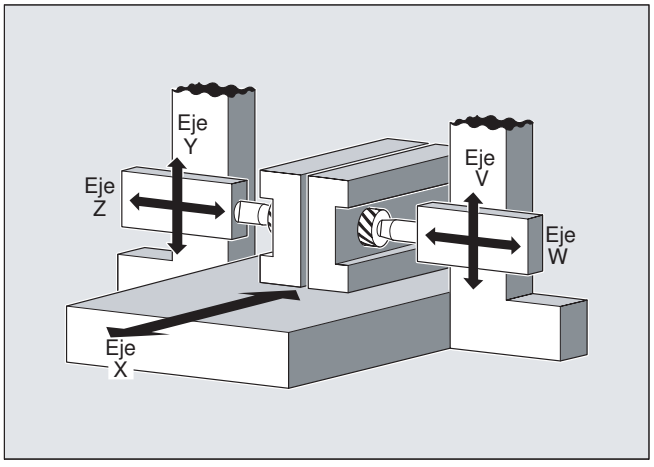
3.16.1 Arrastre de ejes (TRAILON, TRAILOF)

Cuando se desplaza un eje definido como maestro, los ejes arrastrados (= ejes esclavos) asignados al conjunto de ejes arrastrados se desplazan siguiendo los recorridos descritos por el eje maestro, teniendo en cuenta el factor de acoplamiento.

El eje maestro y los esclavos o arrastrados componen un conjunto de ejes maestro-esclavos.

Rango de aplicación

- Desplazamiento de un eje mediante un eje ficticio. El eje maestro es un eje ficticio, el eje arrastrado es un eje real. De esta manera se puede desplazar el eje real teniendo en cuenta el factor de acoplamiento.
- Mecanizado de dos caras con 2 conjuntos de ejes maestro-esclavos:
  1. Eje maestro Y, eje arrastrado V
  2. Eje maestro Z, eje arrastrado W



Sintaxis

```
TRAILON(<Eje esclavo>,<Eje maestro>,<Factor de acoplamiento>)
TRAILOF(<Eje esclavo>,<Eje maestro>,<Eje maestro 2>)
TRAILOF(<Eje esclavo>)
```

Descripción

`TRAILON`:	Comando de activación y definición de un conjunto de ejes	
	Eficacia:	modal
`<Eje esclavo>`:	Parámetro 1: Nombre del eje arrastrado (esclavo) **Nota:** Un eje arrastrado puede actuar también como eje maestro para otros ejes arrastrados. De esta forma se pueden definir diferentes configuraciones para conjuntos de ejes maestro-esclavos.	
`<Eje maestro>`:	Parámetro 2: Nombre del eje maestro.	

<Factor de acoplamiento>:	Parámetro 3: Factor de acoplamiento El factor de acoplamiento define la relación entre los trayectos del eje arrastrado y el eje maestro: <Factor de acoplamiento> = trayecto del eje arrastrado/trayecto del eje maestro	
	Tipo:	REAL
	Ajuste predeterminado:	1
	La introducción de valores negativos provoca que los desplazamientos entre el eje maestro y arrastrado sean en sentidos opuestos. Si al programar dicha función se omite el factor de acoplamiento, éste toma por defecto el valor 1.	
TRAILOF:	Comando de desactivación de un conjunto de ejes	
	Eficacia:	Modal
	La función TRAILOF con 2 parámetros desactiva sólo el acoplamiento al eje maestro indicado: TRAILOF(<Eje esclavo>,<Eje maestro>) Si un eje arrastrado posee 2 ejes maestros, para desactivar los dos acoplamientos puede ejecutarse la función TRAILOF con 3 parámetros: TRAILOF(<Eje esclavo>,<Eje maestro>,<Eje maestro 2>) El mismo resultado se obtiene con la programación de TRAILOF sin indicar un eje maestro: TRAILOF(<Eje esclavo>)	

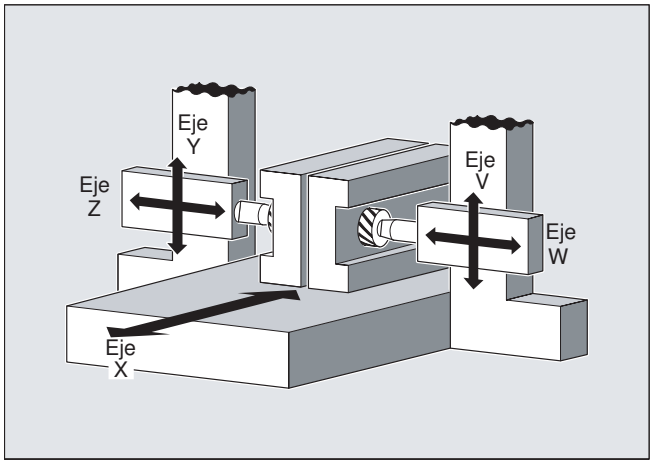
Nota

La función “eje arrastrado” siempre se activa en el sistema de coordenadas básico.

La cantidad máxima de conjuntos de ejes maestro-esclavos activables simultáneamente está limitada únicamente por las posibilidades de combinación de los ejes de la máquina.

Ejemplo

La pieza se debe de mecanizar por las dos caras simultáneamente tal como indica la configuración de la figura. Para esto se definen 2 conjuntos de ejes maestro-esclavos.



Código de programa	**Comentarios**
…	
N100 TRAILON(V,Y)	; Activación del 1.er conjunto de ejes maestro-esclavos
N110 TRAILON(W,Z,-1)	; Activación del 2.° conjunto de ejes maestro-esclavos Factor de acoplamiento negativo: el eje arrastrado se desplaza en la dirección contraria al eje maestro.
N120 G0 Z10	; Penetración de los ejes Z y W en sentidos opuestos.
N130 G0 Y20	; Penetración de los ejes Y y V en el mismo sentido.
…	
N200 G1 Y22 V25 F200	; Superposición de un desplazamiento del eje arrastrado V dependiente e independiente.
…	
TRAILOF(V,Y)	; Desactivación del 1.er conjunto de ejes maestro-esclavos.
TRAILOF(W,Z)	; Desactivación del 2.° conjunto de ejes maestro-esclavos.

Información adicional

Tipos de eje

Un conjunto de ejes maestro-esclavos puede estar formado por cualquier combinación de ejes lineales y giratorios. Como eje maestro también se puede definir un eje ficticio.

Ejes arrastrados

Un eje arrastrado puede estar asignado como máximo a dos ejes maestros. La asignación se puede realizar en diferentes conjuntos de ejes maestro-esclavos.

Un eje arrastrado puede programarse con todos los comandos de desplazamiento disponibles (G0, G1, G2, G3...). Adicionalmente a las trayectorias definidas de forma independiente, el eje arrastrado también puede efectuar desplazamientos que se derivan de los desplazamientos de su eje maestro por medio de los factores de acoplamiento.

limitación de la dinámica

La limitación de la dinámica depende del tipo de activación del conjunto de ejes:

- Activación en el programa de pieza
  Si la activación se efectúa en el programa de pieza y todos los ejes maestros actúan como ejes de programa en el canal que se va a activar, durante el desplazamiento de los ejes maestros se tiene en cuenta la dinámica de todos los ejes arrastrados para no sobrecargar ninguno de estos ejes.
  Si la activación se lleva a cabo en el programa de pieza con ejes maestros que no están activos como ejes de programa en el canal que se va a activar ($AA_TYP ≠ 1), durante el desplazamiento de los ejes maestros no se tiene en cuenta la dinámica del eje arrastrado. Esto puede provocar una sobrecarga en los ejes arrastrados con una dinámica menor que la necesaria para el acoplamiento.
- Activación en acción síncrona
  Si la activación tiene lugar en una acción síncrona, durante el desplazamiento de los ejes maestros no se tiene en cuenta la dinámica de los ejes arrastrados. Esto puede provocar una sobrecarga en los ejes arrastrados con una dinámica menor que la necesaria para el acoplamiento.

⚠ PRECAUCIÓN

Sobrecarga del eje

Si se activa un conjunto de ejes maestro-esclavos

- en acciones síncronas
- dentro del programa de pieza con ejes maestros que no actúan como ejes de programa en el canal del eje arrastrado,

es responsabilidad del usuario/fabricante de la máquina adoptar las medidas oportunas para que el desplazamiento del eje maestro no provoque una sobrecarga de los ejes arrastrados.

Estado de acoplamiento

El estado de acoplamiento de un eje puede consultarse en el programa de pieza mediante estas variables de sistema:

$AA_COUP_ACT[<Eje>]

Valor	Descripción
0	ningún acoplamiento activo
8	Arrastre de ejes activo

Visualización del trayecto residual del eje arrastrado con ejes giratorios con módulo

Si el eje maestro y el esclavo son ejes giratorios con módulo, los desplazamientos del eje maestro de n * 360°, donde n = 1, 2, 3... , se suman en la visualización del trayecto residual del eje arrastrado hasta la desactivación del acoplamiento.

Ejemplo: Sección de programa con TRAILON, eje maestro B y eje esclavo C

Código de programa	Comentarios
`TRAILON(C,B,1)`	`; Activación de un acoplamiento`
`G0 B0`	`; Posición inicial`

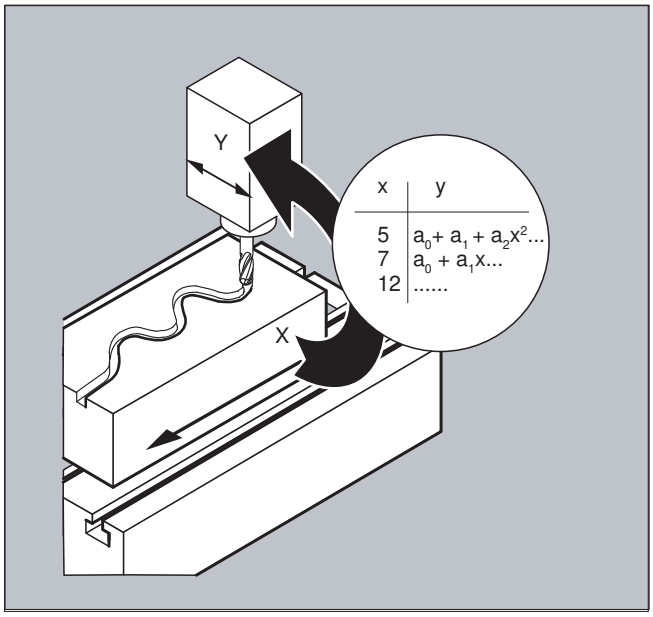
Código de programa	Comentarios
	; Visualización del trayecto residual al inicio de la secuencia:
G91 B360	; B=360, C=360
G91 B720	; B=720, C=1080
G91 B360	; B=360, C=1440

3.16.2 Tablas de levas (CTAB)

Las tablas de levas sirven para programar relaciones de posición y velocidad entre dos ejes (eje maestro y esclavo). Las tablas de levas se definen en el programa de pieza.

Aplicación

Las tablas de levas sustituyen a los perfiles de levas mecánicos. La tabla de levas constituye la base para el acoplamiento entre ejes de valores maestros, creando la relación funcional entre los valores maestros y de seguimiento: Con la correspondiente programación, el control numérico calcula, a partir de las posiciones de eje maestro y arrastrado asignadas entre sí, un polinomio que permite simular el disco de leva.

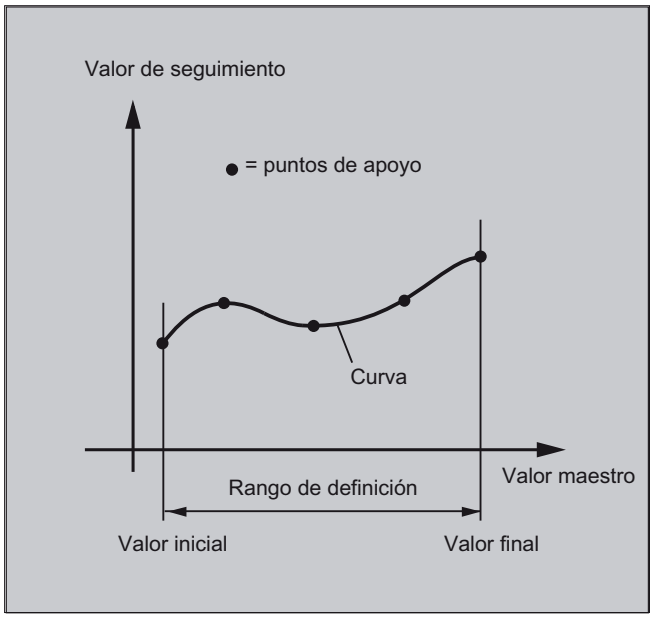


3.16.2.1 Definición de tablas de levas (CTABDEF, CATBEND)

Una tabla de levas es un programa de pieza o una sección del programa de pieza que comienza con el comando CTABDEF y termina con CTABEND.

Por medio de comandos de desplazamiento, en esta sección del programa de pieza se asignan a las diferentes posiciones del eje maestro posiciones unívocas del eje esclavo, las

cuales sirven como puntos de apoyo para calcular una curva en forma de polinomio de 5.º grado como máximo.



Requisitos

Para definir tablas de levas debe reservarse espacio en la memoria configurando el DM correspondiente (→ fabricante de la máquina).

Sintaxis

```
CTABDEF(<Eje esclavo>,<Eje
maestro>,<n>,<Periodicidad>[,<Ubicación>])
...
CTABEND
```

Descripción

`CTABDEF( )`:	Inicio de la definición de la tabla de levas
`CTABEND`:	Fin de la definición de la tabla de levas
`<Eje esclavo>`:	Eje cuyo desplazamiento debe calcularse mediante la tabla de levas
`<Eje maestro>`:	Eje que proporciona los valores maestros para calcular el desplazamiento del eje esclavo
`<n>`:	Número (**ID**) de la tabla de levas El número de una tabla de levas es unívoco e independiente de la ubicación. En la memoria CN estática y dinámica no puede haber tablas con el mismo número.

<Periodicidad>:	Periodicidad de la tabla	
	0	La tabla no es periódica (se ejecuta una sola vez, también en el caso de los ejes giratorios)
	1	La tabla es periódica en relación con el eje maestro
	2	La tabla es periódica en relación con el eje maestro y el eje esclavo
<Ubicación>:	Indicación de la ubicación (opcional)	
	"SRAM"	La tabla de levas se crea en la memoria CN **estática**.
	"DRAM"	La tabla de levas se crea en la memoria CN **dinámica**.
	Nota: Si no se programa ningún valor para este parámetro, se utiliza la ubicación estándar ajustada con MD20905 $MC_CTAB_DEFAULT_MEMORY_TYPE.	

Nota

Sobrescribir

Las tablas de levas se sobrescriben en cuanto se utiliza su número (<n>) para definir una nueva tabla (excepción: cuando la tabla de levas está activa en un acoplamiento de ejes o bloqueada con CTABLOCK). **¡Al sobrescribir no se emite ningún aviso de advertencia!**

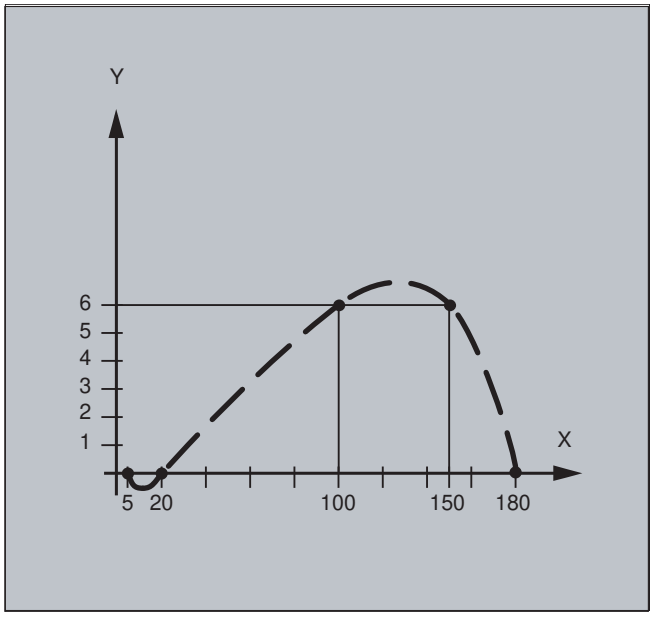
Ejemplos

Ejemplo 1: sección del programa como definición de la tabla de levas

Se desea utilizar una sección de programa, sin modificación alguna, para definir una nueva tabla de levas. El comando de parada de decodificación previa STOPRE contenido en dicha sección puede conservarse y se activa nuevamente en cuanto la sección del programa deja de utilizarse para la definición de la tabla y se han borrado CTABDEF y CTABEND.

Código de programa	**Comentarios**
…	
CTABDEF(Y,X,1,1)	; Definición de una tabla de levas.
…	
IF NOT ($P_CTABDEF)	
STOPRE	
ENDIF	
…	
CTABEND	

Ejemplo 2: definición de una tabla de levas no periódica



Código de programa	Comentarios
N100 CTABDEF(Y,X,3,0)	; Inicio de la definición de una tabla de levas no periódica con el número 3.
N110 X0 Y0	; El 1.er comando de desplazamiento establece valores iniciales y el 1.er punto de apoyo:
	Valor maestro: 0, valor esclavo: 0
N120 X20 Y0	; 2.° punto de apoyo:
	Valor maestro: 0…20, valor esclavo: valor inicial …0
N130 X100 Y6	; 3.er punto de apoyo:
	Valor maestro: 20…100, valor esclavo: 0…6
N140 X150 Y6	; 4.° punto de apoyo:
	Valor maestro: 100…150, valor esclavo: 6…6
N150 X180 Y0	; 5.° punto de apoyo:
	Valor maestro: 150…180, valor esclavo: 6…0
N200 CTABEND	; Fin de la definición. La representación interna de la tabla de levas adopta la forma de un polinomio de 5.° grado como máximo. El cálculo de la curva con los puntos de apoyo indicados depende del tipo de interpolación seleccionada modalmente (interpolación circular, lineal, spline). El estado del programa de pieza se restablece antes del inicio de la definición.

Ejemplo 3: definición de una tabla de levas periódica

Definición de una tabla de levas periódica con el número 2, rango de valores maestros de 0 a 360, desplazamiento de eje arrastrado desde 0 hasta 45 y de regreso a 0:

Código de programa	Comentarios
N10 DEF REAL DEPPOS	

```
Código de programa                                    Comentarios
N20 DEF REAL GRADIENT
N30 CTABDEF(Y,X,2,1)                                  ; Inicio de la definición.
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 definición.
;Test de la curva mediante acoplamiento de Y en 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)                     ; Lectura de la función de tabla
                                                      para el valor maestro 75.0.
N180 G0 X75 Y=DEPPOS                                  ; Posicionamiento de eje maestro y
                                                      esclavo.
;Tras la activación del acoplamiento no se requiere ninguna sincronización del eje
esclavo.
N190 LEADON(Y,X,2)
N200 G1 X110 F1000
N210 LEADOF(Y,X)
N220 M30
```

Información adicional

Valor inicial y final de la tabla de levas

Como valor inicial para el comienzo de la zona de definición de la tabla de levas se toma la primera indicación correspondiente a posiciones de ejes coherentes (el primer comando de desplazamiento) dentro de la definición de la tabla. El valor final de la zona de definición de dicha tabla queda determinado correspondientemente por el último comando de desplazamiento.

Repertorio de lenguaje disponible

En la definición de la tabla de levas se pueden utilizar todas las instrucciones del lenguaje de programación CN.

Nota

No se admiten las siguientes indicaciones en las definiciones de tablas de levas:

- Parada de decodificación previa
- Saltos en el desplazamiento del eje maestro
  (p. ej., en el cambio de transformadas)
- Comando de desplazamiento sólo para el eje arrastrado
- Inversión de desplazamiento del eje maestro; es decir, la posición del eje maestro debe ser siempre unívoca
- Instrucción CTABDEF y CTABEND en diferentes planos del programa

Efecto de las instrucciones modales

Todas las instrucciones modalmente activas programadas dentro de la definición de la tabla de levas pierden su vigencia al concluir la definición de la tabla. Por consiguiente, el estado del programa de pieza en el cual se define la tabla de levas es el mismo antes y después de la definición de la tabla.

Asignaciones a parámetros R

Las asignaciones a parámetros R dentro de la definición de la tabla se resetean tras CTABEND.

Ejemplo:

Código de programa	**Comentarios**
...	
R10=5 R11=20	; R10=5
...	
CTABDEF	
G1 X=10 Y=20 F1000	
R10=R11+5	; R10=25
X=R10	
CTABEND	
...	; R10=5

Activación de ASPLINE, BSPLINE, CSPLINE

Si dentro de una definición de tabla de levas CTABDEF ... CTABEND se activa un ASPLINE, BSPLINE o CSPLINE, se debería programar al menos un punto inicial antes de esta activación de spline. Se debería evitar una activación inmediata después de CTABDEF, dado que, de lo contrario, el spline depende de la posición actual del eje antes de la definición de la tabla de levas.

Ejemplo:

Código de programa

```
...
CTABDEF(Y,X,1,0)
X0 Y0
ASPLINE
X=5 Y=10
X10 Y40
...
CTABEND
```

Aplicación múltiple de tablas de levas

La relación funcional entre los ejes maestro y esclavo calculada mediante la tabla de levas se conserva bajo el número de tabla seleccionado tras finalizar el programa de pieza y tras POWER OFF, siempre que la tabla esté guardada en la memoria CN estática (SRAM).

Las tablas creadas en la memoria dinámica (DRAM) se borran con POWER ON y posiblemente se tengan que volver a crear.

Las tablas de levas ya creadas se pueden utilizar para cualquier tipo de combinación de ejes maestro y esclavo, sin importar qué ejes hayan sido utilizados para crear dichas tablas.

Sobrescribir tablas de levas

Las tablas de levas se sobrescriben tan pronto como se utiliza su número en una nueva definición de tabla.

Excepción: cuando una tabla de levas está activa en un acoplamiento de ejes o bloqueada con CTABLOCK.

Nota

¡Al sobrescribir las tablas de levas no se emite ningún aviso de advertencia!

¿Definición de tabla de levas activa?

La variable de sistema `$P_CTABDEF` permite consultar en cualquier momento desde el programa de pieza si está activa una definición de tabla de levas.

Cancelación de la definición de la tabla de levas

La sección de programa de pieza se puede utilizar nuevamente como programa de pieza real si se excluyen las instrucciones de definición de la tabla de levas.

Carga de tablas de levas mediante "Ejecución de externo"

En la ejecución externa de tablas de levas, el tamaño del búfer de recarga (DRAM) debe seleccionarse a través de DM18360 $MN_MM_EXT_PROG_BUFFER_SIZE de modo que toda la definición de la tabla de levas pueda almacenarse simultáneamente en el búfer de recarga. De lo contrario, la ejecución del programa de pieza se cancela con una alarma.

Saltos del eje esclavo

En función del ajuste en el dato de máquina
MD20900 $MC_CTAB_ENABLE_NO_LEADMOTION,
pueden tolerarse saltos del eje esclavo en caso de falta de movimiento del eje maestro.

3.16.2.2 Comprobación de la existencia de una tabla de levas (CTABEXISTS)

El comando CTABEXISTS permite comprobar si la memoria CN contiene un número concreto de tabla de levas.

Sintaxis

```
CTABEXISTS(<n>)
```

Descripción

CTABEXISTS:	Comprueba si la tabla de levas con el número <n> está presente en la memoria CN estática o dinámica	
	0	La tabla no existe
	1	La tabla existe
<n>:	Número (**ID**) de la tabla de levas	

3.16.2.3 Borrado de tablas de levas (CTABDEL)

Con CTABDEL se pueden borrar tablas de levas.

Nota

Las tablas de levas que están activas en un acoplamiento de ejes no se pueden borrar.

Sintaxis

```
CTABDEL(<n>)
CTABDEL(<n>,<m>)
CTABDEL(<n>,<m>,<Ubicación>)
CTABDEL()
CTABDEL(,,<Ubicación>)
```

Descripción

CTABDEL:	Comando para borrar tablas de levas
<n>:	Número (**ID**) de la tabla de levas que desea borrarse Al borrar un área de tablas de levas CTABDEL(<n>,<m>), <n> indica el número de la primera tabla de levas del área.

<m>:	Al borrar un área de tablas de levas CTABDEL(<n>,<m>), <m> indica el número de la última tabla de levas del área. ¡<m> debe ser mayor que <n>!	
<Ubicación>:	Indicación de la ubicación (opcional) Al efectuar el borrado **sin** especificar la ubicación, las tablas de levas indicadas se borran de la memoria CN estática y dinámica. Al efectuar el borrado **con** especificación de la ubicación, de las tablas de levas indicadas se borran sólo las que están presentes en la memoria indicada. Las demás se conservan.	
	"SRAM"	Borrado en la memoria CN **estática**
	"DRAM"	Borrado en la memoria CN **dinámica**

Si se programa CTABDEL sin indicar la tabla de levas que debe borrarse, se borran **todas** las tablas de levas o bien todas las que se encuentran en la memoria indicada:

CTABDEL():	Borra todas las tablas de levas de la memoria CN estática y dinámica
CTABDEL(,,"SRAM"):	Borra todas las tablas de levas de la memoria CN estática
CTABDEL(,,"DRAM"):	Borra todas las tablas de levas de la memoria CN dinámica

Nota

Si a la hora de efectuar un borrado múltiple CTABDEL(<n>,<m>) o CTABDEL() está activa en un acoplamiento al menos una de las tablas de levas que desean borrarse, el comando de borrado no se ejecuta, es decir, no se borra **ninguna** de las tablas de levas especificadas.

3.16.2.4 Bloqueo de las tablas de levas contra borrado y sobrescritura (CTABLOCK, CTABUNLOCK)

Las tablas de levas pueden protegerse contra el borrado y la sobrescritura accidentales activando bloqueos. Los bloqueos activados pueden anularse en cualquier momento.

Sintaxis

Activar un bloqueo:

```
CTABLOCK(<n>)
CTABLOCK(<n>,<m>)
CTABLOCK(<n>,<m>,<Ubicación>)
CTABLOCK()
CTABLOCK(,,<Ubicación>)
```

Anular un bloqueo:

```
CTABUNLOCK(<n>)
CTABUNLOCK(<n>,<m>)
CTABUNLOCK(<n>,<m>,<Ubicación>)
CTABUNLOCK()
CTABUNLOCK(,,<Ubicación>)
```

Descripción

CTABLOCK:	Comando para **activar** un bloqueo contra borrado/sobrescritura
CTABUNLOCK:	Comando para **anular** un bloqueo contra borrado/sobrescritura CTABUNLOCK vuelve a habilitar las tablas de levas bloqueadas con CTABLOCK. Las tablas de levas que actúan en un acoplamiento activo permanecen bloqueadas y no pueden borrarse. El bloqueo con CTABLOCK queda anulado en cuanto el bloqueo ejercido por el acoplamiento activo se anula con la desactivación de éste. De este modo se puede borrar esta tabla. No es necesario volver a ejecutar CTABUNLOCK.
<n>:	Número (**ID**) de la tabla de levas que desea bloquearse/desbloquearse Al bloquear/desbloquear una área de tablas de levas CTABLOCK(<n>,<m>) / CTABUNLOCK(<n>,<m>), <n> indica el número de la primera tabla de levas del área.
<m>:	Al bloquear/desbloquear una área de tablas de levas CTABLOCK(<n>,<m>) / CTABUNLOCK(<n>,<m>), <m> indica el número de la última tabla de levas del área. ¡<m> debe ser mayor que <n>!
<Ubicación>:	Indicación de la ubicación (opcional) Al activar/anular un bloqueo **sin** especificar la ubicación, las tablas de levas indicadas se bloquean/desbloquean en la memoria CN estática y dinámica. Al activar/anular un bloqueo **con** especificación de la ubicación, de las tablas de levas indicadas se bloquean/desbloquean sólo las que están presentes en la memoria indicada. Las demás no se bloquean/desbloquean.
	"SRAM" — Activar/anular bloqueo en la memoria CN **estática**
	"DRAM" — Activar/anular bloqueo en la memoria CN **dinámica**

Si se programa CTABLOCK/CTABUNLOCK sin indicar la tabla de levas que debe bloquearse/desbloquearse, se bloquean/desbloquean **todas** las tablas de levas o bien todas las que se encuentran en la memoria indicada:

CTABLOCK():	Bloquea todas las tablas de levas en la memoria CN estática y dinámica
CTABLOCK(,,"SRAM"):	Bloquea todas las tablas de levas en la memoria CN estática
CTABLOCK(,,"DRAM"):	Bloquea todas las tablas de levas en la memoria CN dinámica
CTABUNLOCK():	Desbloquea todas las tablas de levas en la memoria CN estática y dinámica
CTABUNLOCK(,,"SRAM"):	Desbloquea todas las tablas de levas en la memoria CN estática
CTABUNLOCK(,,"DRAM"):	Desbloquea todas las tablas de levas en la memoria CN dinámica

3.16.2.5 Tablas de levas: determinación de las propiedades de la tabla (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD)

Estos comandos permiten consultar propiedades importantes de una tabla de levas (número de tabla, estado de bloqueo, ubicación, periodicidad).

Sintaxis

```
CTABID(<p>)
```

```
CTABID(<p>,<Ubicación>)
CTABISLOCK(<n>)
CTABMEMTYP(<n>)
TABPERIOD(<n>)
```

Descripción

CTABID:	Proporciona el **número de tabla** consignado en la memoria indicada como tabla de levas número <p>. Ejemplo: CTABID(1,"SRAM") proporciona el número de la primera tabla de levas en la memoria CN estática. En este caso, la primera tabla de levas se corresponde con la tabla con el número más alto. **Nota:** Si entre dos llamadas consecutivas de CTABID se modifica el orden de las tablas de levas en la memoria, p. ej., borrando tablas de levas con CTABDEL, una función CTABID(<p>,...) con el mismo número <p> podría suministrar una tabla de levas diferente de la anterior.	
CTABISLOCK:	Devuelve el **estado de bloqueo** de la tabla de levas con el número <n>:	
	0	La tabla no está bloqueada
	1	La tabla está bloqueada con CTABLOCK
	2	La tabla está bloqueada por un acoplamiento activo
	3	La tabla está bloqueada con CTABLOCK y por un acoplamiento activo
	-1	La tabla no existe
CTABMEMTYP:	Devuelve la **ubicación** de la tabla de levas con el número <n>:	
	0	Tabla en la memoria CN estática
	1	Tabla en la memoria CN dinámica
	-1	La tabla no existe
CTABPERIOD:	Devuelve la **periodicidad** de la tabla de levas con el número <n>:	
	0	La tabla no es periódica
	1	La tabla es periódica en relación con el eje maestro
	2	La tabla es periódica en relación con los ejes maestro y esclavo
	-1	La tabla no existe
<p>:	Número de entrada en la memoria	
<n>:	Número (**ID**) de la tabla de levas	
<Ubicación>:	Indicación de la ubicación (opcional)	
	"SRAM"	Memoria CN **estática**
	"DRAM"	Memoria CN **dinámica**
	Nota: Si no se programa ningún valor para este parámetro, se utiliza la ubicación estándar ajustada con MD20905 $MC_CTAB_DEFAULT_MEMORY_TYPE.	

3.16.2.6 Lectura de valores de las tablas de levas (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX)

En el programa de pieza pueden leerse los siguientes valores de las tablas de levas:

- Valores de eje esclavo y maestro al principio y al final de una tabla de levas
- Valores de eje esclavo al principio y al final de un segmento de curva
- Valor de eje esclavo en relación con un valor de eje maestro
- Valor de eje maestro en relación con un valor de eje esclavo
- Valor mínimo y máximo del eje esclavo
  - en toda la zona de definición de la tabla de levas
    o bien
  - en un intervalo definido de la tabla de levas

Sintaxis

```
CTABTSV(<n>,<Gradiente>[,<Eje esclavo>])
CTABTEV(<n>,<Gradiente>[,<Eje esclavo>])
CTABTSP(<n>,<Gradiente>[,<Eje maestro>])
CTABTEP(<n>,<Gradiente>[,<Eje maestro>])
CTABSSV(<Valor maestro>,<n>,<Gradiente>[,<Eje esclavo>])
CTABSEV(<Valor maestro>,<n>,<Gradiente>[,<Eje esclavo>])
CTAB(<Valor maestro>,<n>,<Gradiente>[,<Eje esclavo>,<Eje maestro>]
CTABINV(<Valor esclavo>,<Valor aproximado>,<n>,<Gradiente>[,<Eje
esclavo>,<Eje maestro>]
CTABTMIN(<n>[,<Eje esclavo>])
CTABTMAX(<n>[,<Eje esclavo>])
CTABTMIN(<n>,<a>,<b>[,<Eje esclavo>,<Eje maestro>])
CTABTMAX(<n>,<a>,<b>[,<Eje esclavo>,<Eje maestro>])
```

Descripción

CTABTSV:	Leer el valor de eje esclavo al **principio** de la tabla de levas n.º <n>
CTABTEV:	Leer el valor de eje esclavo al **final** de la tabla de levas n.º <n>
CTABTSP:	Leer el valor de eje maestro al **principio** de la tabla de levas n.º <n>
CTABTEP:	Leer el valor de eje maestro al **final** de la tabla de levas n.º <n>
CTABSSV:	Leer el valor de eje esclavo al **principio** del segmento de curva correspondiente al valor de eje maestro indicado (<Valor maestro>)
CTABSEV:	Leer el valor de eje esclavo al **final** del segmento de curva correspondiente al valor de eje maestro indicado (<Valor maestro>)
CTAB:	Leer el valor de eje esclavo en relación con el valor de eje maestro indicado (<Valor maestro>)
CTABINV:	Leer el valor de eje maestro en relación con el valor de eje esclavo indicado (<Valor esclavo>)

CTABTMIN:	Calcular el **valor mínimo** del eje esclavo: • en toda la zona de definición de la tabla de levas o bien • en un intervalo definido <a> ... <b>
CTABTMAX:	Calcular el **valor máximo** del eje esclavo: • en toda la zona de definición de la tabla de levas o bien • en un intervalo definido <a> ... <b>
<n>:	Número (**ID**) de la tabla de levas
<Gradiente>:	El parámetro <Gradiente> devuelve el **paso** de la función de la tabla de levas respecto a la posición calculada
<Eje esclavo>:	Eje cuyo desplazamiento debe calcularse mediante la tabla de levas (opcional)
<Eje maestro>:	Eje que proporciona los valores maestros para calcular el desplazamiento del eje esclavo (opcional)
<Valor esclavo>:	Valor de eje esclavo para leer el valor de eje maestro correspondiente con CTABINV
<Valor maestro>:	Valor de eje maestro: • para leer el valor de eje esclavo correspondiente con CTAB o bien • para seleccionar el segmento de curva con CTABSSV/CTABSEV
<Valor aproximado>:	No es necesario que la asignación de un valor de eje maestro a un valor de eje esclavo con CTABINV sea siempre unívoca. Por tanto, CTABINV requiere como parámetro un valor aproximado para el valor de eje maestro esperado.
<a>:	Límite **inferior** del intervalo de valores maestros con CTABTMIN/CTABTMAX
<b>:	Límite **superior** del intervalo de valores maestros con CTABTMIN/CTABTMAX
	Nota: el intervalo de valores maestros <a> ... <b> debe estar dentro de la zona de definición de la tabla de levas.

Ejemplos

Ejemplo 1:

Cálculo de los valores de eje esclavo y maestro al principio y al final de la tabla de levas, así como del valor mínimo y máximo del eje esclavo en toda la zona de definición de la tabla de levas.

```
Código de programa                    Comentarios
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
```

Código de programa	Comentarios
...	
N100 CTABDEF(Y,X,1,0)	; Inicio de la definición de tabla
N110 X0 Y10	; Posición inicial del 1.er segmento de la tabla
N120 X30 Y40	; Posición final del 1.er segmento de la tabla = posición inicial del 2.° segmento de la tabla
N130 X60 Y5	; Posición final del 2.° segmento de la tabla = ...
N140 X70 Y30	
N150 X80 Y20	
N160 CTABEND	; Fin de la definición de la tabla.
...	
N200 STARTPOS=CTABTSV(1,GRADIENT)	; Valor de eje esclavo al principio de la tabla de levas = 10
N210 ENDPOS=CTABTEV(1,GRADIENT)	; Valor de eje esclavo al final de la tabla de levas = 20
N220 STARTPARA=CTABTSP(1,GRADIENT)	; Valor de eje maestro al principio de la tabla de levas = 0
N230 ENDPARA=CTABTEP(1,GRADIENT)	; Valor de eje maestro al final de la tabla de levas = 80
N240 MINVAL=CTABTMIN(1)	; Valor mínimo del eje esclavo con Y = 5
N250 MAXVAL=CTABTMAX(1)	; Valor máximo del eje esclavo con Y = 40

Ejemplo 2:

Cálculo de los valores de eje esclavo al principio y al final del segmento de curva correspondiente al valor de eje maestro X = 30.

Código de programa	Comentarios
N10 DEF REAL STARTPOS	
N20 DEF REAL ENDPOS	
N30 DEF REAL GRADIENT	
...	
N100 CTABDEF(Y,X,1,0)	; Inicio de la definición de tabla.
N110 X0 Y0	; Posición inicial del 1.er segmento de la tabla
N120 X20 Y10	; Posición final del 1.er segmento de la tabla = posición inicial del 2.° segmento de la tabla
N130 X40 Y40	; Posición final del 2.° segmento de la tabla = ...
N140 X60 Y10	
N150 X80 Y0	
N160 CTABEND	; Fin de la definición de la tabla.
...	
N200 STARTPOS=CTABSSV(30.0,1,GRADIENT)	; Posición inicial Y en el 2.° segmento = 10
N210 ENDPOS=CTABSEV(30.0,1,GRADIENT)	; Posición final Y en el 2.° segmento = 40

Información adicional

Uso en acciones síncronas

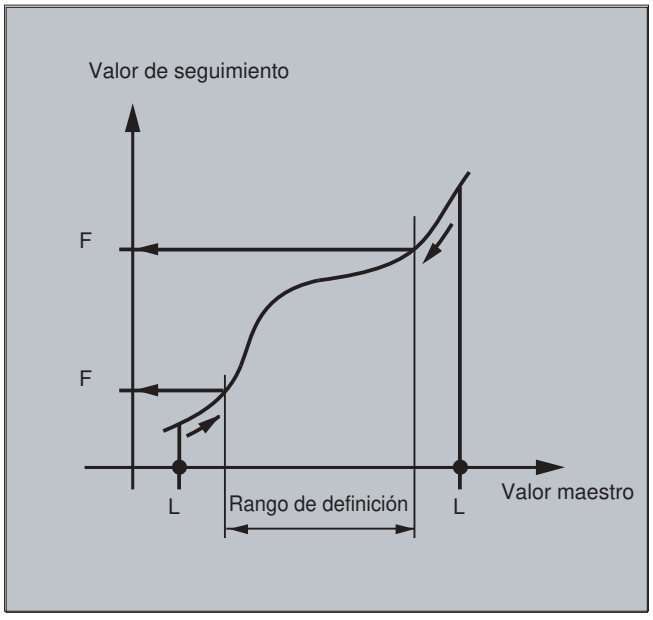
Todos los comandos para leer valores de las tablas de levas pueden utilizarse también en acciones síncronas (ver el apartado "Acciones síncronas a desplazamientos").

A la hora de emplear los comandos CTABINV, CTABTMIN y CTABTMAX es preciso asegurarse de que:

- el rendimiento de CN sea suficiente en el momento de ejecución
  o bien
- se consulte el número de segmentos de la tabla de levas antes de la llamada para poder subdividir la tabla en cuestión si es necesario

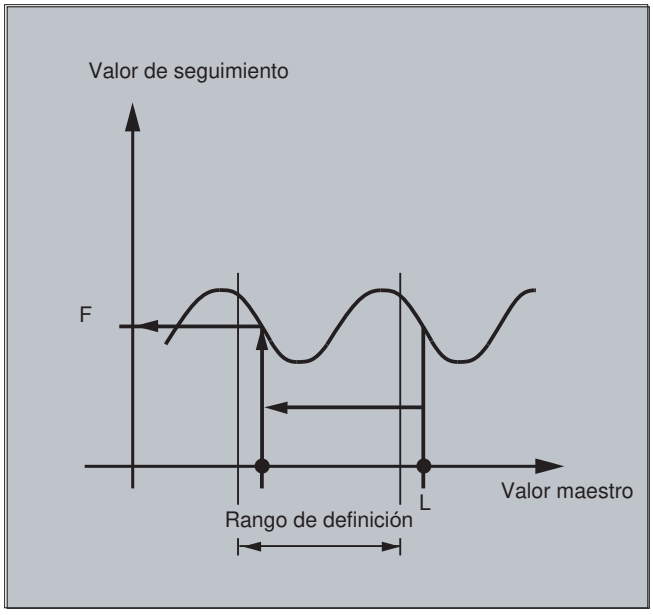
CTAB en tablas de levas no periódicas

Si el <Valor maestro> indicado está fuera de la zona de definición, se emite como valor esclavo el límite superior o inferior:



CTAB en tablas de levas periódicas

Si el <Valor maestro> indicado está fuera de la zona de definición, se evalúa el valor maestro Módulo de la zona de definición y se emite el valor esclavo correspondiente:



Valor aproximado para CTABINV

El comando CTABINV requiere un valor aproximado para el valor maestro esperado. CTABINV devuelve el valor maestro más cercano al valor aproximado. El valor aproximado puede ser, por ejemplo, el valor maestro del ciclo de interpolador anterior.

Paso de la función de la tabla de levas

La emisión del paso (<Gradiente>) permite calcular la velocidad del eje maestro o del eje esclavo en la posición correspondiente.

Indicación del eje maestro o del eje esclavo

La indicación opcional del eje maestro y/o del eje esclavo es importante en el caso de que dichos ejes hayan sido configurados con diferentes unidades de longitud.

CTABSSV, CTABSEV

En los siguientes casos, los comandos CTABSSV y CTABSEV **no** son aptos para consultar segmentos programados:

- Se han programado círculos o evolventes.
- El achaflanado o redondeo con CHF/RND está activo.
- El matado de esquinas con G643 está activo.
- La compresión de secuencias CN (COMP...) está activa.

3.16.2.7 Tablas de levas: comprobación del uso de los recursos (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL)

Estos comandos ofrecen al programador la posibilidad de obtener información actual sobre la ocupación de los recursos para tablas de levas, segmentos de tablas y polinomios.

Sintaxis

```
CTABNO
CTABNOMEM(<Ubicación>)
CTABFNO(<Ubicación>)
CTABSEGID(<n>,<Ubicación>)
CTABSEG(<Ubicación>,<Tipo de segmento>)
CTABFSEG(<Ubicación>,<Tipo de segmento>)
CTABMSEG(<Ubicación>,<Tipo de segmento>)
CTABPOLID(<n>)
CTABPOL(<Ubicación>)
CTABFPOL(<Ubicación>)
CTABMPOL(<Ubicación>)
```

Descripción

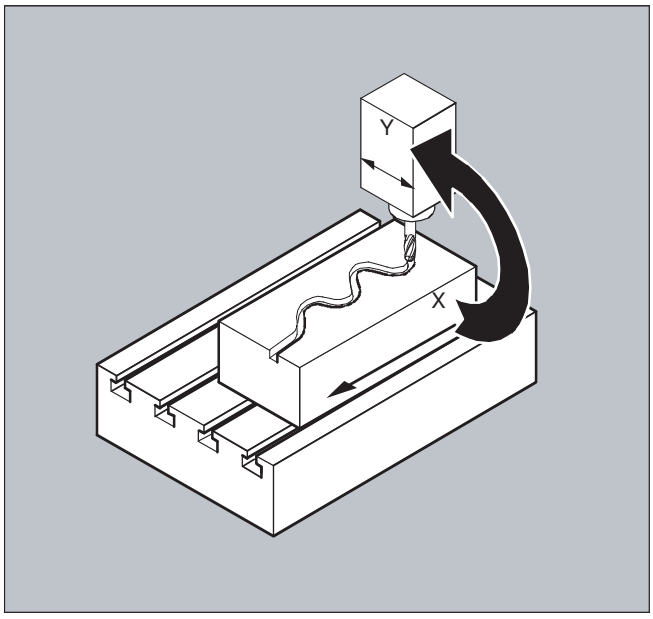
CTABNO:	Calcular el número total de tablas de levas **definidas** (en la memoria CN estática y dinámica)	
CTABNOMEM:	Calcular el número de tablas de levas **definidas** en la <Ubicación> indicada	
CTABFNO:	Calcular el número de tablas de levas **aún posibles** en la <Ubicación> indicada	
CTABSEGID:	Calcular el número de segmentos de curva del <Tipo de segmento> indicado utilizados por la tabla de levas con el número <n>	
CTABSEG:	Calcular el número de segmentos de curva **utilizados** del <Tipo de segmento> indicado en la <Ubicación> especificada	
CTABFSEG:	Calcular el número de segmentos de curva **aún posibles** del <Tipo de segmento> indicado en la <Ubicación> especificada	
CTABMSEG:	Calcular el número **máximo posible** de segmentos de curva del <Tipo de segmento> indicado en la <Ubicación> especificada	
CTABPOLID:	Calcular el número de polinomios de leva utilizados por la tabla de levas con el número <n>	
CTABPOL:	Calcular el número de polinomios de leva **utilizados** en la <Ubicación> indicada	
CTABFPOL:	Calcular el número de polinomios de leva **aún posibles** en la <Ubicación> indicada	
CTABMPOL:	Calcular el número **máximo posible** de polinomios de leva en la <Ubicación> indicada	
<n>:	Número (**ID**) de la tabla de levas	
<Ubicación>:	Indicación de la ubicación (opcional)	
	"SRAM"	Memoria CN **estática**
	"DRAM"	Memoria CN **dinámica**
	Nota: Si no se programa ningún valor para este parámetro, se utiliza la ubicación estándar ajustada con MD20905 $MC_CTAB_DEFAULT_MEMORY_TYPE.	
<Tipo de segmento>:	Indicación del tipo de segmento (opcional)	
	"L"	Segmentos lineales
	"P"	Segmentos de polinomio
	Nota: Si no se programa ningún valor para este parámetro, se emite la suma de los segmentos lineales y de polinomio.	

3.16.3 Acoplamiento entre ejes de valores maestros (LEADON, LEADOF)

Nota

¡Esta función no está disponible para SINUMERIK 828D!

En el acoplamiento entre ejes de valores maestros se desplazan sincrónicamente un eje maestro y un eje arrastrado. En este caso, la posición correspondiente del eje arrastrado está asignada unívocamente, a través de una tabla de levas o del polinomio calculado a partir de dicha tabla, a una posición eventualmente simulada del eje maestro.



Eje maestro es el eje que suministra los valores iniciales para la tabla de levas. **Eje arrastrado** es el eje que asume las posiciones calculadas a través de la tabla de levas.

Acoplamiento de valores reales y de valores de consigna

Como valores maestros, es decir, los valores iniciales para determinar el posicionamiento del eje arrastrado, se pueden utilizar:

- Valores reales de la posición del eje maestro: Acoplamiento de valores reales
- Valores de consigna de la posición del eje maestro: acoplamiento de valores de consigna

El acoplamiento de valores maestros está referido siempre al sistema de coordenadas básico.

Para la creación de tablas de levas, ver el capítulo "Tablas de levas".

Sintaxis

```
LEADON(<eje_esclavo>,<eje_maestro>,<n>)
LEADOF(<eje_esclavo>,<eje_maestro>)
```

o desconexión sin indicar el eje maestro:

```
LEADOF(<eje_esclavo)
```

El acoplamiento de valor maestro puede activarse o desactivarse tanto desde el programa de pieza como durante el movimiento desde acciones síncronas.

Descripción

`LEADON`:	Activar acoplamiento de valores maestros
`LEADOF`:	Desactivar acoplamiento de valores maestros
`<Eje esclavo>`:	Eje arrastrado o esclavo
`<Eje maestro>`:	Eje maestro
`<n>`:	Número de la tabla de levas
`$SA_LEAD_TYPE`:	Conmutación entre acoplamiento de valores de consigna y de valores reales

Desactivar el acoplamiento de valores maestros, LEADOF

Con la desactivación del acoplamiento de los valores maestros, el eje arrastrado vuelve a ser el eje de comando normal.

Acoplamiento entre ejes de valores maestros y diferentes estados operativos, RESET

Los acoplamientos de valores maestros se desactivan con RESET, dependiendo del ajuste en el dato de máquina.

Ejemplo: acoplamiento de valores maestros desde acción síncrona

En una línea de prensas se desea sustituir un acoplamiento mecánico convencional entre un eje maestro (eje de estampado) y los ejes de un sistema de transferencia, compuesto de ejes de transferencia y auxiliares, por un sistema de acoplamiento electrónico.

Demuestra la forma de sustituir un sistema de transferencia mecánico por uno electrónico en una línea de prensas. Las operaciones de acoplamiento y desacoplamiento han sido concebidas como **acciones síncronas estáticas**.

El eje maestro LW (eje de estampado) manda de forma definida los ejes de transferencia y auxiliares, los ejes arrastrados, a través de tablas de levas.

Ejes esclavos

Eje longitudinal o de avance (X)
Eje transversal o de cierre (YL)
Eje de elevación (ZL)
Eje auxiliar, avance de rodillos (U)
Eje auxiliar, cabeza de ajuste (V)
Eje auxiliar, engrase (W)

Acciones

En las acciones síncronas se presentan, p. ej., las siguientes acciones:

- Acoplar, `LEADON(<eje_esclavo>,<eje_maestro>,<num_tabla_levas>)`
- Desacoplar, `LEADOF(<eje_esclavo>,<eje_maestro>)`
- Poner valor real, `PRESETON(<eje>, <valor>)`

- Poner marcas, `$AC_MARKER[i]=<valor>`
- Modo de acoplamiento: valor maestro real/virtual
- Aprox. a posiciones de eje, `POS[<eje>]=<valor>`

Condiciones

Como condiciones se evalúan entradas digitales rápidas, variables de tipo real `$AC_META` y comparaciones de posición enlazadas con el operando lógico AND.

Nota

En el ejemplo siguiente se utilizaron cambios de línea, sangrado y **negrita**exclusivamente para mejorar la legibilidad de la programación. En el control numérico todo el contenido bajo un número de línea contiene una sola línea.

Comentarios

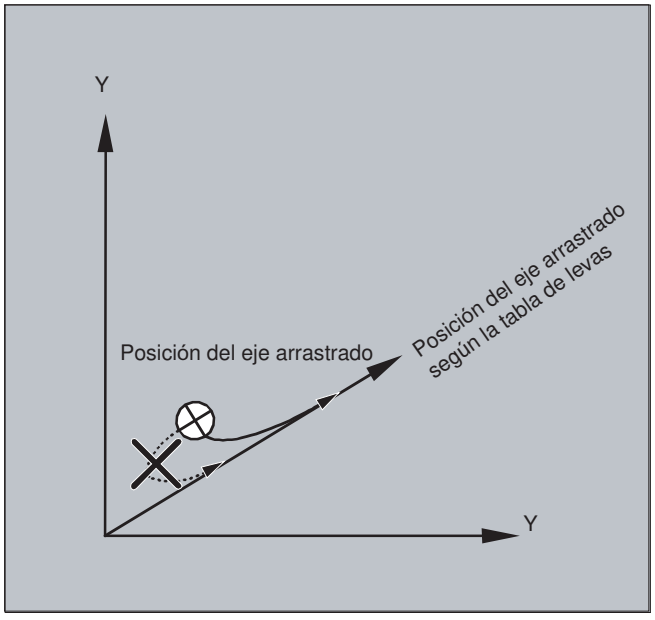
Código de programa	**Comentarios**
	; Define todas las acciones síncronas estáticas.
	; **** Resetear meta
N2 $AC_META[0]=0 $AC_META[1]=0 $AC_META[2]=0 $AC_META[3]=0 $AC_META[4]=0 $AC_META[5]=0 $AC_META[6]=0 $AC_META[7]=0	
	; **** E1 0=>1 Acoplamiento de transferencia activado
N10 IDS=1 EVERY ($A_IN[1]==1) AND ($A_IN[16]==1) AND ($AC_META[0]==0) DO LEADON(X,LW,1) LEADON(YL,LW,2) LEADON(ZL,LW,3) $AC_META[0]=1	
	; **** E1 0=>1 Acoplamiento Avance de rodillos activado
N20 IDS=11 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_META[5]==0) DO LEADON(U,LW,4) PRESETON(U,0) $AC_META[5]=1	
	; **** E1 0->1 Acoplamiento Cabeza de ajuste activado
N21 IDS=12 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_META[6]==0) DO LEADON(V,LW,4) PRESETON(V,0) $AC_META[6]=1	
	; **** E1 0->1 Acoplamiento Engrase activado
N22 IDS=13 EVERY ($A_IN[1]==1) AND ($A_IN[5]==0) AND ($AC_META[7]==0) DO LEADON(W,LW,4) PRESETON(W,0) $AC_META[7]=1	
	; **** E2 0=>1 Acoplamiento desactivado
N30 IDS=3 EVERY ($A_IN[2]==1) DO LEADOF(X,LW) LEADOF(YL,LW) LEADOF(ZL,LW) LEADOF(U,LW) LEADOF(V,LW) LEADOF(W,LW) $AC_META[0]=0 $AC_META[1]=0 $AC_META[3]=0 $AC_META[4]=0 $AC_META[5]=0 $AC_META[6]=0 $AC_META[7]=0	
....	
N110 G04 F01	
N120 M30	

Descripción

Para el acoplamiento de valores maestros se requiere la sincronización de los ejes maestro y arrastrado. Esta sincronización sólo puede lograrse si, al activar dicho acoplamiento, el eje arrastrado se encuentra dentro de la zona de tolerancia del desarrollo de la curva calculado en la tabla de levas.

La zona de tolerancia para la posición del eje arrastrado está definida en el dato de máquina `DM 37200: COUPLE_POS_POL_COARSE A_LEAD_TYPE .`

Si el eje arrastrado no se encuentra todavía en la posición correspondiente al activarse el acoplamiento de valores maestros, se establece la marcha síncrona de forma automática tan pronto el valor de posicionamiento calculado para el eje arrastrado se aproxima a la posición real de dicho eje. Durante la sincronización, el eje arrastrado se desplaza en el sentido definido por la velocidad de consigna de dicho eje (calculado en base a la velocidad del eje maestro y la tabla de levas CTAB).

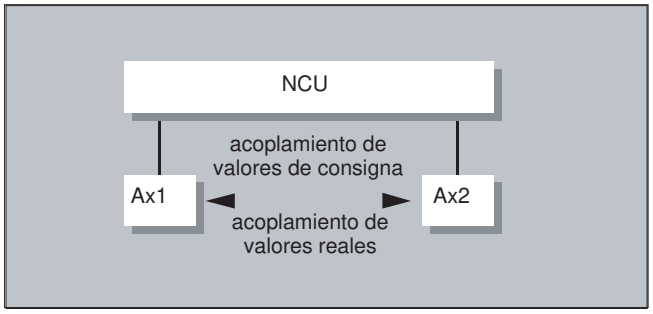


Sin marcha síncrona

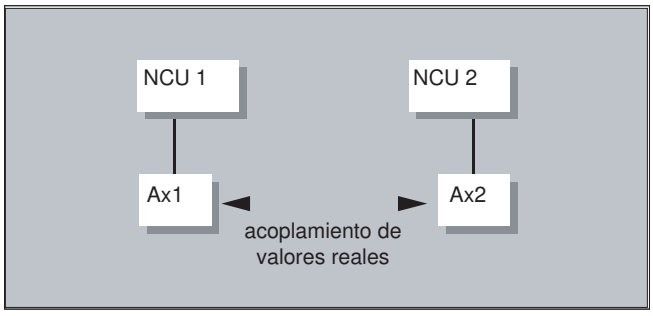
Si al activar el acoplamiento de valores maestros la posición de consigna del eje arrastrado difiere de la posición actual de dicho eje, no se establece la marcha síncrona.

Acoplamiento de valores reales y de valores de consigna

El acoplamiento de valores de consigna proporciona, en comparación con el de valores reales, un mejor sincronismo entre los ejes maestro y arrastrado, por lo que está ajustado de modo estándar.



El acoplamiento de valores de consigna sólo es posible cuando el eje maestro y el arrastrado son interpolados con la misma NCU. En caso de eje maestro externo, el eje arrastrado solo se puede acoplar al eje maestro a través de valores reales.



La **conmutación** es posible mediante el dato de operador $SA_LEAD_TYPE.

La conmutación entre acoplamiento de valores reales y de consigna se debe realizar siempre estando parado el eje arrastrado. En efecto, sólo se resincroniza en una parada o tras una conmutación.

Ejemplo de aplicación

Los valores reales no se pueden leer correctamente en caso de grandes vibraciones de la máquina. Por consiguiente, al utilizar el acoplamiento de valores maestros durante la transferencia de prensa, en operaciones de grandes vibraciones puede resultar necesario conmutar del acoplamiento los valores reales al de valores de consigna.

Simulación de valores maestros en el acoplamiento de valores de consigna

El interpolador para el eje maestro se puede separar del servo a través de los datos de máquina. Por consiguiente, en el acoplamiento de valores de consigna se pueden generar valores de consigna sin que el eje maestro se haya desplazado realmente.

Los valores maestros generados con el acoplamiento de valores de consigna se leen, para su aplicación (p. ej., en acciones síncronas) de las variables siguientes:

- $AA_LEAD_P	Valor maestro posición
- $AA_LEAD_V	Valor maestro velocidad

Crear valores maestros

Los valores maestros se pueden generar opcionalmente con otros procedimientos programados. Los valores maestros generados de esta manera se escriben en la variable

- $AA_LEAD_SP	Valor maestro posición
- $AA_LEAD_SV	Valor maestro velocidad

y se leen desde allí. Para utilizar estas variables es necesario ajustar el dato de operador $SA_LEAD_TYPE = 2.

Estado del acoplamiento

Desde el programa de pieza CN se puede consultar el estado del acoplamiento mediante las variables del sistema siguientes:

`$AA_COUP_ACT[eje]`
0: ningún acoplamiento activo
16: acoplamiento de valores maestro activos

Gestión de estados en acciones síncronas

Los procesos de conmutación y de acoplamiento se gestionan a través de variables de tiempo real:

`$AC_META[i] = n`
gestiona con:
i Número de meta (lábel)
n Valor de estado

3.16.4 Reductor electrónico (EG)

Con la función “Reductor electrónico” es posible controlar el movimiento de un **eje esclavo** según una secuencia de desplazamiento lineal dependiente de hasta cinco **ejes maestros**. El factor de acoplamiento define las relaciones entre los ejes maestro y el eje esclavo para cada eje maestro.

La proporción de movimiento del eje esclavo se forma mediante adición en base a las proporciones de movimiento del eje maestro multiplicado por los correspondientes factores de acoplamiento. Al activar un conjunto de ejes EG se puede forzar la sincronización del eje esclavo a una posición definida. Un conjunto de reductor se puede:

- Definir
- Activar
- Desactivar
- Borrar

.

El movimiento del eje esclavo se puede derivar, a elección, de los:

- Valores de consigna de los ejes maestros
- Valores reales de los ejes maestros

Como ampliación se pueden realizar también relaciones no lineales entre los ejes maestro y el eje esclavo a través de **tablas de levas** (ver cap. Comportamiento de contorneado). Los reductores electrónicos se pueden conectar en cascada; es decir, que el eje esclavo de un reductor electrónico puede actuar como eje maestro para otro reductor electrónico.

3.16.4.1 Definir reductor electrónico (EGDEF)

Un conjunto de ejes EG se define mediante la indicación del eje esclavo, y por lo menos uno y como máximo cinco ejes maestros con el correspondiente tipo de acoplamiento.

Requisitos

Requisito para la definición de un conjunto de ejes EG:

No debe haber ninguna definición de acoplamiento de eje para el eje esclavo (si la hay, se debe borrar dicha definición previamente con EGDEL).

Sintaxis

```
EGDEF(Eje esclavo,Eje maestro1,Tipo de acoplamiento1,Eje
maestro2,Tipo de acoplamiento2,...)
```

Descripción

EGDEF:	Definición de un reductor electrónico	
Eje esclavo:	Eje influido por los ejes maestro	
Eje maestro1 ,…, Eje maestro5	Ejes que tienen influencia en el eje esclavo	
Tipo de acoplamiento1 ,…, Tipo de acoplamiento5	Tipo de acoplamiento El tipo de acoplamiento no ha de ser igual para todos los ejes maestro y ha de indicarse individualmente para cada uno de los ejes maestro.	
	Valor:	**Significado:**
	0	El eje esclavo recibe la influencia del **valor real** del eje maestro correspondiente.
	1	El eje esclavo recibe la influencia del **valor de consigna** del eje maestro correspondiente.

Nota

Los factores de acoplamiento se preasignan con cero mediante la definición del conjunto de acoplamientos EG.

Nota

EGDEF activa la parada de decodificación previa. La definición de reductor con EGDEF también se tiene que utilizar de forma invariable si en sistemas actúan uno o varios ejes maestros en el eje esclavo a través de la **tabla de levas**.

Ejemplo

Código de programa	**Comentarios**
EGDEF(C,B,1,Z,1,Y,1)	; Definición de un conjunto de ejes EG. Los ejes maestros B, Z, Y tienen influencia en el eje esclavo C a través del valor de consigna.

3.16.4.2 Activación del reductor electrónico (EGON, EGONSYN, EGONSYNE)

Para la activación de un conjunto de ejes EG existen 3 variantes.

Sintaxis

Variante 1:

El conjunto de ejes EG se activa de manera selectiva sin sincronización con:

```
EGON(FA,"Modo de cambio de
secuencia",LA1,Z1,N1,LA2,Z2,N2,...,LA5,Z5,N5)
```

Variante 2:

El conjunto de ejes EG se activa de manera selectiva con sincronización con:

```
EGONSYN(FA,"Modo de cambio de secuencia",SynPosFA,
[,LAi,SynPosLAi,Zi,Ni])
```

Variante 3:

El conjunto de ejes EG se activa de manera selectiva con sincronización y el modo de aproximación se predetermina con:

```
EGONSYNE(FA,"Modo de cambio de secuencia",SynPosFA,Modo de
aproximación[,LAi,SynPosLAi,Zi,Ni])
```

Descripción

Variante 1:

FA	Eje arrastrado o esclavo	
`Modo de cambio de secuencia:`	Pueden utilizarse los modos siguientes:	
	`"NOC"`	El cambio de secuencia se lleva a cabo de inmediato
	`"FINE"`	El cambio de secuencia se lleva a cabo con "Marcha síncrona fina"
	`"COARSE"`	El cambio de secuencia se lleva a cabo con "Marcha síncrona basta"
	`"IPOSTOP"`	El cambio de secuencia se lleva a cabo con la sincronización del valor de consigna
LA1, ... LA5	Ejes maestros	
Z1, ... Z5	Numerador para el factor de acoplamiento i	
N1, ... N5	Denominador para el factor de acoplamiento i Factor de acoplamiento i = Numerador i/Denominador i	

Solo deben programarse los ejes maestros que se hayan especificado antes con EGDEF. Como mínimo, debe programarse un eje maestro.

Variante 2:

FA	Eje arrastrado o esclavo	
Modo de cambio de secuencia:	Pueden utilizarse los modos siguientes:	
	"NOC"	El cambio de secuencia se lleva a cabo de inmediato
	"FINE"	El cambio de secuencia se lleva a cabo con "Marcha síncrona fina"
	"COARSE"	El cambio de secuencia se lleva a cabo con "Marcha síncrona basta"
	"IPOSTOP"	El cambio de secuencia se lleva a cabo con la sincronización del valor de consigna
[,LAi,SynPosLAi,Zi,Ni]	(no escribir los corchetes) Mín. 1, máx. 5 secuencias de:	
LA1, ... LA5	Ejes maestros	
SynPosLAi	Posición síncrona para el eje maestro i.	
Z1, ... Z5	Numerador para el factor de acoplamiento i	
N1, ... N5	Denominador para el factor de acoplamiento i Factor de acoplamiento i = Numerador i/Denominador i	

Solo deben programarse ejes maestros que se hayan especificado antes con EGDEF. Con las "Posiciones síncronas" programadas para el eje esclavo (SynPosFA) y para los ejes maestros (SynPosLA) se definen las posiciones en las que el conjunto de acoplamientos es considerado *síncrono*. Si el reductor electrónico no se encuentra en estado síncrono al ser activado, se desplaza el eje esclavo a su posición síncrona definida.

Variante 3:

Los parámetros equivalen a los de la variante 2, además de:

Modo de aproximación:	Pueden utilizarse los modos siguientes:	
	"NTGT"	Desplazamiento optimizado en el tiempo al próximo hueco entre dientes
	"NTGP"	Desplazamiento optimizado en el recorrido al próximo hueco entre dientes
	"ACN"	Desplazamiento de eje giratorio en sentido de giro negativo, absoluto
	"ACP"	Desplazamiento de eje giratorio en sentido de giro positivo, absoluto
	"DCT"	Con optimización del tiempo para la posición síncrona programada
	"DCP"	Con optimización del recorrido para la posición síncrona programada

La variante 3 sólo actúa en ejes esclavo de valor módulo acoplados a ejes maestro de valor módulo. La optimización en el tiempo tiene en cuenta los límites de velocidad del eje esclavo.

Información adicional

Descripción de las variantes de activación

Variante 1:

Las posiciones de los ejes maestro y del eje esclavo en el momento de la activación se memorizan como "posiciones síncronas". Las "posiciones síncronas" se pueden leer con las variables de sistema `$AA_EG_SYN`.

Variante 2:

Si hay ejes módulo en un conjunto de acoplamiento, sus valores de posición se reducen en el módulo. De esta manera, se garantiza que se realice la aproximación a la siguiente posición síncrona posible (la *sincronización relativa*: p. ej., el siguiente hueco entre dientes). Cuando no se ha emitido para el eje esclavo la "Señal de desbloqueo superposición del eje esclavo" interfaz DB(30 +número de eje), DBX 26, bit 4, no se realiza el desplazamiento a la posición síncrona. En vez de ello, se para el programa en la secuencia EGONSYN y se emite la alarma autoborrante 16771 hasta que se activa la señal mencionada.

Variante 3:

La distancia entre dientes (grados) resulta de: 360 * Zi/Ni. En caso de que el eje esclavo esté parado en el momento de la llamada, el eje optimizado en el recorrido se comporta exactamente igual que el optimizado en el tiempo.

Si el eje esclavo ya está en marcha, se sincroniza con NTGP, independientemente de la velocidad actual del eje esclavo, en el siguiente hueco entre dientes. Si el eje esclavo ya está en marcha, se sincroniza con NTGT, en función de la velocidad actual del eje esclavo, en el siguiente hueco entre dientes. Para este fin, el eje también se frena si es necesario.

Tablas de levas

En caso de utilizar una **tabla de levas** para alguno de los ejes maestros, es obligatorio:

Ni	Poner a 0 el denominador del factor de acoplamiento de los acoplamientos lineales. (El denominador 0 no estaría permitido para acoplamientos lineales). El denominador cero indica para el control que
Zi	se tiene que interpretar como número de la tabla de levas a utilizar. La tabla de levas con el número indicado ya tiene que estar definida en el momento de la activación.
LAi	La especificación del eje maestro equivale a la especificación del eje maestro en caso de acoplamiento con factor de acoplamiento (acoplamiento lineal).

Encontrará más información sobre el uso de tablas de levas y la conexión en cascada de reductores electrónicos y su sincronización en:
Otras funciones:
Manual de funciones Ejes y cabezales

Comportamiento del reductor electrónico con Power On, RESET, cambio del modo de operación, búsqueda de secuencia

- Tras Power On no hay **ningún** acoplamiento activo.
- Los acoplamientos activos permanecen a pesar de RESET y cambio del modo de operación.
- En caso de búsqueda de secuencias no se ejecutan ni se recogen, sino que se pasan por alto las órdenes para conmutar, borrar o definir el reductor electrónico.

Variables de sistema del reductor electrónico

Con las variables de sistema del reductor electrónico, el programa de pieza puede averiguar y si fuera necesario, reaccionar frente a estados actuales de un conjunto de ejes EG.

Las variables del sistema del reductor electrónico se identifican de la siguiente manera:

$AA_EG_ ...

o bien

$VA_EG_ ...

Más información
Manual de funciones Variables de sistema

3.16.4.3 Desactivación del reductor electrónico (EGOFS, EGOFC)

Para la desactivación de un conjunto activo de ejes EG existen 3 variantes.

Programación

Variante 1:

Sintaxis	Descripción
`EGOFS(eje esclavo)`	Se desconecta el reductor electrónico. El eje esclavo se frena hasta detenerlo. La llamada provoca una parada de decodificación previa.

Variante 2:

Sintaxis	Descripción
`EGOFS(Eje esclavo,Eje maestro1, …,Eje maestro5)`	Esta parametrización del comando permite **selectivamente** eliminar la influencia que ejercen los ejes maestro individuales sobre el movimiento del eje esclavo.

Debe indicarse, por lo menos, un eje maestro. Se desconecta de forma intencionada la influencia de los ejes maestro sobre el eje esclavo. La llamada provoca una parada de decodificación previa. Si aún permanecen ejes maestros activos, el eje esclavo sigue el desplazamiento bajo su influencia. Una vez eliminadas todas las influencias de ejes maestros de esta manera, se frena el eje esclavo hasta detenerlo.

Variante 3:

Sintaxis	Descripción
`EGOFC(Cabezal esclavo1)`	Se desconecta el reductor electrónico. El cabezal esclavo sigue su marcha con la velocidad de giro/velocidad actuales en el momento de desconexión. La llamada provoca una parada de decodificación previa.

Nota

Esta variante sólo está permitida para cabezales.

3.16.4.4 Borrar la definición de un reductor electrónico (EGDEL)

Un conjunto de ejes EG debe estar desactivado antes de que pueda borrarse su definición.

Programación

Sintaxis	Descripción
`EGDEL(eje esclavo)`	Se borra la definición de acoplamiento del conjunto de ejes. Hasta alcanzar el número máximo de conjuntos de ejes activados simultáneamente pueden definirse nuevamente otros conjuntos de ejes con `EGDEF`. La llamada provoca una parada de decodificación previa.

3.16.4.5 Avance por vuelta (G95)/Reductor electrónico (FPR)

Con el comando FPR también puede indicarse el eje esclavo de un reductor electrónico como eje determinante del avance por vuelta. En tal caso rige el siguiente comportamiento:

- El avance depende de la velocidad nominal del eje esclavo del reductor electrónico.
- La velocidad de consigna se calcula en base a las velocidades de los cabezales maestro y los ejes maestros tipo módulo (que no son ejes de contorneado) y sus factores de acoplamiento asignados.
- No se tienen en cuenta componentes de velocidad de ejes maestro lineales o bien no módulo y movimientos superpuestos del eje esclavo.

3.16.5 Cabezal síncrono

En modo de cabezal síncrono siempre tenemos un cabezal maestro (CM) y un cabezal esclavo (CE). De esta forma se define el **Par de cabezales síncronos**. El cabezal esclavo sigue los movimientos del cabezal maestro desde el momento en que se activa un acoplamiento (modo síncrono) y atendiendo al tipo de acoplamiento definido mediante los parámetros.

El par de cabezales síncronos se puede definir de forma permanente para una determinada máquina mediante los datos de máquina específicos de canal, o bien se puede definir por el usuario desde el programa de pieza. Se pueden definir hasta 2 pares de cabezales síncronos simultáneamente para un mismo canal CN.

Desde el programa de pieza, el acoplamiento se puede:

- Definir o modificar
- Conectar
- Desconectar
- Borrar

Además, dependiendo de la versión de software, se puede:

- Esperar a que se cumplan las condiciones para la sincronización
- Modificar el comportamiento para el cambio de secuencia

- Seleccionar el modo de acoplamiento como acoplamiento de valores de consigna o de valores reales, o bien definir el decalaje angular entre los cabezales maestro y esclavo
- Aplicar una programación anterior del cabezal esclavo al activar el acoplamiento
- Corregir una desviación de sincronización medida o ya conocida

3.16.5.1 Cabezal síncrono: Programación (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC)

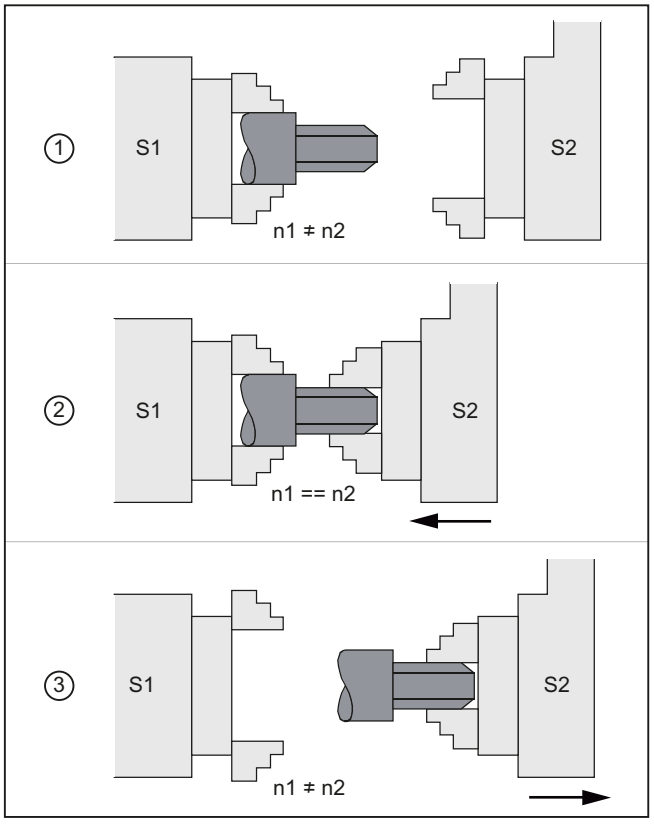
La función "Cabezal síncrono" permite un desplazamiento con sincronismo de la velocidad de los cabezales esclavo (FS) y maestro (LS) con relación de transmisión programada.

La función ofrece los siguientes modos:

- Sincronismo de la velocidad de giro ($n_{FS} = n_{LS}$)
- Sincronismo de posición ($\phi_{FS} = \phi_{LS}$)
- Sincronismo de posición con un decalaje angular adicional ($\phi_{FS} = \phi_{LS}$)

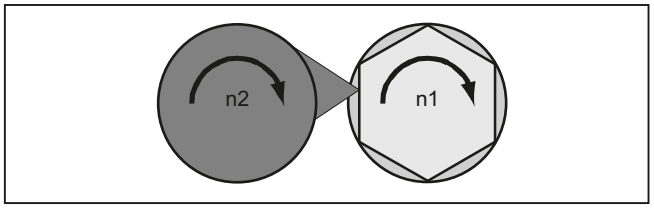
Ejemplos de aplicación:

- Transferencia al vuelo de la pieza, p. ej. para mecanizado del lado posterior, relación de transmisión: 1:1



① Sincronismo de velocidad
② Transferencia de pieza
③ Mecanizado del lado posterior

- Mecanizado de poliedros (torneado de polígonos), sincronismo de velocidad, relación de transmisión: n_1:n_2



Sintaxis

```
COUPDEF(<FS>,<LS>,<ZFS>,<NLS>,<Cambio de secuencia>,<Tipo de
acoplamiento>)
COUPON(<FS>,<LS>,<POSFS>)
COUPONC(<FS>,<LS>)
COUPOF(<FS>,<LS>,<POSFS>,<POSLS>)
COUPOFS(<FS>,<LS>)
COUPOFS(<FS>,<LS>,<POSFS>)
```

```
COUPRES(<FS>,<LS>)
COUPDEL(<FS>,<LS>)
WAITC(<FS>,<Cambio de secuencia>,<LS>,<Cambio de secuencia>)
```

Nota

Notación abreviada

Con las instrucciones COUPOF, COUPOFS, COUPRES y COUPDEL es posible una notación abreviada sin especificar el cabezal maestro.

Descripción

COUPDEF:	Definición/modificación del acoplamiento específica del usuario
COUPON:	Activación de un acoplamiento. El cabezal esclavo se sincroniza con el cabezal maestro partiendo de la velocidad de giro actual.
COUPONC:	Aplicación del acoplamiento con la programación anterior de M3 S... o M4 S... durante la activación. Se aplica inmediatamente una velocidad de giro diferencial del cabezal esclavo.
COUPOF:	Desactivación de un acoplamiento. • Con cambio de secuencia inmediato: COUPOF(<S2>,<S1>) • Cambio de secuencia tras sobrepasar las posiciones de desactivación <POSFS> o <POSLS>: COUPOF(<S2>,<S1>,<POSFS>) COUPOF(<S2>,<S1>,<POSFS>,<POSLS>)
COUPOFS:	Desactivación de un acoplamiento con parada del cabezal esclavo. Cambio de secuencia con la máxima rapidez con cambio de secuencia inmediato: COUPOFS(<S2>,<S1>) Cambio de secuencia tras sobrepasar la posición de desconexión: COUPOFS(<S2>,<S1>,<POSFS>)
COUPRES:	Puesta a cero de los parámetros de un acoplamiento a los DM y DO configurados
COUPDEL:	Borrado de un acoplamiento definido por el usuario
WAITC:	Esperar a que se cumplan las condiciones de sincronización (se anulan los NOC a IPO en caso de cambio de secuencia)
<FS>:	Nombre del cabezal esclavo
Parámetros opcionales:	
<LS>:	Nombre del cabezal maestro Indicación con número de cabezal: p. ej. S2, S1
<ZFS>, <NLS>:	Relación de transformación entre el cabezal esclavo y el cabezal maestro. <ZFS>/<NLS> = numerador/denominador Ajuste predeterminado: <ZFS>/<NLS> = 1.0; opcionalmente puede indicarse el denominador

<Cambio de secuencia>:	Comportamiento en cambio de secuencia El cambio de secuencia se realiza:	
	"NOC"	inmediatamente
	"FINE"	al alcanzar "marcha síncrona fina"
	"COARSE"	al alcanzar "marcha síncrona basta"
	"IPOSTOP"	al alcanzar IPOSTOP, es decir, tras la marcha síncrona del valor de consigna (ajuste previo)
	El comportamiento en cambio de secuencia es modal.	
<Tipo de acoplamiento>:	Modo de acoplamiento: Acoplamiento entre cabezal esclavo y cabezal maestro	
	"DV"	Acoplamiento de valor de consigna (tipo de acoplamiento por defecto)
	"AV"	Acoplamiento de valores reales
	"VV"	Acoplamiento de velocidad
	El tipo de acoplamiento es válido de forma modal.	
<POSFS>:	Decalaje angular entre los cabezales maestro y esclavo	
	Rango de valores:	0°… 359,999°
<POSFS>,<POSLS>:	Posiciones de desconexión de cabezal esclavo y maestro "El cambio de secuencia se habilita después de sobrepasar POS_{FS}, POS_{LS}"	
	Rango de valores:	0°… 359,999°

Ejemplos

Programación de cabezales maestro y esclavo

Código de programa	**Comentarios**
	Cabezal guía = Cabezal maestro = Cabezal 1
	Cabezal esclavo = Cabezal 2
N05 M3 S3000 M2=4 S2=500	El cabezal maestro gira a 3000 rpm, el cabezal esclavo gira a 500 rpm.
N10 COUPDEF(S2,S1,1,1,"NOC","Dv")	Definición del acoplamiento (también puede configurarse).
...	
N70 SPCON	Tomar cabezal guía en regulación de posición (acoplamiento de valores de consigna).
N75 SPCON(2)	Tomar cabezal esclavo en regulación de posición.
N80 COUPON(S2,S1,45)	Acoplamiento al vuelo con offset = 45 grados.
...	
N200 FA[S2]=100	Velocidad de posicionado = 100 grados/min
N205 SPOS[2]=IC(-90)	Superponer 90 grados en sentido negativo al valor de posición.
N210 WAITC(S2,"Fine")	Esperar a la sincronización "fina".
N212 G1 X... Y... F...	Mecanizado
...	

Código de programa	Comentarios
N215 SPOS[2]=IC(180)	Superponer 180 grados en sentido positivo al valor de posición.
N220 G4 S50	Tiempo de espera = 50 vueltas del cabezal maestro
N225 FA[S2]=0	Activar la velocidad configurada (DM).
N230 SPOS[2]=IC(-7200)	20 vueltas. Desplazar con la velocidad configurada en sentido negativo.
...	
N350 COUPOF(S2,S1)	Desacoplar al vuelo, S=S2=3000
N355 SPOSA[2]=0	Detener FS a cero grados.
N360 G0 X0 Y0	
N365 WAITS(2)	Esperar a cabezal 2.
N370 M5	Detener FS.
N375 M30	

Programación de una velocidad de giro diferencial

Código de programa	Comentarios
	Cabezal guía = Cabezal maestro = Cabezal 1
	Cabezal esclavo = Cabezal 2
N01 M3 S500	El cabezal maestro gira a 500 rpm.
N02 M2=3 S2=300	El cabezal esclavo gira a 300 rpm.
...	
N10 G4 F1	Tiempo de espera del cabezal maestro.
N15 COUPDEF (S2,S1,-1)	Factor de acoplamiento con relación de transformación -1:1
N20 COUPON(S2,S1)	Activar acoplamiento. La velocidad del cabezal esclavo resulta de la velocidad del cabezal maestro y del factor de acoplamiento.
...	
N26 M2=3 S2=100	Programación de una velocidad de giro diferencial.

Ejemplos de la aplicación de un movimiento para la velocidad de giro diferencial

1. Activación del acoplamiento con la programación anterior del cabezal esclavo con COUPON

Código de programa	Comentarios
	Cabezal guía = Cabezal maestro = Cabezal 1
	Cabezal esclavo = Cabezal 2
N05 M3 S100 M2=3 S2=200	El cabezal maestro gira a 100 rpm; el cabezal esclavo, a 200 rpm.
N10 G4 F5	Tiempo de espera = 5 segundos del cabezal maestro
N15 COUPDEF(S2,S1,1)	La relación de transformación del cabezal esclavo con respecto al cabezal maestro es 1,0 (ajuste predeterminado).
N20 COUPON(S2,S1)	Acoplamiento al vuelo al eje maestro.
N10 G4 F5	El cabezal esclavo gira a 100 rpm.

2. Activación del acoplamiento con la programación anterior del cabezal esclavo con COUPONC

Código de programa	Comentarios
	Cabezal guía = Cabezal maestro = Cabezal 1
	Cabezal esclavo = Cabezal 2
N05 M3 S100 M2=3 S2=200	El cabezal maestro gira a 100 rpm; el cabezal esclavo, a 200 rpm.
N10 G4 F5	Tiempo de espera = 5 segundos del cabezal maestro
N15 COUPDEF(S2,S1,1)	La relación de transformación del cabezal esclavo con respecto al cabezal maestro es 1,0 (ajuste predeterminado).
N20 COUPONC(S2,S1)	Acoplamiento al vuelo al cabezal maestro y aplicación de la velocidad de giro anterior a S2.
N10 G4 F5	S2 gira a 100 rpm + 200 rpm = 300 rpm

3. Activación del acoplamiento con el cabezal esclavo parado con COUPON

Código de programa	Comentarios
	Cabezal guía = Cabezal maestro = Cabezal 1
	Cabezal esclavo = Cabezal 2
N05 SPOS=10 SPOS[2]=20	Cabezal esclavo S2 en modo de posicionamiento.
N15 COUPDEF(S2,S1,1)	La relación de transformación del cabezal esclavo con respecto al cabezal maestro es 1,0 (ajuste predeterminado).
N20 COUPON(S2,S1)	Acoplamiento al vuelo al eje maestro.
N10 G4 F1	Se cierra el acoplamiento, S2 permanece a 20 grados.

4. Activación del acoplamiento con el cabezal esclavo parado con COUPONC

Nota

Modo de posicionamiento o de eje

Si el cabezal esclavo se encuentra en el modo de posicionamiento o en el modo Eje antes del acoplamiento, se comporta del mismo modo con COUPON(<FS>,<LS>) que con COUPONC(<FS>,<LS>).

Nota

Cabezal maestro y modo Eje

Si el cabezal maestro se encuentra en el modo Eje antes de la definición del acoplamiento, tras la activación de éste también actúa el límite de velocidad del dato de máquina:

DM32000 $MA_MAX_AX_VELO (velocidad de eje máxima)

Para evitar este comportamiento, el eje debe pasar al modo Cabezal (M3 S... o M4 S...) antes de la definición del acoplamiento.

Información adicional

Acoplamiento configurado

En el acoplamiento configurado, los cabezales maestro y esclavo se definen mediante datos de máquina. Los cabezales configurados no pueden modificarse en el programa de pieza. El acoplamiento puede parametrizarse con `COUPDEF` en el programa de pieza (siempre que no exista protección contra escritura).

Acoplamiento definido por el usuario

Con `COUPDEF` puede definirse o modificarse un acoplamiento en el programa de pieza. Si ya hay un acoplamiento activo, éste debe borrarse primero con `COUPDEL` antes de definir uno nuevo.

Un acoplamiento queda totalmente definido por:

```
COUPDEF(<FS>,<LS>,<ÜFS>,<ÜLS>, Comportamiento en cambio de
secuencia, Tipo de acoplamiento)
```

Cabezal esclavo (FS) y cabezal maestro (LS)

Los nombres de eje para el cabezal esclavo (FS) y el cabezal maestro (LS) se utilizan para definir un acoplamiento de forma unívoca. Los nombres de eje deben programarse con la instrucción `COUPDEF`. Los otros parámetros de acoplamiento son modalmente activos y sólo deben programarse cuando sea necesario modificarlos.

Ejemplo:

```
COUPDEF(S2,S1)
```

Relación de transmisión

La relación de transmisión se indica como relación de velocidad entre cabezal esclavo y cabezal maestro:

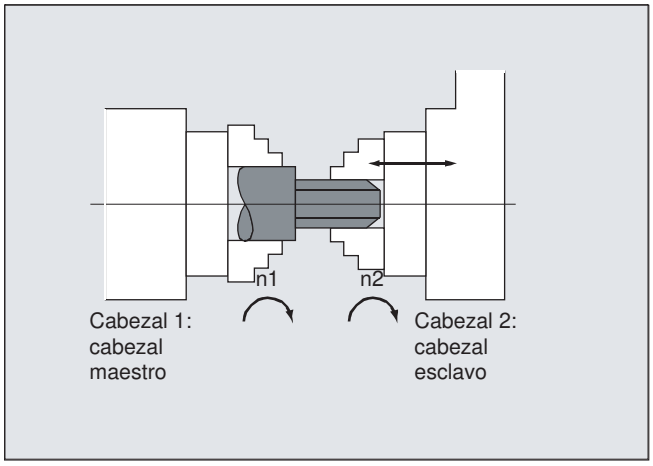
Cabezal esclavo/cabezal maestro = numerador/denominador

El numerador debe programarse. El denominador no necesita programarse. En este caso, al denominador se le asignará el valor por defecto 1.0.

Ejemplo:

Cabezal esclavo S2 y cabezal maestro S1, relación de transmisión = 1/1

```
COUPDEF(S2, S1, 1.0)
```



Nota

La relación de transformación también puede modificarse cuando el acoplamiento está activado y los cabezales girando.

Comportamiento de cambio de secuencia NOC, FINE, COARSE, IPOSTOP

En la programación del comportamiento en cambio de secuencia se puede utilizar la siguiente notación abreviada:

- "NO": inmediato (ajuste previo)
- "FI": al alcanzar "marcha síncrona fina"
- "CO": al alcanzar "marcha síncrona basta"
- "IP": al alcanzar IPOSTOP, es decir, tras la marcha síncrona del valor de consigna

Modo de acoplamiento

Nota

El tipo de acoplamiento solo puede modificarse cuando se encuentra desactivado.

Activación del modo síncrono COUPON, <POSFS>

- Activación del acoplamiento con cualquier decalaje angular entre LS y FS:
  - COUPON(S2,S1)
  - COUPON(S2)
- Activación del acoplamiento con decalaje angular <POSFS>
  <POSFS> se refiere a la posición 0° del cabezal maestro en sentido de giro positivo
  Rango de valores <POSFS>: 0°… 359,999°
  - COUPON(S2,S1,30)

Nota

El decalaje angular puede modificarse también con el acoplamiento activo.

Posicionado del cabezal esclavo

Es posible posicionar FS independientemente del LS con un rango de ±180° aunque esté activado el acoplamiento de cabezal síncrono.

- Posicionamiento del cabezal FS con `SPOS`
  Ejemplo: `SPOS[2] = IC(-90)`
  Encontrará más información sobre `SPOS` en:
  Más información
  Posicionamiento de cabezales (SPOS, SPOSA, M19, M70, WAITS) (Página 121)

Velocidad diferencial

La velocidad diferencial se produce en el modo de control de velocidad de giro, estando activo el acoplamiento de cabezal síncrono, al superponerse con signo una velocidad de FS basada en el desplazamiento del LS y una velocidad de FS basada en la programación de cabezal:

- Acoplamiento de cabezal síncrono con `COUPONC`
- `S<FS>=<velocidad>` [`M<FS>=<sentido de giro>]`

  Nota

  Condiciones

  - Con el sentido de giro `M3`/`M4`, debe reprogramarse también la velocidad de giro `S...`
  - Las superposiciones de una velocidad de giro de cabezal (`M<sentido de giro> S<FS>`) debido al desplazamiento del LS estando activo el acoplamiento de cabezal síncrono `COUPONC` solo son válidas si está habilitada la superposición.
  - La dinámica del cabezal maestro debe limitarse hasta tal punto que no se sobrepasen los límites de dinámica del cabezal esclavo al producirse una superposición de éste.

  Encontrará más información sobre la velocidad de giro diferencial en:
  Más información
  Manual de funciones Ejes y cabezales

Velocidad, aceleración: FA, ACC, OVRA, VELOLIMA

La velocidad y la aceleración por eje de un cabezal esclavo pueden programarse con:

- `FA[SPI(S<n>)]` o `FA[S<n>]` (velocidad por eje)
- `ACC[SPI(S<n>)]` o `ACC[S<n>]` (aceleración por eje)
- `OVRA[SPI(S<n>)]` o `OVRA[S<n>]` (corrección por eje)
- `VELOLIMA[SPI(S<n>)]` o `VELOLIMA[S<n>]` (aumento o reducción de la velocidad por eje)

Con `<n>` = 1, 2, 3... (número de cabezal de los cabezales esclavos)

Información adicional
Ajuste de avance (Página 108)

Nota

Una reducción o un aumento del tirón axial máximo no tiene efecto en los cabezales.

Encontrará más información sobre la dinámica por eje en:
Más información
Manual de funciones Ejes y cabezales

Comportamiento en cambio de secuencia programable WAITC

`WAITC` permite predefinir el comportamiento en cambio de secuencia, p. ej. después de modificar parámetros de acoplamiento o procesos de posicionado, con diferentes condiciones de marcha síncrona (basta, fina, `IPOSTOP`). Si no se indican condiciones de marcha síncrona, se aplica el comportamiento en cambio de secuencia definido en `COUPDEF`.

Ejemplos

- Esperar hasta que se cumpla la condición de marcha síncrona `FINE` en el cabezal esclavo S2 y `COARSE` en el cabezal esclavo S4: `WAITC(S2,"FINE",S4,"COARSE")`
- Esperar hasta que se cumplan las condiciones de marcha síncrona conforme a `COUPDEF`: `WAITC( )`

Desactivación del acoplamiento COUPOF

Con `COUPOF` se puede predefinir el comportamiento de desactivación del acoplamiento:

- Desactivación del acoplamiento con cambio de secuencia inmediato:
  - `COUPOF(S2,S1)` (indicando el cabezal maestro)
  - `COUPOF(S2)` (sin indicar el cabezal maestro)
- Desactivación del acoplamiento tras sobrepasar ciertas posiciones de desactivación. El cambio de secuencia tiene lugar después de sobrepasar las posiciones de desactivación.
  - `COUPOF(S2,S1,150)` (posición de desactivación FS: 150°)
  - `COUPOF(S2,S1,150,30)` (posición de desactivación FS: 150°, de LS: 30°)

Desactivación del acoplamiento con parada del cabezal esclavo COUPOFS

Con `COUPOFS` se puede predefinir el comportamiento de desactivación del acoplamiento con parada del cabezal esclavo:

- Desactivación del acoplamiento con parada del cabezal esclavo y cambio de secuencia inmediato:
  - `COUPOFS(S2,S1)` (indicando el cabezal maestro)
  - `COUPOFS(S2)` (sin indicar el cabezal maestro)
- Desactivación del acoplamiento tras sobrepasar ciertas posiciones de desactivación con parada del cabezal esclavo. El cambio de secuencia tiene lugar después de sobrepasar las posiciones de desactivación.
  - `COUPOFS(S2,S1,150)` (posición de desactivación FS: 150°)

Borrado de acoplamientos COUPDEL

Mediante `COUPDEL` se borra el acoplamiento:

- `COUPDEL(S2,S1)` (indicando el cabezal maestro)
- `COUPDEL(S2)` (sin indicar el cabezal maestro)

Puesta a cero de los parámetros de acoplamiento COUPRES

`COUPRES` se utiliza para activar los valores del acoplamiento parametrizados en los datos de máquina y del operador:

- `COUPRES(S2,S1)` (indicando el cabezal maestro)
- `COUPRES(S2)` (sin indicar el cabezal maestro)

Variable de sistema

- Estado de acoplamiento actual del cabezal esclavo
  El estado de acoplamiento actual de un cabezal esclavo puede leerse codificado al bit mediante:
  `<Valor> = $AA_COUP_ACT[<FS>]`

Bit	<Valor>	Descripción
-	0	ningún acoplamiento activo
2	4	acoplamiento de cabezal síncrono activo

Nota

- Todos los demás valores hacen referencia al modo de eje
- Si el cabezal es un cabezal esclavo de varios acoplamientos, se retorna como valor la suma de los estados de acoplamiento de todos los acoplamientos.

- Decalaje angular actual
  El decalaje angular actual del cabezal esclavo respecto al cabezal maestro puede leerse mediante:
  - `$AA_COUP_OFFS[<FS>]` (decalaje angular del valor de consigna)
  - `$VA_COUP_OFFS[<FS>]` (decalaje angular del valor real)

  Ejemplo de aplicación
  Corrección del diferencial de decalaje angular en el programa CN una vez cancelado el modo de seguimiento:
  Diferencial de decalaje angular = decalaje angular programado - variable del sistema

Información adicional

Para información detallada sobre las variables de sistema, ver:

Manual de listas Variables de sistema

3.16.6 Acoplamiento genérico (CP...)

El "acoplamiento genérico" es una función general de acoplamiento que reúne todas las propiedades de los modos de acoplamiento disponibles (arrastre de ejes, acoplamiento de valor maestro, reductor electrónico y cabezal síncrono).

Esta función permite programar con gran flexibilidad:

- El usuario puede seleccionar libremente las propiedades de acoplamiento necesarias para su aplicación (principio de sistemas modulares).
- Cada propiedad de acoplamiento se puede programar individualmente.
- Las propiedades de un acoplamiento definido (p. ej., factor de acoplamiento) pueden modificarse.
- Es posible un uso posterior de otras propiedades de acoplamiento.
- El sistema de referencia de coordenadas del eje esclavo (sistema de coordenadas básico o sistema de coordenadas de máquina) es programable.
- Determinadas propiedades de acoplamiento pueden programarse también en acciones síncronas.
  Información adicional: Manual de funciones Acciones síncronas

Nota

Las llamadas de acoplamiento anteriores para el arrastre de ejes (TRAIL*), acoplamiento de valor maestro (LEAD*), reductor electrónico (EG*) y cabezal síncrono (COUP*) se siguen soportando mediante ciclos de adaptación.

Resumen de todas las palabras reservadas y propiedades de acoplamiento

La siguiente tabla ofrece un resumen de todas las palabras reservadas del acoplamiento genérico y de las propiedades de acoplamiento programables:

Palabra reservada	Propiedad de acoplamiento/ significado	Sintaxis
CPDEF	Creación de un módulo de acoplamiento	CPDEF=(<FAx>)
CPDEL	Borrado de un módulo de acoplamiento	CPDEL=(<FAx>)
CPLA	Definición de un eje maestro	CPLA[<FAx>]=(<LAx>)
CPLDEF	Definición de un eje maestro y creación de un módulo de acoplamiento (también posible con CPDEF + CPLA)	CPLDEF[<FAx>]=(<LAx>) o bien CPDEF=(<FAx>) CPLA[<FAx>]=(<LAx>)
CPLDEL	Borrado de un eje maestro de un módulo de acoplamiento (también posible con CPDEL + CPLA)	CPLDEL[<FAx>]=(<LAx>) o bien CPDEL=(<FAx>) CPLA[<FAx>]=(<LAx>)

Palabra reservada	Propiedad de acoplamiento/ significado	Sintaxis		
CPON	Activación de un módulo de acoplamiento	CPON=(<FAx>)		
CPOF	Desactivación de un módulo de acoplamiento	CPOF=(<FAx>)		
CPLON	Activación de un eje maestro de un módulo de acoplamiento	CPLON[<FAx>]=<LAx>		
CPLOF	Desactivación de un eje maestro de un módulo de acoplamiento	CPLOF[<FAx>]=<LAx>		
CPLNUM	Numerador del factor de acoplamiento	CPLNUM[FAx,LAx]=<Valor>		
CPLDEN	Denominador del factor de acoplamiento	CPLDEN[FAx,LAx]=<Valor>		
CPLCTID	Número de la tabla de levas	CPLCTID[FAx,LAx]=<Valor>		
CPLSETVAL	Referencia de acoplamiento	CPLSETVAL[FAx,LAx]="<Referencia de acoplamiento>"		
		"<Referencia de acoplamiento>":	"CMDPOS"	Acoplamiento de valores de consigna
			"CMDVEL"	Acoplamiento de velocidad
			"ACTPOS"	Acoplamiento de valores reales
CPFRS	Sistema de referencia de coordenadas	CPFRS[FAx]="<Referencia de coordenadas>"		
		"<Referencia de coordenadas>":	"BCS"	Sistema de coordenadas básico
			"MCS"	Sistema de coordenadas de máquina
CPBC	Criterio de cambio de secuencia	CPBC[FAx]="<Criterio de cambio de secuencia>"		
		"<Criterio de cambio de secuencia>":	"NOC"	El cambio de secuencia se lleva a cabo independientemente del estado de acoplamiento.
			"IPOSTOP"	El cambio de secuencia se lleva a cabo con la marcha síncrona del valor de consigna.
			"COARSE"	El cambio de secuencia se lleva a cabo con la marcha síncrona del valor real "basta".
			"FINE"	El cambio de secuencia se lleva a cabo con la marcha síncrona del valor real "fina".
CPFPOS + CPON	Posición síncrona del eje esclavo al conectar	CPON=FAx CPFPOS[FAx]=<Valor>		

Palabra reservada	Propiedad de acoplamiento/ significado	Sintaxis
CPLPOS + CPON	Posición síncrona del eje maestro al conectar	CPLPOS[FAx,LAx]=<Valor>

Palabra reservada	Propiedad de acoplamiento/ significado	Sintaxis		
CPFMSON	Modo de sincronización	CPFMSON[FAx]="<Modo de sincronización>"		
		"<Modo de sincronización>":	"CFAST"	El acoplamiento se cierra con el tiempo optimizado.
			"CCOARSE"	El acoplamiento solo se activa cuando la posición del eje esclavo requerida por la regla de acoplamiento se encuentra en el área de la posición actual del eje esclavo.
			"NTGT"	El siguiente hueco de diente se desplaza con el tiempo optimizado.
			"NTGP"	El siguiente hueco de diente se desplaza con el trayecto optimizado.
			"NRGT"	El siguiente segmento conforme a la relación entre número de filetes y número de dientes se desplaza con el tiempo optimizado.
			"NRGP"	El siguiente segmento conforme a la relación entre número de filetes y número de dientes se desplaza con el trayecto optimizado.
			"ACN"	¡Solo con ejes giratorios! El eje giratorio se desplaza en sentido negativo a la posición síncrona. La sincronización se produce de inmediato.
			"ACP"	¡Solo con ejes giratorios! El eje giratorio se desplaza en sentido positivo a la posición síncrona. La sincronización se produce de inmediato.
			"DCT"	¡Solo con ejes giratorios! El eje giratorio se desplaza hacia la posición síncrona con el tiempo optimizado. La sincronización se produce de inmediato.
			"DCP"	¡Solo con ejes giratorios! El eje giratorio se desplaza hacia la posición síncrona programada con el trayecto optimizado. La sincroniza-

Palabra reservada	Propiedad de acoplamiento/ significado	Sintaxis		
				ción se produce de inmediato.
CPFMON	Comportamiento del eje esclavo en la activación	CPFMON[FAx]="<Comportamiento de activación>"		
		"<Comportamiento de activación>":	"STOP"	¡Solo con cabezales! Se para todo movimiento activo del cabezal esclavo antes de la conexión.
			"CONT"	¡Solo con cabezales y ejes de marcha principal! El movimiento actual del eje/cabezal esclavo se transfiere al acoplamiento como movimiento inicial.
			"ADD"	¡Solo con cabezales! Las partes de movimiento del acoplamiento contribuyen también al movimiento superpuesto actual, es decir, el movimiento actual del eje/cabezal esclavo se mantiene como movimiento superpuesto.
CPFMOF	Comportamiento del eje esclavo con desactivación completa	CPFMOF[FAx]="<Comportamiento de desactivación>"		
		"<Comportamiento de desactivación>":	"STOP"	Parada del eje/cabezal esclavo. Se frena todo movimiento superpuesto activo hasta pararse. Después se abre el acoplamiento
			"CONT"	¡Solo con cabezales y ejes de marcha principal! El cabezal esclavo sigue su marcha con la velocidad de giro/velocidad actuales en el momento de desconexión.
CPFPOS + CPOF	Posición de desactivación del eje esclavo al desconectar	CPOF=(FAx) CPFPOS[FAx]=<Valor>		

Palabra reservada	Propiedad de acoplamiento/ significado	Sintaxis		
CPMRESET	Comportamiento de acoplamiento con RESET	CPMRESET[FAx]="<Comportamiento Reset>"		
		"<Comportamiento Reset>":	"NONE"	El estado actual del acoplamiento se mantiene.
			"ON"	Si se ha creado el módulo de acoplamiento correspondiente, el acoplamiento se activa. Todas las relaciones de eje maestro definidas se activan. Esto sucede también cuando todas las relaciones de eje maestro o partes de ellas ya estaban activas, lo que significa que cuando un acoplamiento está completamente activo tiene lugar una nueva sincronización.
			"OF"	Se frena todo movimiento superpuesto activo hasta pararse. Después se desactiva el acoplamiento. Si el módulo de acoplamiento correspondiente se había creado sin una definición explícita (CPDEF), dicho módulo será eliminado. En caso contrario se mantiene y, por tanto, se puede seguir utilizando.
			"OFC"	¡Solo posible con cabezales! El cabezal esclavo sigue su marcha con la velocidad de giro/velocidad actuales en el momento de desconexión. El acoplamiento se desactiva. Si el módulo de acoplamiento correspondiente se había creado sin una definición explícita (CPDEF), dicho módulo será eliminado. En caso contrario se mantiene y, por tanto, se puede seguir utilizando.
			"DEL"	Se frena todo movimiento superpuesto activo hasta pararse. A continuación, el acoplamiento se desactiva y se borra.
			"DELC"	¡Solo posible con cabezales!

Palabra reservada	Propiedad de acoplamiento/ significado	Sintaxis		
				El cabezal esclavo sigue su marcha con la velocidad de giro/velocidad actuales en el momento de desconexión. A continuación, el acoplamiento se desactiva y se borra.
CPMSTART	Comportamiento de acoplamiento con inicio de programa de pieza	CPMSTART[FAx]="<Comportamiento de arranque>"		
		"<Comportamiento de arranque>":	"NONE"	El estado actual del acoplamiento se mantiene.
			"ON"	Acoplamiento activado. Todas las relaciones de eje maestro definidas se activan. Esto sucede también cuando todas las relaciones de eje maestro o partes de ellas ya estaban activas, lo que significa que cuando un acoplamiento está completamente activo tiene lugar una nueva sincronización.
			"OF"	El acoplamiento se desactiva. Si el módulo de acoplamiento correspondiente se había creado sin una definición explícita (CPDEF), dicho módulo será eliminado. En caso contrario se mantiene y, por tanto, se puede seguir utilizando.
			"DEL"	A continuación, el acoplamiento se desactiva y se borra.
CPMPRT	Comportamiento de acoplamiento al inicio de programa de pieza con búsqueda de secuencia vía test de programa	CPMPRT[FAx]="<Comportamiento de arranque>"		
		"<Comportamiento de arranque>":	Ver CPMSTART	
CPLINTR	Valor de decalaje para el valor de entrada de un eje maestro	CPLINTR[FAx,LAx]=<Valor>		
CPLINSC	Factor de escala para el valor de entrada de un eje maestro	CPLINSC[FAx,LAx]=<Valor>		
CPLOUTTR	Valor de decalaje para el valor de salida de un acoplamiento	CPLOUTTR[FAx,LAx]=<Valor>		
CPLOUTSC	Factor de escala del valor de salida de un acoplamiento	CPLOUTSC[FAx,LAx]=<Valor>		
CPSYNCOP	Umbral para la marcha síncrona de posición "basta"	CPSYNCOP[FAx]=<Valor>		

Palabra reservada	Propiedad de acoplamiento/ significado	Sintaxis		
CPSYNFIP	Umbral para la marcha síncrona de posición "fina"	CPSYNFIP[FAx]=<Valor>		
CPSYNCOP2	Segundo valor umbral para la marcha síncrona de posición "basta"	CPSYNCOP2[FAx]=<Valor>		
CPSYNFIP2	Segundo valor umbral para la marcha síncrona de posición "fina"	CPSYNFIP2[FAx]=<Valor>		
CPSYNCOV	Valor umbral para la marcha síncrona de velocidad "basta"	CPSYNCOV[FAx]=<Valor>		
CPSYNFIV	Valor umbral para la marcha síncrona de velocidad "fina"	CPSYNFIV[FAx]=<Valor>		
CPMBRAKE	Comportamiento del eje esclavo con determinados comandos y señales de parada	CPMBRAKE[FAx]=<Valor codificado en bits>		
CPMVDI	Comportamiento del eje esclavo con determinadas señales de interfaz CN/PLC	CPMVDI[FAx]=<Valor codificado en bits>		
CPMALARM	Supresión de salidas de alarma especiales referidas a acoplamientos	CPMALARM[FAx]=<Valor codificado en bits>		
CPSETTYPE	Tipo de acoplamiento	CPSETTYPE[FAx]="<Tipo de acoplamiento>"		
		"<Tipo de acoplamiento>":	"CP"	Programabilidad libre
			"TRAIL"	Tipo de acoplamiento "Arrastre de ejes"
			"LEAD"	Tipo de acoplamiento "Acoplamiento de valor maestro"
			"EG"	Tipo de acoplamiento "Reductor electrónico"
			"COUP"	Tipo de acoplamiento "Cabezal síncrono"

FAx: Eje/cabezal esclavo

LAx: Eje/cabezal maestro

Nota

Las propiedades de acoplamiento que no se programan explícitamente (en el programa de pieza o en las acciones síncronas) se activan con sus ajustes por defecto.

Dependiendo del ajuste de la palabra reservada CPSETTYPE, en lugar de los ajustes por defecto (CPSETTYPE="CP") también se pueden activar las propiedades de acoplamiento predefinidas.

Información adicional

Para información detallada sobre el acoplamiento genérico:

- Manual de funciones Ejes y cabezales

3.16.7 Control tangencial

3.16.7.1 Definición del acoplamiento (TANG)

A través del procedimiento predefinido TANG(...) se define un acoplamiento tangencial entre un eje giratorio como eje esclavo y dos ejes geométricos como ejes maestro. El eje esclavo se orienta de forma continua hacia la trayectoria tangente de los ejes maestro.

Nota

Factor de acoplamiento

Un factor de acoplamiento de 1 no se necesita programar explícitamente.
Con el factor de acoplamiento –1 se invierte el sentido del eje tangencial.

Sintaxis

```
TANG(<Eje esclavo>, <Eje maestro_1>, <Eje maestro_2>, <Factor de
acoplamiento>, <Sistema de coordenadas>, <Optimización>)
```

Descripción

`TANG(...)`:	Definición de un acoplamiento tangencial	
`<Eje esclavo>`:	Nombre del eje esclavo (eje giratorio)	
	Tipo de dato:	AXIS
	Rango de valores:	Nombres de eje de canal
`<Eje maestro_1>` `<Eje maestro_2>`:	Nombres de los ejes maestro (ejes geométricos) [1)]	
	Tipo de dato:	AXIS
	Rango de valores:	Nombres de eje geométrico del canal
`<Factor de acoplamiento>`:	Factor n de la variación angular del eje esclavo para modificar la trayectoria tangente de los ejes maestro: Cambio de ángulo$_{\text{eje esclavo}}$ = Cambio de ángulo$_{\text{trayectoria tangente}}$ * n	
	Tipo de dato:	REAL
	Valor por defecto:	1,0

<Sistema de coordenadas>:	Sistema de coordenadas activo [2]		
	Tipo de dato:	CHAR	
	Valor:	"B":	Sistema de coordenadas básico (valor por defecto)
		"W":	Sistema de coordenadas de pieza (no disponible)
<Optimización>:	Tipo de optimización		
	Tipo de dato:	CHAR	
	Valor:	"S":	Estándar (valor por defecto) La dinámica del eje giratorio no tiene ningún efecto sobre los ejes maestro. Si la dinámica del eje giratorio es superior a la necesitada para realizar el seguimiento, este procedimiento tendrá la precisión suficiente. Si la dinámica del eje giratorio no es lo suficientemente alta como para seguir la modificación de las trayectorias tangentes, la orientación del eje giratorio a lo largo de un recorrido de matado de esquinas no definido diferirá de la orientación de consigna.
		"P":	Se tiene en cuenta la dinámica del eje giratorio al planificar la trayectoria de los ejes maestro. Para ello deben indicarse dos parámetros adicionales al conectar el acoplamiento tangencial con TANGON(): • Recorrido de matado de esquinas • Tolerancia angular Ver el capítulo "Activación de un acoplamiento (TANGON) (Página 976)" **Nota** En relación con las transformadas cinemáticas se recomienda utilizar el tipo de optimización "P".

Nota

Los valores por defecto no necesitan programarse explícitamente.

[1] **Nota**

Como ejes maestro del acoplamiento tangencial deben utilizarse los ejes geométricos que recorren la trayectoria programada en el sistema de coordenadas de máquina (MKS) en relación con la posición básica de la máquina. Por ejemplo, si en una fresadora con cabezal orientable se utiliza el ciclo de orientación CYCLE800, dependiendo de la configuración del ciclo, la interpolación tendrá lugar en el WKS, por ejemplo, con los ejes geométricos X e Y. Sin embargo, el acoplamiento tangencial debe definirse con los ejes geométricos como ejes maestro que recorren la trayectoria programada en el MKS. Para ello deben utilizarse los ejes geométricos como ejes maestro en el estado **previo a la orientación** de la máquina.

[2] **Nota**

El sistema de coordenadas básico (BKS) no debe estar girado en relación con el MKS. Si el BKS se gira, por ejemplo, a través del comando ROT o el ciclo de orientación CYCLE800, el control tangencial dejará de trabajar correctamente.

3.16.7.2 Conexión de la creación de secuencia intermedia (TLIFT)

Si la modificación de tangente del eje esclavo supera el valor límite parametrizado en el dato de máquina MD37400 $MA_EPS_TLIFT_TANG_STEP en una posición de la trayectoria programada de los ejes maestro, el resto de la planificación de trayectoria se realizará en función del comportamiento en esquinas ajustado. Sin utilizar el procedimiento predefinido TLIFT(...), se procede según el comportamiento de matado de esquinas programado en relación con TANG(...) (Página 973) y TANGON(...) (Página 976).

Activación de la creación de secuencia intermedia

Con la programación de TLIFT(...) después de TANG(...), la decodificación previa añade una secuencia intermedia generada automáticamente por el control al reconocer una esquina en esta posición de la trayectoria.

En el procesamiento del programa, los ejes maestro se detienen al alcanzar la secuencia intermedia. En la secuencia intermedia se gira el eje esclavo que posee la dinámica de eje máxima en dirección a la trayectoria tangente de la siguiente secuencia. A continuación, los ejes maestro siguen desplazándose por la trayectoria programada.

Desactivación de la creación de secuencia intermedia

Para desconectar la creación de secuencia intermedia debe volver a definirse el acoplamiento tangencial mediante TANG(...), pero sin conectar después la creación de secuencia intermedia mediante TLIFT(...).

Sintaxis

```
TLIFT (<Eje esclavo>)
```

Descripción

`TLIFT(...)`:	Conexión de la detección de esquinas con cálculo de secuencia intermedia	
`<Eje esclavo>`:	Nombre del eje esclavo (eje giratorio)	
	Tipo de dato:	AXIS
	Rango de valores:	Nombres de eje de canal

Velocidad de giro del eje esclavo

Eje de contorneado

Si el eje esclavo ya se desplazó como eje de contorneado antes de activar el acoplamiento tangencial, el movimiento de giro tiene lugar en la secuencia intermedia como eje de contorneado.

Especificando el radio de referencia con `FGREF[<Eje>]=0.001`, el movimiento de giro se realiza con la velocidad de eje máxima parametrizada:

MD32000 $MA_MAX_AX_VELO[<Eje esclavo>]

Eje de posicionamiento

Si el eje esclavo no se desplazó como eje de contorneado antes de activar el acoplamiento tangencial, el giro tiene lugar en la secuencia intermedia como eje de posicionamiento.

El movimiento de giro se realiza con la velocidad de eje de posicionamiento parametrizada:

MD32060 $MA_POS_AX_VELO[<Eje esclavo>]

3.16.7.3 Activación de un acoplamiento (TANGON)

Mediante el procedimiento predefinido TANGON(...) se activa un acoplamiento tangencial definido previamente con TANG(...) (Página 973). A continuación, el eje esclavo se orienta de forma continua hacia la trayectoria tangente en el siguiente desplazamiento de los ejes maestro.

Ángulo del eje esclavo

El ángulo que adopta el eje esclavo en relación con la trayectoria tangente depende de la relación de transmisión predefinida en TANG(...), del ángulo de decalaje parametrizado en el dato de máquina MD37402 $MA_TANG_OFFSET y del ángulo de decalaje de efecto aditivo predefinido en TANGON(...).

Optimización "P"

Si se indicó el valor "P" como parámetro de optimización en la definición del acoplamiento tangencial (TANG(...)), al activar el acoplamiento deben indicarse el parámetro "Recorrido de matado de esquinas" y opcionalmente el parámetro "Tolerancia angular".

Si se especifica el valor 0 como tolerancia angular, solo tiene efecto el parámetro "Recorrido de matado de esquinas".

Si se especifica un valor mayor que 0 como tolerancia angular, el recorrido de matado de esquinas activo se obtiene a partir del mínimo del recorrido de matado de esquinas parametrizado y del recorrido de matado de esquinas debido a la tolerancia angular parametrizada.

Si el eje esclavo no posee dinámica suficiente para seguir las condiciones parametrizadas, la velocidad de contorneado de los ejes maestro se reduce de manera correspondiente.

Sintaxis

```
TANGON(<Eje esclavo>, <Ángulo de decalaje>, <Recorrido de matado de
esquinas>, <Tolerancia angular>)
```

Descripción

`TANGON(...)`:	Activación de un acoplamiento tangencial	
`<Eje esclavo>`:	Nombre del eje esclavo (eje giratorio)	
	Tipo de dato:	AXIS
	Rango de valores:	Nombres de eje de canal

<Ángulo de decalaje>:	Ángulo de decalaje del eje esclavo respecto a la trayectoria tangente El punto de referencia es el origen del eje giratorio.	
	Tipo de dato:	REAL
<Recorrido de matado de esquinas>:	Recorrido de matado de esquinas máximo admisible Si el recorrido de matado de esquinas aumenta debido a las condiciones de la dinámica, se reduce la velocidad de contorneado de los ejes maestro.	
	Tipo de dato:	REAL
<Tolerancia angular>:	Máxima tolerancia permitida en relación con el ángulo especificado entre la posición de origen del eje esclavo y la trayectoria tangente	
	Tipo de dato:	REAL

3.16.7.4 Desactivación del acoplamiento (TANGOF)

Mediante el procedimiento predefinido TANGOF(...) se desactiva un acoplamiento tangencial definido previamente con TANG(...) (Página 973) y activado con TANGON(...) (Página 976). El eje esclavo ya no se orienta hacia la trayectoria tangente del eje maestro. Sin embargo, el acoplamiento del eje esclavo a los ejes maestro se conserva después de la desconexión, cosa que impide, por ejemplo, las siguientes funciones:

- Cambio de plano
- Conmutación de ejes geométricos
- Definición de un nuevo acoplamiento tangencial para el eje esclavo

El enlace del eje esclavo con los ejes maestro no se elimina por completo hasta que no se borra el acoplamiento mediante TANGDEL(...) (Página 977).

Programación

```
TANGOF (<Eje esclavo>)
```

Descripción

TANGOF(...):	Desconexión de un acoplamiento tangencial	
<Eje esclavo>:	Nombre del eje esclavo (eje giratorio)	
	Tipo de dato:	AXIS
	Rango de valores:	Nombres de eje de canal

3.16.7.5 Borrado de un acoplamiento (TANGDEL)

Un acoplamiento tangencial definido con TANG(...) (Página 973) se conserva aún después de desconectar el acoplamiento tangencial con TANGOF(...) (Página 977). El acoplamiento tangencial existente impide entonces, por ejemplo, las siguientes funciones:

- Cambio de plano
- Conmutación de ejes geométricos
- Definición de un nuevo acoplamiento tangencial para el eje esclavo

El procedimiento predefinido TANGDEL(...) borra el acoplamiento tangencial existente después de su desconexión con TANGOF(...).

Sintaxis

```
TANGDEL(<Eje esclavo>)
```

Descripción

TANGDEL(...):	Borrado de un acoplamiento tangencial definido con TANG()	
	Efecto:	Secuencial
<Eje esclavo>:	Nombre del eje esclavo cuyo acoplamiento tangencial va a borrarse	
	Tipo de dato:	AXIS
	Rango de valores:	Nombres de eje de canal

Ejemplos

Cambio de eje maestro

Antes de poder definir un nuevo acoplamiento tangencial para el eje esclavo con otro eje maestro, debe borrarse el acoplamiento tangencial existente.

Código de programa	Comentarios
N10 **TANG**(A, X, Y, 1)	; Definir el acoplamiento tangencial para el eje esclavo A: A respecto a X e Y
N20 TANGON(A)	; Activar el acoplamiento tangencial para el eje esclavo A
N30 X10 Y20	
...	
N80 TANGOF(A)	; Desactivar el acoplamiento tangencial para el eje esclavo A
N90 **TANGDEL**(A)	; Borrar el acoplamiento tangencial para el eje esclavo A
...	
N120 **TANG**(A, X, Z)	; Definir el nuevo acoplamiento tangencial para el eje esclavo A
N130 TANGON(A)	; Activar el nuevo acoplamiento tangencial para el eje esclavo A
...	

Conmutación de ejes geométricos

Antes de poder conmutar los ejes geométricos para un acoplamiento existente, debe borrarse el acoplamiento.

Código de programa	Comentarios
N10 GEOAX(2, Y1)	; 2.° eje geométrico = eje de máquina Y1

Código de programa	Comentarios
N20 TANG(A, X, Y)	; Definir el acoplamiento tangencial para el eje esclavo A
N30 TANGON(A, 90)	; Activar el acoplamiento tangencial para el eje esclavo A
N40 G2 F8000 X0 Y0 I0 J50	; Secuencia de desplazamiento
N50 TANGOF(A)	; Desactivar el acoplamiento tangencial para el eje esclavo A
N60 **TANGDEL**(A)	; Borrar el acoplamiento tangencial para el eje esclavo A
N70 **GEOAX**(2, Y2)	; 2.° eje geométrico = eje de máquina Y2
N80 TANG(A, X, Y)	; Definir el nuevo acoplamiento tangencial para el eje esclavo A
N90 TANGON(A, 90)	; Activar el nuevo acoplamiento tangencial para el eje esclavo A
...	

3.16.8 Acoplamiento maestro-esclavo (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS)

El "acoplamiento maestro-esclavo" permite:

- acoplar los ejes esclavos al eje maestro solamente cuando los ejes participantes están parados.
- acoplar y separar cabezales con control de la velocidad de giro **que están girando**.
- la configuración dinámica.

Nota

Modo de posicionado

En el caso de los ejes y los cabezales en modo de posicionamiento, el acoplamiento sólo se cerrará y separará en reposo.

Sintaxis

```
MASLON(<Slave_1>,<Slave_2>,...)
MASLOF(<Slave_1>,<Slave_2>,...)
MASLOFS(<Slave_1>,<Slave_2>,...)
```

Configuración dinámica:

```
MASLDEF(<Slave_1>,<Slave_2>,...,<Master>)
MASLDEL(<Slave_1>,<Slave_2>,...)
```

Descripción

MASLON:	Activar un acoplamiento maestro-esclavo temporal	
	<Slave_x>,...:	Eje esclavo 1 ... n
MASLOF:	Separar un acoplamiento de maestro-esclavo activo	
	<Slave_1>,...:	Eje esclavo 1 ... n
MASLOFS:	Separar acoplamiento maestro-esclavo y frenar automáticamente cabezales esclavos (ver nota "Comportamiento de acoplamiento de cabezales en el modo de control de velocidad de giro")	
	<Slave_1>,...:	Eje esclavo 1 ... n
MASLDEF:	Crear/separar conjunto de maestro-esclavo desde el programa de pieza	
	<Slave_1>,...:	Eje esclavo 1 ... n
	<Master>:	Eje maestro
MASLDEL:	Separar acoplamiento maestro/esclavo y borrar definición del conjunto	
	<Slave_1>,...:	Eje esclavo 1 ... n
	Nota: Se conservan las definiciones maestro-esclavo configuradas en los datos de máquina.	

Nota

Comportamiento de acoplamiento de cabezales en el modo de control de velocidad de giro

Con cabezales en el modo de control de velocidad de giro, el comportamiento de acoplamiento de MASLON, MASLOF, MASLOFS y MASLDEL se especifica explícitamente mediante el siguiente dato de máquina:

MD37263 $MA_MS_SPIND_COUPLING_MODE

En el ajuste estándar con MD37263 = 0, el acoplamiento y la separación de los ejes esclavos se realizan únicamente cuando los ejes implicados están en reposo. MASLOFS equivale a MASLOF.

En el caso de MD37263 = 1, la instrucción de acoplamiento es inmediata y, por lo tanto, se ejecuta en movimiento. El acoplamiento se cierra de inmediato con MASLON y se separa de inmediato con MASLOFS o MASLOF. Los cabezales esclavos que giran en este momento con MASLOF mantienen su velocidad de giro hasta que se programa una nueva velocidad de giro. Con MASLOFS, se frenan automáticamente.

Nota

Con MASLOF/MASLOFS se suprime la parada de decodificación previa implícita. Debido a la falta de parada de decodificación previa, las variables de sistema $P no suministran valores actuales para los ejes esclavo hasta el momento de una nueva programación.

Nota

Para el eje esclavo, el valor real puede sincronizarse mediante `PRESETON` al mismo valor que el eje maestro. Para ello, el acoplamiento de maestro-esclavo permanente debe desactivarse por unos instantes para ajustar el valor real del eje esclavo no referenciado con POWER ON en el valor del eje maestro. Después se restablece el acoplamiento permanente.

El acoplamiento de maestro-esclavo permanente se activa con el siguiente ajuste:
MD37262 $MA_MS_COUPLING_ALWAYS_ACTIVE = 1
No tiene efecto alguno sobre las instrucciones de programación del acoplamiento temporal.

Ejemplos

Ejemplo 1: ajuste de valor real con el eje esclavo de un acoplamiento maestro/esclavo

En caso de acoplamiento maestro/esclavo permanente, con PRESETON se ajusta el valor real del eje esclavo al valor del eje maestro.

Código de programa	Comentarios
`$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=0`	`; Desactivar el acoplamiento permanente del eje esclavo`
`NEWCONF`	`; Activar modificación de dato de máquina`
`STOPRE`	
`MASLOF(Y1)`	`; Desactivar el acoplamiento temporal`
`PRESETON(AX2,$VA_IM(M_AX))`	`; Valor real del eje esclavo = valor real del eje maestro`
`$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=1`	`; Activar el acoplamiento permanente del eje esclavo`
`NEWCONF`	`; Activar modificación de dato de máquina`

Ejemplo 2: configuración dinámica de un acoplamiento de maestro-esclavo

Para que, después del giro de contenedor de ejes, se pueda establecer el acoplamiento con otro cabezal, se tiene que separar primero el acoplamiento anterior, borrar la configuración y configurar el nuevo acoplamiento.

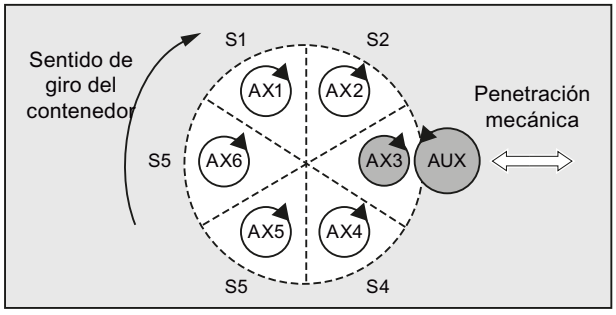


Figura 3-14 Antes del giro del contenedor de ejes

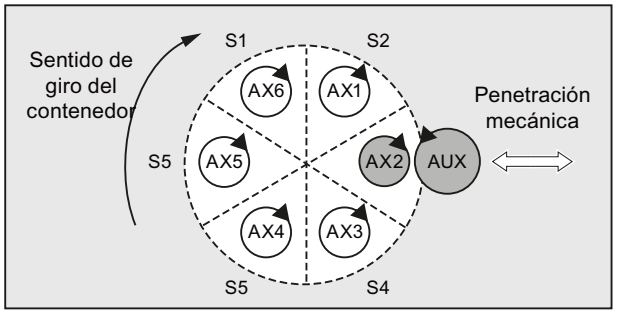


Figura 3-15 Después del giro del contenedor de ejes en una ranura

Código de programa	**Comentarios**
`MASLDEF(AUX,S3)`	`; AUX: Esclavo, S3: Maestro = AX3`
`MASLON(AUX)`	`; Acoplamiento activado`
`M3=3 S3=4000`	`; Girar maestro`
`MASLDEL(AUX)`	`; Separar y suprimir acoplamiento`
`AXCTSWE(CT1)`	`; Habilitar el giro del contenedor de ejes`
`MASLDEF(AUX,S3)`	`; AUX: Esclavo, S3: Maestro = AX2`

Información adicional

- Manual de funciones Tecnologías
- Manual de funciones Funciones básicas

3.17 Acciones síncronas

3.17.1 Definición de una acción síncrona

Una acción síncrona se define en una secuencia de un programa de pieza. En esta secuencia, no deben programarse otros comandos que no formen parte de la acción síncrona.

Una acción síncrona consta de los siguientes componentes:

Ámbito de vigencia; n.º de ident. (opcional)	Parte de condición (opcional)			Parte de acción si se cumple la condición			Parte de acción si no se cumple la condición (opcional)		
	Frecuencia	Comando G (opcional)	Condición	Palabra reservada	Comando G (opcional)	Acciones	Palabra reservada	Comando G (opcional)	Acciones
--- [1] ID=<N.°> IDS=<N.°>	---[1] WHENEVER FROM WHEN EVERY	G...	Expresión lógica	DO	G...	Acción 1 ... Acción n	ELSE	G...	Acción 1 ... Acción n

[1] no programada

Sintaxis

```
DO <Acción 1> ... <Acción n>
<Frecuencia> [<Función G>] <Condición> DO <Acción 1> ... <Acción n>
ID=<N.°> <Frecuencia> [<Función G>] <Condición> DO <Acción 1> ...
<Acción n>
IDS=<N.°> <Frecuencia> [<Función G>] <Condición> DO <Acción 1> ...
<Acción n>
IDS=<N.°> <Frecuencia> [<Función G>] <Condición> DO <Acción 1...n>
ELSE <Acción 1...n>
```

Información adicional

Manual de funciones Acciones síncronas

3.18 Vaivén

3.18.1 Vaivén asíncrono (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB)

Un eje de vaivén u oscilación se desplaza alternativamente entre los dos puntos de inversión del sentido de movimiento 1 y 2 con un determinado avance hasta que se desactiva dicho tipo de movimiento (vaivén u oscilación).

El resto de los ejes se pueden interpolar a discreción mientras se realiza el desplazamiento de vaivén. Mediante contorneado o bien mediante un posicionado por eje se puede conseguir una penetración continua. En este caso, no existe **ninguna relación** entre el movimiento de vaivén y el movimiento de penetración.

Características del vaivén asíncrono

- El vaivén asíncrono actúa de forma específica por eje, superando los límites de secuencias.
- Desde el programa de pieza se asegura la activación sincronizada con la secuencia del movimiento de vaivén.
- No se pueden interpolar varios ejes a la vez ni superponer trayectos de vaivén.

Programación

A través de los siguientes comandos es posible la activación e influenciación del vaivén asíncrono correspondiente a la ejecución del programa CN desde el programa de pieza.

Los valores programados se introducen de forma síncrona al mecanizado principal en el bloque de datos de operador correspondiente y permanecen activos hasta que se vuelven a modificar.

Sintaxis

```
OSP1[<Eje>]=<Valor> OSP2[<Eje>]=<Valor>
OST1[<Eje>]=<Valor> OST2[<Eje>]=<Valor>
FA[<Eje>]=<Valor>
OSCTRL[<Eje>]=(<Opción de activación>,<Opción de desactivación>)
OSNSC[<Eje>]=<Valor>
OSE[<Eje>]=<Valor>
OSB[<Eje>]=<Valor>
OS[<Eje>]=1
OS[<Eje>]=0
```

Descripción

`<Eje>`:	Nombre del eje de vaivén		
`OS`:	Activación/desactivación del vaivén		
	Valor:	1	Activación del vaivén (**Con**)
		0	**Des**activación del vaivén
`OSP1`:	Definición de la posición del punto de inversión 1		

OSP2:	Definición de la posición del punto de inversión 2 **Nota:** Si se ha activado un desplazamiento incremental, la posición se calcula de manera incremental respecto a la última posición de inversión correspondiente programada en el programa CN.		
OST1:	Definición del tiempo de parada en [s] en el punto de inversión 1		
OST2:	Definición del tiempo de parada en [s] en el punto de inversión 2		
	<Valor>:	-2	La interpolación continúa sin esperar la parada exacta fina.
		-1	Espera hasta parada exacta basta.
		0	Espera hasta parada precisa fina.
		>0	Espera hasta parada precisa fina y, a continuación, durante el tiempo de parada indicado **Nota:** La unidad del tiempo de parada es la misma que cuando se programa dicho tiempo mediante G4.
FA:	Definición de la velocidad de avance El valor de la velocidad de avance se define mediante la velocidad de avance definida para el eje de posicionado. Si no se ha definido ningún valor de avance, entonces se toma por defecto el valor introducido en datos de máquina.		
OSCTRL:	Indicación de las opciones de activación y desactivación Los valores optativos 0-3 codifican el comportamiento en los puntos de inversión durante la desactivación. Puede seleccionarse una de las variantes 0-3. El resto de ajustes pueden combinarse a voluntad con la variante seleccionada. Se concatenan varias opciones mediante el carácter más (+)		
	<Valor>:	0	Parada en el siguiente punto de inversión al desactivar al vaivén (ajuste previo) **Nota:** solo es posible reseteando los valores 1 y 2.
		1	Cuando se desactiva el vaivén, parada en el punto de inversión 1.
		2	Cuando se desactiva el vaivén, parada en el punto de inversión 2.
		3	Cuando se desactiva el vaivén, no desplazarse hacia el punto de inversión si no se han programado carreras de afinado.
		4	Aproximación a la posición final tras el afinado
		8	Si se interrumpe el vaivén por medio del borrado de trayecto residual, a continuación deben ejecutarse carreras de afinado y, en caso necesario, efectuar una aproximación a la posición final.
		16	Si se interrumpe el vaivén por medio del borrado de trayecto residual, debe efectuarse una aproximación a la respectiva posición de inversión igual que en la desactivación.
		32	El nuevo valor del avance se activa sólo tras el siguiente punto de inversión.
		64	FA igual a 0, FA = 0: Corrección de la trayectoria activa FA distinto a 0, FA <> 0: Corrección de la velocidad activa
		128	Para eje giratorio DC (por el camino más corto).
		256	La carrera de afinado se ejecuta como carrera doble (por defecto); 1 = la carrera de afinado se ejecuta como carrera individual.
		512	Primero, desplazamiento a la posición inicial
OSNSC:	Definición del número de carreras de afinado		

OSE:	Definición de la posición final (en el WKS) hacia la que debe efectuarse la aproximación tras la desactivación del vaivén **Nota:** al programar "OSE" se activa implícitamente la opción 4 para "OSCTRL".
OSB:	Definición de la posición inicial (en el WKS) hacia la que debe efectuarse la aproximación tras la activación del vaivén La aproximación a la posición inicial se realiza antes del punto de inversión 1. Si la posición inicial coincide con la posición de inversión 1, a continuación se realiza la aproximación a la posición de inversión 2. Al alcanzar la posición inicial no se activa el tiempo de parada, ni siquiera si dicha posición coincide con la posición de inversión 1; en vez de ello, se espera a que se produzca una parada precisa fina. Se cumple una condición de parada precisa configurada. **Nota:** Para que se efectúe la aproximación a la posición inicial debe estar activado el dato de operador SD43770 $SA_OSCILL_CTRL_MASK bit 9.

Ejemplos

Ejemplo 1: el eje de vaivén debe oscilar entre dos puntos de inversión

El eje de vaivén Z debe oscilar entre las posiciones 10 y 100. La aproximación al punto de inversión 1 debe realizarse con parada precisa fina, y al punto de inversión 2, con parada precisa basta. El avance para el eje de vaivén debe ser 250. Al final del mecanizado deben realizarse 3 carreras de afinado y el eje de vaivén debe colocarse en la posición final 200. El avance para el eje de penetración debe ser 1, y el final de la penetración en dirección X debe haberse alcanzado en la posición 15.

Código de programa	**Comentarios**
WAITP(X,Y,Z)	; Posición inicial.
G0 X100 Y100 Z100	; Conmutación al modo Eje de posicionado.
WAITP(X,Z)	
OSP1[Z]=10 OSP2[Z]=100	; Punto de inversión 1, punto de inversión 2.
OSE[Z]=200	; Posición final.
OST1[Z]=0 OST2[Z]=-1	; Tiempo de parada en U1: Parada precisa fina
	; Tiempo de parada en U2: parada exacta basta
FA[Z]=250 FA[X]=1	; Avance del eje de vaivén, avance del eje de penetración.
OSCTRL[Z]=(4,0)	; Opciones de activación.
OSNSC[Z]=3	; 3 carreras de afinado.
OS[Z]=1	; Arranque del vaivén.
WHEN $A_IN[3]==TRUE DO DELDTG(X)	; Borrado de trayecto residual.
POS[X]=15	; Posición inicial del eje X
POS[X]=50	; Posición final del eje X.
OS[Z]=0	; Parada del vaivén.
M30	

Nota

La sucesión de comandos "OSP1[Z]=..." a "OSNCS[Z]=..." también puede programarse en una secuencia.

Ejemplo 2: vaivén con modificación online de la posición de inversión

Los datos de operador necesarios para la definición del movimiento de vaivén asíncrono se pueden ajustar en el programa de pieza.

Si los datos de operador se indican directamente en el programa de pieza, entonces las modificaciones son válidas ya durante el preprocesado. Se puede lograr un comportamiento síncrono mediante una parada de decodificación previa (`STOPRE`).

Código de programa	**Comentarios**
`$SA_OSCILL_REVERSE_POS1[Z]=-10`	
`$SA_OSCILL_REVERSE_POS2[Z]=10`	
`G0 X0 Z0`	
`WAITP(Z)`	
`ID=1 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS1[Z] DO $AA_OVR[X]=0`	`; Cuando el valor real del eje de vaivén sobrepasa el punto de inversión, se detiene el eje de penetración.`
`ID=2 WHENEVER $AA_IM[Z] < $$AA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[X]=0`	
`OS[Z]=1 FA[X]=1000 POS[X]=40`	`; Activación del vaivén.`
`OS[Z]=0`	`; Desactivación del vaivén.`
`M30`	

Información adicional

Eje de vaivén

Para el eje de vaivén rige:

- Cualquier eje puede ser utilizado como eje de vaivén.
- Pueden estar activos varios ejes de vaivén a la vez (máximo: número de ejes de posicionado).
- Para el eje de vaivén siempre está activa la interpolación lineal `G1` independientemente del comando G válido actualmente en el programa.

El eje de vaivén puede:

- Ser un eje de entrada para la transformada dinámica.
- Ser un eje maestro en configuraciones tipo Gantry y ejes arrastrados.
- Desplazarse:
  - sin limitación de tirones "BRISK"
    o bien
  - con limitación de tirones "SOFT"
    o bien
  - con rampa de aceleración por tramos (como ejes de posicionado)

Punto de inversión en vaivén

A la hora de fijar los puntos de inversión del desplazamiento se deben de tener en cuenta los decalajes de origen actualmente activados:

- Introducción en valor absoluto
  "OSP1[Z]=<Valor>"
  Punto de inversión = Suma de los decalajes + valor programado
- Introducción incremental
  "OSP1[Z]=IC(<Valor>)"
  Posición Punto de inversión = Punto de inversión 1 +
  valor programado

Ejemplo:

Código de programa

```
N10 OSP1[Z]=100 OSP2[Z]=110
...
N40 OSP1[Z]=IC(3)
```

WAITP

En el caso de que se realice un vaivén con un eje geométrico, este debe liberarse mediante la función "WAITP".

Cuando finaliza el vaivén, "WAITP" permite transformar de nuevo el eje de vaivén en un eje de posicionado, de forma que este pueda volver a ser utilizado normalmente.

Vaivén con acciones síncronas a desplazamientos y tiempos de parada

Tras expirar los tiempos de parada ajustados, en caso de vaivén se realiza el cambio de secuencia interno (el cual es visible en los nuevos trayectos residuales de los ejes). Al cambiar de secuencia se verifica la función de desactivación. Durante esta operación se determina la función de desactivación según la posición de control ajustada para la secuencia de movimientos (`OSCTRL`). *Este comportamiento en el tiempo se puede influir con la corrección del avance.*

Bajo ciertas circunstancias se ejecuta a continuación una carrera de vaivén antes de iniciar las carreras de afinado o de efectuar el desplazamiento a la posición final. *Parece como si se modificara el comportamiento de desactivación. Sin embargo, no es así.*

3.18.2 Vaivén controlado mediante acciones síncronas (OSCILL)

En este tipo de vaivén solamente se permite realizar una penetración en los puntos de inversión o bien dentro del área definida para la inversión del sentido de desplazamiento.

Según las necesidades, el movimiento de vaivén puede

- continuarse o
- detenerse durante la penetración hasta que ésta se haya ejecutado por completo.

Sintaxis

1. **Definición de los parámetros para el vaivén**
2. **Definición de acciones síncronas a desplazamientos**
3. **Asignación de ejes, definición de la penetración**

Descripción

`OSP1[<eje de vaivén>]=`	Posición del punto de inversión 1
`OSP2[<eje de vaivén>]=`	Posición del punto de inversión 2
`OST1[<eje de vaivén>]=`	Tiempo de parada en el punto de inversión 1 en segundos
`OST2[<eje de vaivén>]=`	Tiempo de parada en el punto de inversión 2 en segundos
`FA[<eje de vaivén>]=`	Avance para el eje de vaivén
`OSCTRL[<eje de vaivén>]=`	Activación/desactivación de las opciones
`OSNSC[<eje de vaivén>]=`	Número de carreras de afinado
`OSE[<eje de vaivén>]=`	Posición final
`WAITP(<eje de vaivén>)`	Liberación del vaivén para el eje

Asignación de ejes, penetración

```
OSCILL[<eje de vaivén>]=(<eje de penetración 1>,<eje de
penetración 2>,<eje de penetración 3>)
POSP[<eje de penetración>]=(<posición final>,<penetración
parcial>,<modo>)
```

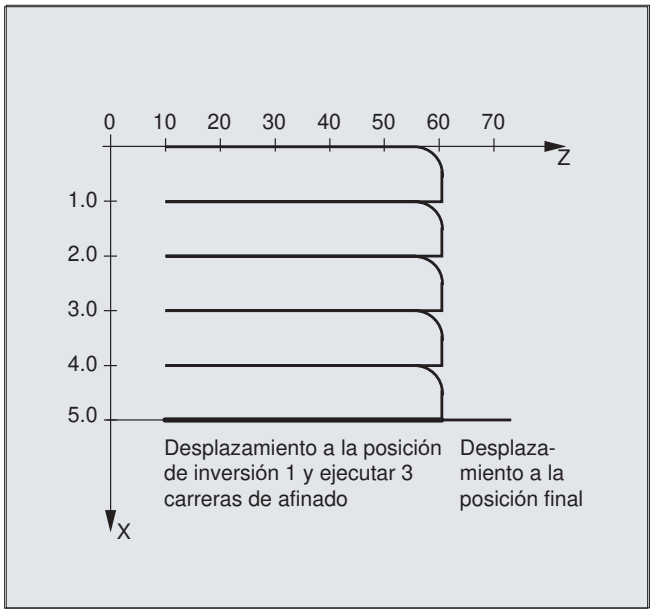
`OSCILL`:	Asignación de los ejes de penetración a los ejes para el vaivén
`POSP`:	Definición de las penetraciones parciales/total (ver cap. Administración de programas y fichero)
`Posición final`:	Posición final para el eje de penetración tras haber realizado todas las penetraciones parciales.
`Penetración parcial`:	Valor de la penetración parcial en el punto de inversión/zona de inversión
`Modo`:	Partición de la penetración total en penetraciones parciales = dos pasos residuales del mismo tamaño (estándar); = mismo tamaño para todas las penetraciones parciales

Acciones síncronas a desplazamiento

`WHEN… … DO`	Si..., entonces...
`WHENEVER … DO`	Siempre que..., entonces...

Ejemplo

En el punto de inversión 1 no se debe realizar penetración. En el punto de inversión 2 se debe realizar la penetración cuando el eje de vaivén se encuentre a una distancia ii2 del punto de inversión 2. El eje de vaivén no deberá esperar en dicho punto a que finalice el desplazamiento de penetración parcial. El eje Z es el eje de vaivén y el eje X el eje de penetración.



1. Parámetros para el vaivén

Código de programa	**Comentarios**
`DEF INT ii2`	`; Definición de la variable para la zona de inversión 2`
`OSP1[Z]=10 OSP2[Z]=60`	`; Definición de los puntos de inversión 1 y 2`
`OST1[Z]=0 OST2[Z]=0`	`; Punto de inversión 1: parada precisa fina`
	`Punto de inversión 2: Parada precisa fina`
`FA[Z]=150 FA[X]=0.5`	`; Avance para el eje de vaivén Z, avance para el eje de penetración X`
`OSCTRL[Z]=(2+8+16,1)`	`; Desactivar el movimiento de vaivén en el punto de inversión 2; afinar (apagado de chispas) tras borrado del trayecto residual y desplazar a posición final; tras borrado del trayecto residual, desplazar a la posición de inversión correspondiente.`
`OSNC[Z]=3`	`; Carreras de afinado`
`OSE[Z]=70`	`; Posición final = 70`
`ii2=2`	`; Definición del valor para la zona de inversión`
`WAITP(Z)`	`; Liberar vaivén para el eje Z`

2. Acciones síncronas a desplazamientos

Código de programa	Comentarios
`WHENEVER $AA_IM[Z]<$SA_OSCILL_REVERSE_POS2[Z] DO ->` `$AA_OVR[X]=0 $AC_META[0]=0`	`; Siempre que la posición actual del eje de vaivén Z en MKS sea más pequeña que el inicio de la zona de inversión 2, fijar la corrección específica del eje de penetración X a 0% y la marca con el índice 0 al valor 0.`
`WHENEVER $AA_IM[Z]>=$SA_OSCILL_REVERSE_POS2[Z] DO $AA_OVR[Z]=0`	`; Siempre que la posición actual del eje de vaivén Z en MKS sea igual de grande que la posición de inversión 2, fijar la corrección específica del eje de vaivén Z a 0%.`
`WHENEVER $AA_DTEPW[X]==0 DO $AC_MARKER[0]=1`	`; Siempre que el trayecto residual de la penetración parcial sea igual a 0, fijar la marca con índice 0 al valor 1.`
`WHENEVER $AC_MARKER[0]==1 DO $AA_OVR[X]=0 $AA_OVR[Z]=100`	`; Siempre que la marca con el índice 0 sea igual a 1, fijar la corrección específica de eje del eje de penetración X a 0%. De esta manera se evita una penetración demasiado temprana (el eje de vaivén Z aún no ha salido de la zona de inversión 2 pero el eje de penetración X ya está preparado para una nueva penetración). Volver a fijar la corrección específica de eje del eje de vaivén Z de 0% (acción de la 2.ª acción síncrona) a 100% para desplazamiento.`

-> se debe programar en una secuencia

3. Iniciar vaivén

Código de programa	Comentarios

`OSCILL[Z]=(X) POSP[X]=(5,1,1)`	`; Iniciar los ejes`
	`El eje X se asigna al eje de vaivén Z como eje de ;penetración.`
	`El eje X debe desplazarse hasta la posición final 5 en pasos de 1.`

`M30`	`; Fin del programa`

Información adicional

1. **Definición de los parámetros para el vaivén**
   Los parámetros para el vaivén se deben definir antes de la secuencia de desplazamiento que contiene la asignación del eje de vaivén al eje de penetración, así como el ajuste de la penetración (ver "Vaivén asíncrono").
2. **Definir acciones síncronas de movimiento**

   A través de condiciones síncronas se realizan las siguientes operaciones:
   Suprimir la penetración hasta que el eje de vaivén se encuentre dentro de una zona de inversión
   (ii1, ii2) o bien en un punto de inversión (U1, U2).
   Parar el movimiento de vaivén durante la penetración en el punto de inversión.
   Iniciar el movimiento de vaivén al finalizar una penetración parcial. Determinar **el inicio de la siguiente penetración parcial**.
3. **Definición de la asignación del eje de vaivén y eje de penetración** así como **de la penetración total y parcial**.

Definición de los parámetros para el vaivén

Asignación del eje de vaivén y del eje de penetración: OSCILL

```
OSCILL[<Eje de vaivén>]=(<Eje de penetración1>,<Eje de
penetración2>,<Eje de penetración3>)
```

El comando "OSCILL" permite definir la asignación de los ejes y el inicio del movimiento de vaivén.

Como máximo se pueden asignar 3 ejes de penetración a un eje de vaivén.

Nota

Antes del inicio de un movimiento de vaivén deben definirse las condiciones síncronas para el comportamiento de los ejes.

Definir penetraciones: POSP

```
POSP[<eje de penetración>]=(<posición final>,<penetración
parcial>,<modo>)
```

El comando "POSP" permite comunicar al control la siguiente información:

- La penetración total (respecto a la posición final)
- El tamaño de la penetración parcial correspondiente en el punto de inversión o bien en la zona de inversión
- El comportamiento de la penetración parcial al alcanzar la posición final (mediante el modo)

Modo = 0	La distancia restante hasta el punto final se divide en dos partes iguales para la penetración parcial (ajuste estándar).
Modo = 1	Las penetraciones parciales se definen del mismo tamaño. Se calculan a partir de la penetración total.

Definir acciones síncronas a desplazamientos

La siguiente lista de acciones síncronas a desplazamientos se utilizan para movimientos de vaivén en general.

Se ofrecen soluciones ejemplares para requisitos individuales que servirán como módulos para la creación de movimientos de vaivén específicos del usuario.

Nota

En casos individuales se pueden programar las condiciones síncronas de forma diferente.

Palabras reservadas

`WHEN … DO …`	si..., entonces...
`WHENEVER … DO`	siempre que..., entonces...

Funciones

Mediante los comandos descritos detalladamente más adelante se pueden realizar las siguientes funciones:

1. Penetración en el punto de inversión.
2. Penetración en la zona de inversión.
3. Penetración en ambos puntos de inversión.
4. Parada del movimiento de vaivén en el punto de inversión.
5. Rearranque del movimiento de vaivén.
6. No arrancar la penetración parcial demasiado pronto.

Para todas las acciones ejemplares anteriormente descritas se supone:

- Punto de inversión 1 < punto de inversión 2
- Z = eje de vaivén
- X = eje de penetración

Nota

Para explicaciones más detalladas, ver el capítulo Acciones síncronas a desplazamientos.

Definición de la asignación del eje de vaivén y eje de penetración así como de la penetración total y parcial.

Penetración en la zona de inversión

El movimiento de penetración se deberá realizar en la zona de inversión antes de que se alcance el punto de inversión.

Esta acción síncrona evita que se realice el movimiento de penetración hasta que el eje de vaivén se encuentre en la zona de inversión.

A partir de los supuestos anteriores (ver arriba) se deducen las siguientes instrucciones:

Zona de inversión 1:

```
WHENEVER $AA_IM[Z]>$SA_OSCILL_RESERVE_POS1[Z]+ii1 DO $AA_OVR[X] = 0
```

Siempre que la posición actual del eje de vaivén en MKS sea más grande que el inicio de la zona de inversión 1, fijar la corrección específica del eje de penetración a 0%.

Zona de inversión 2:

```
WHENEVER $AA_IM[Z]<$SA_OSCILL_RESERVE_POS2[Z]+ii2 DO $AA_OVR[X] = 0
```

Siempre que la posición actual del eje de vaivén en MKS sea más pequeña que el inicio de la zona de inversión 2, fijar la corrección específica del eje de penetración a 0%.

Penetración en el punto de inversión

Hasta que el eje de vaivén no alcance el punto de inversión no se realiza ningún desplazamiento del eje de penetración.

A partir de los supuestos anteriores (ver arriba) se deducen las siguientes instrucciones:

Zona de inversión 1:

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Siempre que la posición actual del eje de vaivén Z en MKS sea más grande o más pequeña que la posición del punto de inversión 1, fijar la corrección específica del eje de penetración X a 0% y la corrección específica del eje de vaivén Z en 100%.

Zona de inversión 2:

Para el punto de inversión 2:

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Siempre que la posición actual del eje de vaivén Zu en MKS sea más grande o más pequeña que la posición del punto de inversión 2, fijar la corrección específica del eje de penetración X a 0% y la corrección específica del eje de vaivén Z en 100%.

Parada del movimiento de vaivén en el punto de inversión

El eje de vaivén se detiene en el punto de inversión y, simultáneamente, comienza el desplazamiento de penetración. El desplazamiento de vaivén continúa cuando haya finalizado completamente el desplazamiento de penetración.

Esta acción síncrona se puede utilizar simultáneamente para arrancar el desplazamiento de penetración, en el caso de que éste hubiera sido detenido por una acción síncrona anterior aún activa.

A partir de los supuestos anteriores (ver arriba) se deducen las siguientes instrucciones:

Zona de inversión 1:

```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Siempre que la posición actual del eje de vaivén en MKS sea igual que la posición de inversión 1, fijar la corrección específica del eje de vaivén a 0% y la corrección específica del eje de penetración en 100%.

Zona de inversión 2:

```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Siempre que la posición actual del eje de vaivén Zu en MKS sea igual que la posición de inversión 2, fijar la corrección específica del eje de vaivén X a 0% y la corrección específica del eje de penetración en 100%.

Evaluación online del punto de inversión

Cuando a la derecha de la comparación se encuentra una variable de proceso principal precedida por los caracteres `$$`, entonces se realiza la evaluación y comparación de las dos variables en cada tiempo de ciclo IPO.

Nota

Para más información al respecto ver capítulo "Acciones síncronas a desplazamientos".

Rearranque del movimiento de vaivén

Esta acción síncrona se utiliza para rearrancar el desplazamiento de vaivén cuando se ha realizado una penetración parcial.

A partir de los supuestos anteriores (ver arriba) se deducen las siguientes instrucciones:

```
WHENEVER $AA_DTEPW[X]==0 DO $AA_OVR[Z] = 100
```

Siempre que el trayecto residual para la penetración parcial del eje de penetración X en WKS sea igual a cero, fijar la corrección específica del eje de vaivén al 100%.

Siguiente penetración parcial

Cuando se ha finalizado la penetración, debe evitarse un arranque prematuro de la siguiente penetración parcial.

Para ello se utiliza la marca específica de canal (`$AC_MARKER[Index]`) que se activa al final de la penetración parcial (trayecto residual ≡ 0) y que se desactiva al abandonar la zona de inversión. Entonces se inhibe el siguiente desplazamiento de penetración mediante una acción síncrona.

Las suposiciones anteriores (ver arriba) dan como resultado, p. ej., las siguientes instrucciones para el punto de inversión 1:

1. Activación de la marca:

```
WHENEVER $AA_DTEPW[X]==0 DO $AC_MARKER[1]=1
```

Siempre que el trayecto residual para la penetración parcial del eje de penetración X en WKS sea igual a cero, fijar la marca con índice 1 a 1.

2. Desactivación de la marca

```
WHENEVER $AA_IM[Z]<> $SA_OSCILL_RESERVE_POS1[Z] DO $AC_MARKER[1] = 0
```

Siempre que la posición actual del eje de vaivén Z en MKS sea más grande o más pequeña que la posición del punto de inversión 1, fijar la marca 1 a 0.

3. Inhibición de la penetración

```
WHENEVER $AC_MARKER[1]==1 DO $AA_OVR[X]=0
```

Siempre que la marca sea igual a 1, fijar la corrección específica del eje de penetración X a 0%.

3.19 Troquelado y punzonado

3.19.1 Activación/desactivación

3.19.1.1 Activar o desactivar punzonado y troquelado (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC)

Activación/desactivación del troquelado o el punzonado

`PON` y `SON` permiten activar la función de troquelado o punzonado. `SPOF` finaliza todas las funciones específicas de troquelado y punzonado. Los comandos modales `PON` y `SON` se excluyen mutuamente, es decir que `PON` desactiva a `SON` y viceversa.

Troquelado/punzonado controlado a nivel de servo

Las funciones `SONS` y `PONS` activan también las funciones de troquelado y punzonado.

Al contrario que el mando de carrera en plano de interpolación activo con `SON`/`PON`, en estas funciones el mando de señal del disparo de carrera se realiza a nivel servo. De este modo, se consiguen mayores frecuencias de punzonado, así como un mayor rendimiento para el troquelado.

Durante la evaluación de las señales en el control a nivel de servo se bloquean todas las funciones que puedan provocar un cambio de posición de los ejes de punzonado o troquelado (p. ej., desplazamiento manual con volante, modificaciones de frames vía PLC, funciones de medida).

Troquelado con retardo

`PDELAYON` genera una salida con retraso del
punzón. El comando modal realiza una preparación
y generalmente se escribe delante de `PON`. Tras `PDELAYOF` se prosigue con el troquelado con normalidad.

Nota

La temporización de retardo se ajusta en el dato de operador
DO42400 $SC_PUNCH_DWELLTIME.

Aceleración dependiente del recorrido

Con `PUNCHACC` puede establecerse una característica de aceleración que defina diferentes aceleraciones en función de la distancia entre agujeros.

Segunda interfaz de troquelado

Las máquinas que deban utilizar alternativamente una segunda interfaz de troquelado (segunda unidad de troquelado o un medio equiparable), pueden cambiar a un segundo par de entradas y salidas digitales rápidas del control (par E/S). En ambas interfaces de troquelado pueden usarse todas las funciones de troquelado/punzonado. La conmutación entre la primera

y la segunda interfaz de troquelado se lleva a cabo por medio de los comandos `SPIF1` y `SPIF2`.

Nota

Requisito: debe haberse definido mediante los datos de máquina un segundo par de E/S para las funciones de troquelado (→ ver datos del fabricante de la máquina).

Sintaxis

```
PON G... X... Y... Z...
SON G... X... Y... Z...
SONS G... X... Y... Z...
PONS G... X... Y... Z...
PDELAYON
PDELAYOF
PUNCHACC(<Smin>,<Amin>,<Smax>,<Amax>)
SPIF1/SPIF2
SPOF
```

Descripción

`PON`:	Activar el troquelado	
`SON`:	Activar el punzonado	
`PONS`:	Activar el troquelado controlado a nivel de servo	
`SONS`:	Activar el punzonado controlado a nivel de servo	
`SPOF`:	Desactivar el troquelado/punzonado	
`PDELAYON`:	Activar el troquelado con retardo	
`PDELAYOF`:	Desactivar el troquelado con retardo	
`PUNCHACC`:	Activar la aceleración dependiente del recorrido Parámetros:	
	`<Smin>`	Menor distancia entre agujeros
	`<Amin>`	Aceleración inicial `<Amin>` puede ser más grande que `<Amax>`.
	`<Smax>`	Mayor distancia entre agujeros
	`<Amax>`	Aceleración final `<Amax>` puede ser mayor que `<Amin>`.
`SPIF1`:	Activar la **primera** interfaz de troquelado El mando de carrera tiene lugar a través del primer par de E/S rápidas.	
`SPIF2`:	Activar la **segunda** interfaz de troquelado El mando de carrera tiene lugar a través del segundo par de E/S rápidas.	
	Nota: Después de un RESET o de un arranque del control siempre está activa la primera interfaz de troquelado. Si sólo se utiliza una interfaz de troquelado, no es necesario programarla.	

Ejemplos

Ejemplo 1: Activar el punzonado

Código de programa	Comentarios
...	
N70 X50 SPOF	; Posicionamiento sin disparo de troquelado.
N80 X100 SON	; Activación del punzonado, disparo de una carrera antes del movimiento (X = 50) y al acabar el movimiento programado (X = 100).
...	

Ejemplo 2: Troquelado con retardo

Código de programa	Comentarios
...	
N170 PDELAYON X100 SPOF	; Posicionamiento sin disparo de troquelado, activación del disparo retardado de troquelado.
N180 X800 PON	; Activación del troquelado. Tras alcanzar la posición final, la carrera de troquelado se realiza con retraso.
N190 PDELAYOF X700	; Desactivación del troquelado con retardo, disparo de troquelado normal al acabar el movimiento programado.
...	

Ejemplo 3: Troquelado con dos interfaces de troquelado

Código de programa	Comentarios
...	
N170 SPIF1 X100 PON	; Al final de la secuencia se produce un disparo de carrera en la primera salida rápida. La señal "Carrera activa" se vigila en la primera entrada.
N180 X800 SPIF2	; El segundo disparo de carrera se produce en la segunda salida rápida. La señal "Carrera activa" se vigila en la segunda entrada.
N190 SPIF1 X700	; El mando de carrera para todas las demás carreras tiene lugar con la primera interfaz.
...	

Información adicional

Punzonado y troquelado controlados a nivel de servo (PONS/SONS)

El troquelado y el punzonado controlados a nivel de servo no se pueden realizar simultáneamente en varios canales. `PONS` y `SONS` sólo pueden activarse en un canal a la vez.

Aceleración dependiente del recorrido (PUNCHACC)

Ejemplo:
`PUNCHACC(2,50,10,100)`

Distancias entre agujeros inferiores a 2mm:

El desplazamiento se efectúa con una aceleración del 50% de la aceleración máxima.

Distancias entre agujeros de 2mm a 10mm:

La aceleración aumenta proporcionalmente a la distancia al 100%.

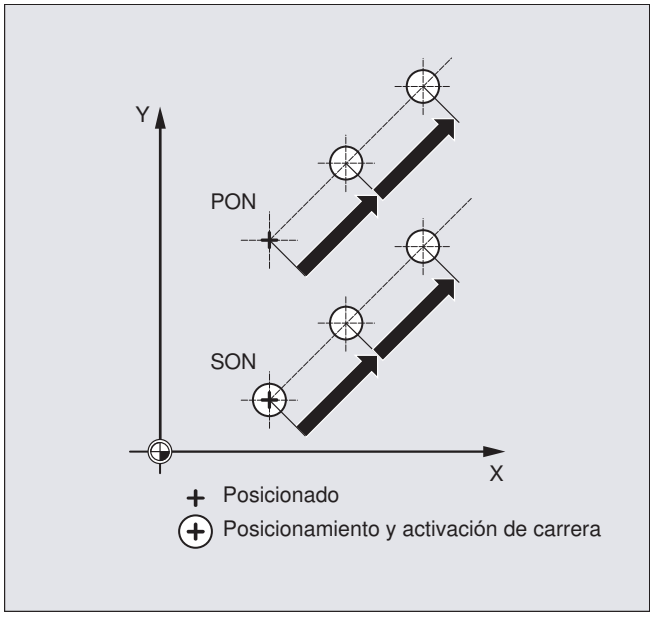
Distancias entre agujeros superiores a 10mm:

Desplazamiento con una aceleración del 100%.

Activación de la primera carrera del punzón

La activación de la primera carrera del punzón tras activar la función se realiza para los procesos de punzonado y troquelado en distinto momento.

- `PON/PONS`:
  - Todas las carreras – también en la primera secuencia después de la activación – se ejecutan al final de la secuencia.
- `SON/SONS`:
  - La primera carrera para el punzón se realiza tras activar el punzonado, ya al principio de la secuencia.
  - El resto de carreras se realizan al final de las secuencias.



Punzonado y troquelado en el sitio

La activación del punzón se realiza sólo cuando la secuencia contenga información sobre el desplazamiento para los ejes de punzonado o troquelado (ejes del plano activo).

Sin embargo, para disparar una carrera en el mismo sitio, debe programarse uno de los ejes de troquelado/punzonado con un recorrido 0.

Trabajos con herramientas giratorias

Nota

Se puede utilizar el control tangencial para conseguir que una herramienta giratoria se posicione de forma tangencial a la trayectoria programada.

Uso de comandos M

Gracias a la técnica de macros sigue siendo posible, como hasta ahora, utilizar funciones M especiales en lugar de instrucciones de programación (compatibilidad). En este caso se aplican las siguientes correspondencias con los sistemas antiguos:

```
M20, M23   ≙  SPOF
M22        ≙  SON
M25        ≙  PON
M26        ≙  PDELAYON
```

Ejemplo de fichero de macros:

Código de programa	**Comentarios**
`DEFINE M25 AS PON`	`; Troquelado activado`
`DEFINE M125 AS PONS`	`; Troquelado controlado a nivel de servo activado`
`DEFINE M22 AS SON`	`; Punzonado activado`
`DEFINE M122 AS SONS`	`; Punzonado controlado a nivel de servo activado`
`DEFINE M26 AS PDELAYON`	`; Troquelado con retardo activado`
`DEFINE M20 AS SPOF`	`; Troquelado, punzonado desactivados`
`DEFINE M23 AS SPOF`	`; Troquelado, punzonado desactivados`

Ejemplo de programación:

Código de programa	**Comentarios**
`...`	
`N100 X100 M20`	`; Posicionamiento sin disparo de troquelado.`
`N110 X120 M22`	`; Activación del punzonado, disparo de carrera antes y después del movimiento.`
`N120 X150 Y150 M25`	`; Activación del troquelado, disparo de carrera al terminar el movimiento.`
`...`	

3.19.2 División automática del recorrido en segmentos

Subdivisión en segmentos

Con el punzonado o el troquelado activados, SPP y también SPN ocasionan una división del desplazamiento total programado para los ejes de trayectoria en una cantidad de segmentos de igual longitud (división equidistante de la trayectoria). De manera interna, cada segmento corresponde a una secuencia.

Cantidad de carreras

En troquelado, la primera carrera del punzón se realiza al final del primer segmento, al contrario que en punzonado, donde la primera carrera del punzón se realiza al inicio del primer segmento. A lo largo de todo el desplazamiento se obtienen las siguientes cifras:

Troquelado: Cantidad de carreras = Cantidad de segmentos

Punzonado: Cantidad de carreras = Cantidad de segmentos + 1

funciones auxiliares

Las funciones auxiliares se ejecutan en la primera de las secuencias generadas.

Sintaxis

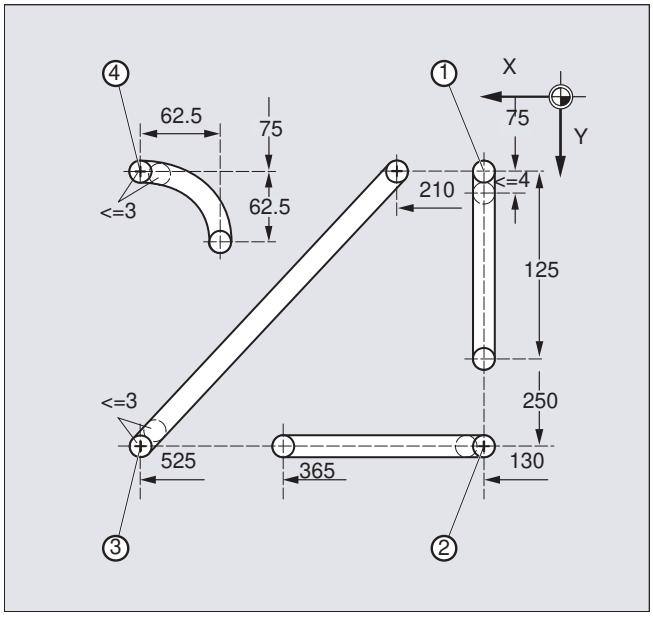
```
SPP=
SPN=
```

Descripción

SPP:	Tamaño de los segmentos (distancia máxima entre punzonados), válido de forma modal
SPN:	Cantidad de segmentos por secuencia; efectivos secuencia a secuencia

Ejemplo 1

Los segmentos de punzonado programados deben dividirse automáticamente en segmentos de igual tamaño.

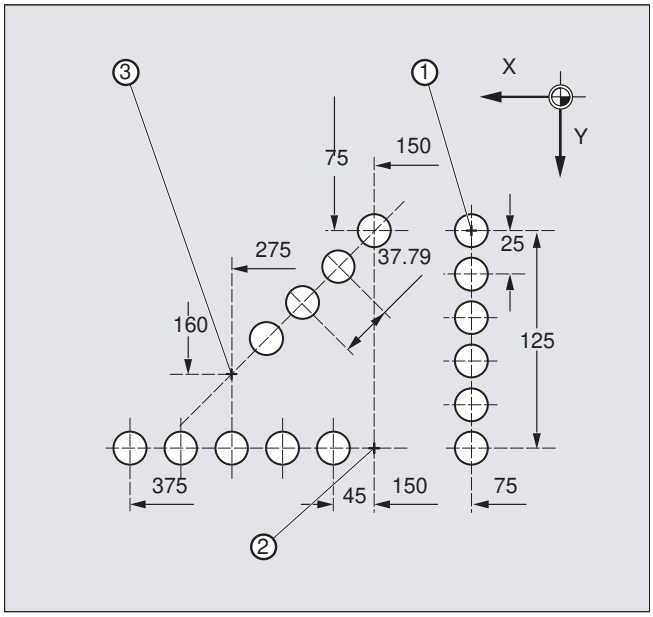


Código de programa	**Comentarios**
N100 G90 X130 Y75 F60 SPOF	; Posicionar en el punto de inicio 1
N110 G91 Y125 SPP=4 SON	; Punzonado activado; longitud de segmento máxima para división automática de la trayectoria: 4 mm
N120 G90 Y250 SPOF	; Punzonado desactivado; posicionado en el punto inicial 2
N130 X365 SON	; Punzonado activado; longitud de segmento máxima para división automática de la trayectoria: 4 mm
N140 X525 SPOF	; Punzonado desactivado; posicionado en el punto inicial 3
N150 X210 Y75 SPP=3 SON	; Punzonado activado; longitud de segmento máxima para división automática de la trayectoria: 3 mm
N160 X525 SPOF	; Punzonado desactivado; posicionado en el punto inicial 4
N170 G02 X-62.5 Y62.5 I J62.5 SPP=3 SON	; Punzonado activado; longitud de segmento máxima para división automática de la trayectoria: 3 mm

Código de programa	Comentarios
N180 G00 G90 Y300 SPOF	; Punzonado desactivado

Ejemplo 2

Para las filas de agujeros individuales debe realizarse una división automática de la trayectoria Para la división se indica respectivamente la longitud de segmento máxima (valor SPP).



Código de programa	Comentarios
N100 G90 X75 Y75 F60 PON	; Posicionado en el punto inicial 1; troquelado de agujero individual activado
N110 G91 Y125 SPP=25	; Longitud de segmento máxima para segmentación automática de la trayectoria: 25 mm
N120 G90 X150 SPOF	; Troquelado desactivado; posicionado en el punto inicial 2
N130 X375 SPP=45 PON	; Troquelado activado; longitud de segmento máxima para división automática de la trayectoria: 45 mm
N140 X275 Y160 SPOF	; Troquelado desactivado; posicionado en el punto inicial 3
N150 X150 Y75 SPP=40 PON	; Troquelado activado; en lugar de la longitud de segmento programada de 40 mm ; se utiliza la longitud de segmento calculada de 37,79 mm.
N160 G00 Y300 SPOF	; Troquelado desactivado; posicionado

3.19.2.1 División en ejes de contorneado

Longitud de los segmentos SPP

Con `SPP` se indica la distancia máxima entre punzonados y, en consecuencia, la longitud máxima de los segmentos en los cuales se dividirá el recorrido total. La desactivación de la función se realiza mediante `SPOF` o bien `SPP=0`.

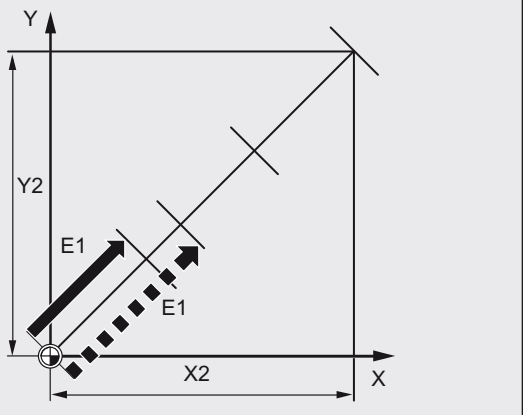
Ejemplo:

```
N10 SON X0 Y0
N20 SPP=2 X10
```

El recorrido total de 10 mm se divide en 5 segmentos de 2 mm cada uno(`SPP=2`).

Nota

La división en segmentos con `SPP` se realiza siempre de forma equidistante: todos los segmentos tienen la misma longitud. Es decir, solamente es válido el valor de longitud programado para los segmentos cuando el cociente entre el trayecto total programado y el valor `SPP` es un número entero. En caso contrario se reduce internamente el número de segmentos, de manera que dicho cociente resulte un número entero.



X2/Y2 Recorrido programado (secuencia de punzonado o troquelado)
E1 Longitud de segmento programada
E1' Longitud de segmento redondeada automáticamente

Ejemplo:

```
N10 G1 G91 SON X10 Y10
N20 SPP=3.5 X15 Y15
```

En un trayecto de 15 mm con una longitud de segmentos de 3,5 mm, el cociente es de (4,28). Como consecuencia se realiza una reducción de forma automática del valor SPP hasta el cociente de un número entero. En este caso, se obtiene una longitud para los segmentos de 3 mm.

Número de segmentos SPN

Con la función `SPP` se define el número de segmentos en los que se desea dividir la trayectoria programada. La longitud de los segmentos se calcula de forma automática. La función `SPP` es válida de forma secuencial, por lo cual se debe activar antes de troquelar o punzonar con `PON` o `SON` N.

3.19.2.2 División en ejes individuales

Si se han definido como ejes de punzonado/troquelado ejes individuales adicionalmente a los ejes de la trayectoria, también se puede utilizar para éstos la división automática en segmentos.

Comportamiento de un eje individual con SPP

La longitud programada del segmento (`SPP`) se refiere básicamente a los ejes de contorneado. Por esta razón, el valor SPP se ignora en las secuencias en las que no se ha programado un eje de contorneado además del desplazamiento de un eje individual y el valor SPP.

Si se ha programado un eje individual y un eje de contorneado en la secuencia, el comportamiento del eje individual dependerá del ajuste del dato de máquina correspondiente.

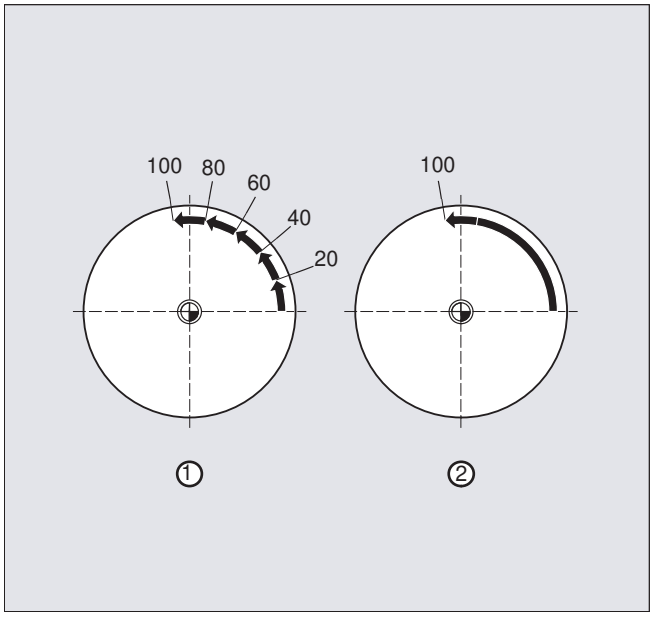
1. Ajuste estándar
   El recorrido del eje individual se distribuye uniformemente en las secuencias intermedias generadas mediante `SPP`.

Ejemplo:

```
N10 G1 SON X10 A0
N20 SPP=3 X25 A100
```

Debido a que los segmentos se han definido de 3 mm, se realizarán en total 5 secuencias a lo largo del desplazamiento total de 15 mm en el eje X (eje de trayectoria).

El eje A realiza así un giro de 20° por secuencia.



1. Eje individual sin división de la trayectoria
   El eje individual recorre la trayectoria completa en la primera de las secuencias generadas.
2. División diferente de la trayectoria
   El comportamiento del eje individual depende de la interpolación de los ejes de contorneado:

- Interpolación circular: con división de la trayectoria
- Interpolación lineal: sin división de la trayectoria

Comportamiento con SPN

El número programado de segmentos se aplica aunque no se haya programado también un eje de contorneado.

Requisitos: eje individual definido como eje de troquelado y punzonado.

3.20 Rectificado

3.20.1 Activación/desactivación de la vigilancia específica de muelas (TMON, TMOF)

Los procedimientos predefinidos TMON(...) y TMOF(...) activan o desactivan la vigilancia específica de muelas (vigilancia de geometría y de velocidad).

Requisitos

Los parámetros específicos de herramienta $TC_TPG1 a $TC_TPG9 deben estar fijados.

Sintaxis

```
TMON(<TNr>)
...
TMOF(<TNr>)
```

Descripción

`TMON(...)`:	Activación de la vigilancia específica de muelas El comando debe programarse en el canal en el cual se desea activar la vigilancia específica de muelas.
`TMOF(...)`:	Desactivar la vigilancia específica de muelas El comando debe programarse en el canal en el cual se desea desactivar la vigilancia específica de muelas.
`<TNr>`:	Número T **Nota:** La indicación solo es necesaria si la vigilancia debe activarse o desactivarse para una muela no activa en vez de para la que se encuentra activa y en uso.
`TMOF(0)`:	Desactivar vigilancia para todas las herramientas

3.21 Parada y retirada ampliada (ESR)

La función "Parada y retirada ampliada", en lo sucesivo "ESR", favorece reacciones flexibles, adaptadas al proceso, en caso de fallo:

- **Parada ampliada**
  Si la situación de error lo permite, se paran en orden todos los ejes habilitados para la parada ampliada.
- **Retirada**
  La herramienta en funcionamiento se retira con la máxima rapidez de la pieza.
- **Modo generador (función de accionamiento SINAMICS "Regulación de Vdc")**
  Si la tensión del circuito intermedio baja de un valor parametrizable, p. ej., si falla la tensión de red, la energía eléctrica necesaria para la retirada se obtiene mediante la realimentación de la energía de frenado de un accionamiento previsto para este fin (modo generador).

Fuentes de disparo

Fuentes generales (externa al CN/global o específico de BAG/de canal)

- Entradas digitales (p. ej., en módulo NCU) o la imagen de proceso interna del control, legible, de entradas digitales ($A_IN, $A_OUT)
- Estado del canal ($AC_STAT)
- Señales VDI ($A_DBB)
- Avisos colectivos de varias alarmas ($AC_ALARM_STAT)

Fuentes de eje

- Umbral de retirada de emergencia del eje esclavo (marcha síncrona del acoplamiento electrónico, $VC_EG_SYNCDIFF[<Eje esclavo>])
- Accionamiento: umbral de alarma del circuito intermedio (riesgo de baja tensión), $AA_ESR_STAT[<Eje>]
- Accionamiento: umbral de velocidad de giro mínima del generador (no queda energía de rotación realimentable), $AA_ESR_STAT[<Eje>].

Lógica combinacional de las acciones síncronas estáticas: combinación lógica de fuente/reacción

Las posibilidades de combinación flexibles de las acciones síncronas estáticas se aprovechan para activar determinadas reacciones relativamente inmediatas debido a las fuentes.

La combinación lógica de todas las fuentes relevantes con la asistencia de las acciones síncronas estáticas está en manos del usuario. Puede evaluar las variables de sistema de la fuente de forma global o selectiva (utilizando máscaras de bits) y combinarlas lógicamente con las reacciones elegidas. Las acciones síncronas estáticas tienen efecto en todos los modos de operación.

Activación

Habilitación de funciones

Las funciones modo generador, parada y retirada se habilitan mediante la activación de la señal de control $AA_ESR_ENABLE correspondiente. Las acciones síncronas pueden modificar esta señal de control.

Activación de funciones

ESR se activa conjuntamente para todos los ejes habilitados mediante activación de las variables de sistema $AC_ESR_TRIGGER.

El modo generador se activa "automáticamente" en el accionamiento si existe riesgo de baja tensión del circuito intermedio.

La función de parada y/o retirada integrada en el propio accionamiento se activa si se detecta un fallo de la comunicación (entre CN y el accionamiento) o una baja tensión del circuito intermedio del accionamiento (previa configuración y habilitación).

La función de parada y/o retirada integrada en el propio accionamiento pueden activarse también desde el CN mediante la activación de la señal de control $AN_ESR_TRIGGER correspondiente (comando de difusión a todos los accionamientos).

Información adicional

Manual de funciones Ejes y cabezales

3.21.1 ESR controlada por CN

3.21.1.1 Retirada controlada por CN (POLF, POLFA, POLFMASK, POLFMLIN)

Para la retirada controlada por CN, se precisan determinadas condiciones básicas (ver "Retirada controlada por CN (POLF, POLFA, POLFMASK, POLFMLIN) (Página 1010)"). Si se cumplen estas condiciones básicas, se activa la retirada rápida (LIFTFAST) de los ejes de retirada configurados en el canal mediante la activación de las variables de sistema $AC_ESR_TRIGGER (o $AA_ESR_TRIGGER para ejes individuales).

Sintaxis

```
POLF(<Eje>)=<Posición>
POLFA(<Eje>,<Tipo>,<Posición>)
POLFMASK(<Eje_1>,<Eje_2>,...)
POLFMLIN(<Eje_1>,<Eje_2>,...)
```

Para `POLFA` se admiten las siguientes formas abreviadas:

```
POLFA(<Eje>,<Tipo>; forma abreviada para retirada de ejes
individuales
POLFA(Eje,0/1/2); desactivación o activación rápida
POLFA(Eje,0,$AA_POLFA[Eje]); provoca una parada de decodificación
previa
```

```
POLFA(Eje,0); no provoca parada de decodificación previa
```

Descripción

POLF:	Dirección para indicar la posición de destino del eje de retirada POLF es modal.			
	<Eje>:	Nombre del eje geométrico o eje de canal/máquina que se retira.		
	<Posición>:	Posición de retirada		
		Tipo:	REAL	
		Para el eje geométrico se aplica el WKS, y en los demás casos el MKS. Si se tienen los mismos identificadores para el eje geométrico y el eje de canal/máquina, se efectúa la retirada en el WKS.		
POLFA:	Llamada predefinida a un subprograma para indicar la posición de retirada de ejes individuales			
	<Eje>:	Identificador de eje de canal		
	<Tipo>:	Modo de indicación de posición		
		Tipo:	INT	
		Valor:	0:	Marcar valor de posición como no válido
			1:	Valor de posición absoluto
			2:	Valor de posición incremental (distancia)
		Nota: Si el eje no es un eje individual o si falta el tipo o se aplica tipo=0, se emite la correspondiente alarma.		
	<Posición>:	Posición de retirada (ver arriba)		
		Nota: El valor de posición también se adopta con tipo=0. Entonces, solo este valor se marca como no válido y se tiene que volver a programar para la retirada.		
POLFMASK:	Llamada predefinida a un subprograma para seleccionar los ejes que deberán retirarse **independientemente** después de activarse la retirada rápida.			
	<Eje_1>,...:	Nombre de los ejes que deben desplazarse a sus posiciones definidas mediante POLF en la retirada rápida. Todos los ejes especificados deben encontrarse en el mismo sistema de coordenadas.		
	El comando POLFMASK() sin indicación de un eje desactiva la retirada rápida para todos los ejes que se han retirado independientemente unos de otros.			
POLFMLIN:	Llamada predefinida a un subprograma para seleccionar los ejes que deberán retirarse **en relación lineal** después de activarse la retirada rápida.			
	<Eje_1>,...:	ver arriba		
	El comando POLFMLIN() sin indicación de un eje desactiva la retirada rápida para todos los ejes que se han retirado en relación lineal.			

Nota

Para poder activar la retirada rápida a una posición fija por medio de `POLFMASK` o `POLFMLIN` es imprescindible programar previamente con `POLF` una posición para los ejes seleccionados.

Nota

Si los ejes se habilitan sucesivamente con `POLFMASK`, `POLFMLIN` o `POLFMLIN`, `POLFMASK`, siempre se aplica la última definición para el eje en cuestión.

Nota

Las posiciones programadas con `POLF` y la activación con `POLFMASK` o `POLFMLIN` se borran durante el inicio del programa de pieza. Esto significa que el usuario debe programar de nuevo en cada programa de pieza los valores para `POLF` y los ejes seleccionados en `POLFMASK` o `POLFMLIN`.

Nota

Si, al utilizar las formas abreviadas `POLFA`, se cambia únicamente el tipo, el usuario debe asegurarse de que la posición o la trayectoria de retirada contienen un valor adecuado. En particular, la posición de retirada y la trayectoria de retirada se tienen que volver a fijar después de POWER ON.

Ejemplo

Retirada de un eje individual:

Código de programa	**Comentarios**
`MD37500 $MA_ESR_REACTION[AX1]=21`	`; Retirada controlada por CN.`
`...`	
`$AA_ESR_ENABLE[AX1]=1`	
`POLFA(AX1,1,20.0)`	`; A AX1 se le asigna la posición de retirada por eje 20.0 (absoluta).`
`$AA_ESR_TRIGGER[AX1]=1`	`; A partir de aquí empieza la retirada.`

Información adicional

Requisitos de la retirada controlada por CN

- En el canal se ha configurado un eje para la retirada controlada por CN:
  MD37500 $MA_ESR_REACTION = 21
- ESR ha de estar habilitada para este eje:
  $AA_ESR_ENABLE = 1
- Se han definido tiempos de retardo:
  MD21380 $MC_ESR_DELAY_TIME1
  MD21381 $MC_ESR_DELAY_TIME2

- En el programa de pieza se han programado las posiciones de retirada específicas de eje con POLF.
- Los ejes se han seleccionado con POLFMASK/POLFMLIN para la retirada controlada por CN.
- Para el movimiento de retirada, las señales de habilitación o desbloqueo tienen que estar activadas y permanecer así.

Liberar e iniciar retirada controlada por CN

Si se activa la variable de sistema $AC_ESR_TRIGGER = 1, se ha configurado un eje de retirada en este canal (es decir, MD37500 $MA_ESR_REACTION = 21) y se ha activado $AA_ESR_ENABLE = 1 para este eje, se activa la retirada rápida (LIFTFAST) en este canal.

El movimiento de retirada del/de los eje(s) seleccionado(s) con POLFMASK o POLFMLIN, configurado con POLF o LFPOS, sustituye el desplazamiento de trayectoria establecido para dicho(s) eje(s) en el programa de pieza.

Para la retirada se dispone como máximo de la suma de los tiempos MD21380 $MC_ESR_DELAY_TIME1 y MD21381 $MC_ESR_DELAY_TIME2. Una vez transcurrido dicho tiempo, se inicia una deceleración rápida también para el eje de retirada, con corrección posterior,

Nota

La retirada ampliada (es decir, LIFTFAST/LFPOS activado por $AC_ESR_TRIGGER) **no se puede interrumpir** y solo se puede cancelar prematuramente mediante una parada de emergencia.

Nota

La retirada disparada mediante $AC_ESR_TRIGGER está bloqueada contra retirada múltiple.

Retirada de ejes individuales

En la retirada de ejes individuales con POLFA se debe haber programado la posición de retirada del eje individual y se deben cumplir las siguientes condiciones:

- $AA_ESR_ENABLE = 1
- <Eje> ha de ser un eje individual en el momento de la conmutación ($AA_ESR_TRIGGER = 1).
- <Tipo> ha de ser 1 ó 2.

Dirección de retirada durante la retirada rápida

En el momento de la activación de la retirada rápida se considera el frame válido.

Nota

Los frames con giro determinan también la dirección de retirada mediante POLF.

Intercambio de eje

Los ejes de retirada se deben asignar siempre de forma precisa a un canal CN y no se deben intercambiar entre canales. Al intentar cambiar un eje de retirada a otro canal, se notifica una

alarma. Solo después de volver a desactivar ese eje mediante $AA_ESR_ENABLE[AX] = 0 es posible cambiar el eje a un canal nuevo. Una vez realizado el intercambio, los ejes se pueden volver a habilitar con $AA_ESR_ENABLE[AX] = 1.

Ejes neutros

Los ejes neutros no pueden ejecutar ninguna ESR controlada por CN.

3.21.1.2 Parada controlada por CN

Mediante la activación de las variables de sistema $AC_ESR_TRIGGER (o $AA_ESR_TRIGGER para ejes individuales), se activa la parada controlada por CN de los ejes de parada configurados en el canal.

Requisitos

- En el canal se ha configurado un eje para la parada controlada por CN:
  MD37500 $MA_ESR_REACTION = 22
- ESR ha de estar habilitada para este eje:
  $AA_ESR_ENABLE = 1
- Se han definido tiempos de retardo:
  MD21380 $MC_ESR_DELAY_TIME1 (tiempo de retardo ejes ESR)
  MD21381 $MC_ESR_DELAY_TIME2 (tiempo ESR para frenado con interpolación)

Proceso

Durante el intervalo de tiempo en DM21380, el eje interpola sin inconvenientes según la programación. Tras el fin del intervalo de tiempo en DM21380 se introduce el frenado con interpolación (parada con rampa). Para el frenado con interpolación, se dispone como máximo del intervalo de tiempo en MD21381. Una vez transcurrido dicho tiempo, se inicia una deceleración rápida con corrección posterior.

Ejemplo

Parada de un eje individual:

Código de programa	**Comentarios**
`MD37500 $MC_ESR_REACTION[AX1] = 22`	`; Parada controlada por CN.`
`MD21380 $MC_ESR_DELAY_TIME1[AX1] = 0.3`	
`MD21381 $MC_ESR_DELAY_TIME2[AX1] = 0.06`	
`...`	
`$AA_ESR_ENABLE[AX1]=1`	
`$AA_ESR_TRIGGER[AX1]=1`	`; A partir de aquí empieza la parada.`

3.21.2 ESR integrada en el accionamiento

3.21.2.1 Configurar la parada ejecutada en el propio accionamiento (ESRS)

La función `ESRS(...)` permite configurar los parámetros de accionamiento para la "parada" de la función ESR ejecutada en el propio accionamiento.

Sintaxis

```
ESRS(<Eje_1>,<Tiempo de parada_1>[,...,<Eje_n>,<Tiempo de parada_n>])
```

Descripción

`ESRS(...)`:	Función para escribir los parámetros de accionamiento para la función ESR "Parada" La función: • debe encontrarse sola en la secuencia. • dispara una parada de decodificación previa. • no puede utilizarse en acciones síncronas.	
`<Eje_1>`, `...`, `<Eje_n>`:	Eje para el que debe configurarse la parada ejecutada en el propio accionamiento En el accionamiento se escribe el parámetro de accionamiento p0888 (configuración) para este eje: p0888 = 1	
	Tipo:	AXIS
	Rango de valores:	Identificador de eje de canal
`<Tiempo de parada_1>`, `...`, `<Tiempo de parada_n>`:	Periodo en el que el accionamiento se desplaza de manera constante con la consigna de velocidad actual después de producirse un error En el accionamiento se escribe el parámetro de accionamiento p0892 (temporización) para el eje indicado: p0892 = <Tiempo de parada>	
	Unidad:	s
	Tipo:	REAL
	Rango de valores:	0.00 - 20.00
En una llamada a una función pueden programarse como máximo 5 ejes; n = 5		

3.21.2.2 Configurar la retirada ejecutada en el propio accionamiento (ESRS)

La función `ESRR(...)` permite configurar los parámetros de accionamiento para la "retirada" de la función ESR integrada en el accionamiento.

Sintaxis

```
ESRR(<Eje_1>,<Trayecto de retirada_1>,<Velocidad de
retirada_1>[,...,<Eje_n>,<Trayecto de retirada_n>,<Velocidad de
retirada_n>])
```

Descripción

`ESRR(...)`:	Función para escribir los parámetros de accionamiento para la función ESR "Retirada" La función: • debe encontrarse sola en la secuencia. • dispara una parada de decodificación previa. • no puede utilizarse en acciones síncronas.	

`<Eje_1>,`
`...,`
`<Eje_n>`:
	Eje para el que debe configurarse la retirada ejecutada en el propio accionamiento
	En el accionamiento se escribe el parámetro de accionamiento p0888 (configuración) para este eje:
	p0888 = 2	

	Tipo:	AXIS
	Rango de valores:	Identificador de eje de canal

`<Trayecto de retirada_1>,`
`...,`
`<Trayecto de retirada_n>`:
	El trayecto de retirada se convierte a una velocidad de retirada para el accionamiento. El valor se escribe en el parámetro de accionamiento p0893 (velocidad) para el eje indicado:
	p0893 = (<Trayecto de retirada_n> convertido a velocidad de retirada)	

	Unidad:	mm/min, pulgadas/min, grados/min (en función de la unidad del eje)
	Tipo:	REAL
	Rango de valores:	MÍN - MÁX

`<Velocidad de retirada_1>,`
`...,`
`<Velocidad de retirada_n>`:
	La velocidad de retirada se convierte a una duración para el accionamiento. El valor se escribe en el parámetro de accionamiento p0892 (temporización) [s] para el eje indicado:
	p0892 = <Trayecto de retirada_n>/<Velocidad de retirada_n>	

	Unidad:	mm/min, pulgadas/min, grados/min (en función de la unidad del eje)
	Tipo:	REAL
	Rango de valores:	0,00 - MÁX
En una llamada a una función pueden programarse como máximo 5 ejes; n = 5		

3.22 Tiempo de ejecución de programa/contador de piezas

Para ayudar al operador de máquinas herramienta, se proporciona información sobre el tiempo de ejecución del programa y la cantidad de piezas.

Esta información puede procesarse como variables del sistema en el programa CN o PLC. Al mismo tiempo, esta información está disponible para la visualización en la interfaz de usuario.

3.22.1 Tiempo de ejecución del programa

La función "Tiempo de ejecución del programa" proporciona temporizadores internos de CN para la vigilancia de procesos tecnológicos que pueden leerse mediante variables del sistema específicas de canal y de CN en el programa de pieza y en acciones síncronas.

El disparador de la medición de tiempo de ejecución ($AC_PROG_NET_TIME_TRIGGER) es la única variable del sistema de la función que puede escribirse y sirve para la medición selectiva de partes del programa. Es decir, mediante la descripción del disparador en el programa CN, la medición de tiempo puede activarse y desactivarse.

Variable del sistema	Descripción	Actividad
Específico de CN		
$AN_SETUP_TIME	Tiempo en minutos desde el último arranque del control con valores estándar ("arranque en frío") Se resetea automáticamente a "0" con cada arranque del control con valores estándar.	• siempre activo
$AN_POWER ON_TIME	Tiempo en minutos desde el último arranque normal del control ("arranque en caliente") Se resetea automáticamente a "0" con cada arranque normal del control.	
Específico de canal		
$AC_OPERATING_TIME	Tiempo de ejecución total en segundos de programas CN en el modo automático El valor se resetea automáticamente a "0" con cada arranque del control.	• Activación con MD27860 • sólo modo de operación AUTOMÁTICO
$AC_CYCLE_TIME	Tiempo de ejecución en segundos del programa CN seleccionado El valor se resetea automáticamente a "0" con el inicio de un nuevo programa CN.	
$AC_CUTTING_TIME	Tiempo de mecanizado en segundos Se mide el tiempo de ejecución de los ejes de contorneado (al menos uno debe estar activo) sin activar el desplazamiento en rápido en todos los programas CN entre Marcha CN y Fin de programa/Reset CN. La medición se interrumpe adicionalmente cuando está activo el tiempo de espera. El valor se resetea automáticamente a "0" con cada arranque del control con valores estándar.	

Variable del sistema	Descripción		Actividad
$AC_ACT_PROG_NET_TIME	Tiempo de ejecución neto actual en segundos del programa CN actual Se resetea automáticamente a "0" con el inicio de un programa CN.		• siempre activo • sólo modo de operación AUTOMÁTICO
$AC_OLD_PROG_NET_TIME	Tiempo neto de ejecución del programa finalizado correctamente con M30 en segundos		
$AC_OLD_PROG_NET_TIME_COUNT	Modificaciones en $AC_OLD_PROG_NET_TIME Tras POWER ON se encuentra $AC_OLD_PROG_NET_TIME_COUNT en "0". $AC_OLD_PROG_NET_TIME_COUNT se incrementa siempre que el control $AC_OLD_PROG_NET_TIME se escribe de nuevo.		
$AC_PROG_NET_TIME_TRIGGER	Disparador para la medición de tiempo de ejecución:		• sólo modo de operación AUTOMÁTICO
	0	Estado neutro El disparador no está activo.	
	1	Salir Finaliza la medición y copia el valor de $AC_ACT_PROG_NET_TIME en $AC_OLD_PROG_NET_TIME. $AC_ACT_PROG_NET_TIME se pone a "0" y luego continúa.	
	2	Inicio Inicia la medición y pone $AC_ACT_PROG_NET_TIME a "0". $AC_OLD_PROG_NET_TIME no se modifica.	
	3	Parada Para la medición. No modifica $AC_OLD_PROG_NET_TIME y mantiene $AC_ACT_PROG_NET_TIME constante hasta el momento de proseguir.	
	4	Continuar Continuación de la medición, es decir, se reanuda una medición parada anteriormente. $AC_ACT_PROG_NET_TIME continúa. $AC_OLD_PROG_NET_TIME no se modifica.	
¡Mediante POWER ON, todas las variables del sistema vuelven a "0"!			

Nota

Fabricante de la máquina

La conexión de los temporizadores activables se realiza mediante el dato de máquina MD27860 $MC_PROCESSTIMER_MODE.

El comportamiento de las mediciones de tiempo activas en ciertas funciones (p. ej. GOTOS, corrección = 0%, avance de recorrido de prueba activo, test de programa, ASUP, PROG_EVENT…) se configura mediante los datos de máquina MD27850 $MC_PROG_NET_TIMER_MODE y MD27860 $MC_PROCESSTIMER_MODE.

Información adicional: Manual de funciones Funciones básicas

Nota

Tiempo residual de una pieza

Si se producen las mismas piezas de forma consecutiva, puede calcularse el tiempo residual de una pieza con los siguientes valores de temporización.

- tiempo de mecanizado de la última pieza producida (ver $AC_OLD_PROG_NET_TIME)
- Tiempo de mecanizado actual (ver $AC_ACT_PROG_NET_TIME)

El tiempo residual se muestra en la interfaz de usuario junto con el tiempo de mecanizado actual.

Nota

Uso de STOPRE

Las variables del sistema $AC_OLD_PROG_NET_TIME y $AC_OLD_PROG_NET_TIME_COUNT no generan ninguna parada de decodificación previa implícita. Esto no es ningún problema si se usa en el programa de pieza cuando el valor de las variables del sistema proviene de la ejecución de programa anterior. Sin embargo, cuando el disparador de la medición del tiempo de ejecución ($AC_PROG_NET_TIME_TRIGGER) se escribe con alta frecuencia y a causa de ello $AC_OLD_PROG_NET_TIME cambia muy a menudo, debe utilizarse un STOPRE explícito en el programa de pieza.

Condiciones

- **Búsqueda de secuencia**
  En la búsqueda de secuencia no se calculan tiempos de ejecución del programa.
- **REPOS**
  La duración de un proceso REPOS se suma al tiempo de mecanizado actual ($AC_ACT_PROG_NET_TIME).

Ejemplos

Ejemplo 1: medir la duración de "mySubProgrammA"

Código de programa

```
...
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
```

Cuando el programa ha procesado la línea N80, en $AC_OLD_PROG_NET_TIME aparece el tiempo de ejecución neto de "mySubProgrammA".

El valor de $AC_OLD_PROG_NET_TIME:

- se mantiene más allá de M30.
- se actualiza tras cada rectificado.

Ejemplo 2: medir la duración de "mySubProgrammA" y "mySubProgrammC"

Código de programa
...
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 Contador de piezas

La función "Contador de piezas" ofrece diversos contadores que pueden utilizarse, sobre todo, para contar piezas dentro del control.

Los contadores existen como variables de sistema específicas del canal con acceso de escritura y lectura, en la gama de valores de 0 a 999 999 999.

Variable del sistema	Descripción
$AC_REQUIRED_PARTS	Número de piezas que deben fabricarse (cantidad teórica de piezas) En este contador puede definirse el número de piezas que, al ser alcanzado, pone a "0" el número real de piezas ($AC_ACTUAL_PARTS).
$AC_TOTAL_PARTS	Número total de piezas fabricadas (cantidad real total de piezas) Este contador indica el número total de piezas fabricadas desde el momento de arranque. El valor sólo se resetea automáticamente a "0" con un arranque del control con valores estándar.
$AC_ACTUAL_PARTS	Número de piezas fabricadas (cantidad real de piezas) En este contador se registra el número total de piezas fabricadas desde el momento del arranque. Al alcanzar el número teórico de piezas ($AC_REQUIRED_PARTS), el contador se pone a "0" automáticamente (siempre que $AC_REQUIRED_PARTS > 0).
$AC_SPECIAL_PARTS	Número de piezas contadas por el usuario Este contador permite al usuario contar las piezas según su propia definición. Puede definirse que se emita una alarma al alcanzar el número teórico de piezas ($AC_REQUIRED_PARTS). El mismo usuario tiene que realizar la puesta a cero del contador.

Nota

Todos los contadores de piezas se ponen a "0" en un arranque del control con valores estándar y pueden leerse y escribirse independientemente de su activación.

Nota

A través de los datos de máquina específicos del canal se puede influir en la activación de contadores, el momento de la puesta a cero y el algoritmo de recuento.

Nota

Contador de piezas con comando M definido por el usuario

A través de los datos de máquina puede definirse que los impulsos de contaje para los diferentes contadores de pieza se disparen mediante comandos M definidos por el usuario en vez de hacerlo mediante el fin del programa `M2`/`M30`.

3.23 Otras funciones

3.23.1 Activar los datos de máquina (NEWCONF)

El comando NEWCONF permite activar todos los datos de máquina. Esta función también se puede activar en la interfaz hombre-máquina (HMI), accionando el pulsador de menú "Activar DM".

Durante la ejecución de la función "NEWCONF" se efectúa una parada de decodificación previa implícita, es decir, se interrumpe el movimiento interpolado.

Sintaxis

```
NEWCONF
```

Descripción

NEWCONF:	Comando para activar todos los datos de máquina del escalón de activación "NEW_CONFIG"

Ejecutar NEWCONF desde el programa de pieza en varios canales

Si se modifican datos de máquina por ejes desde el programa de pieza y se activan después con NEWCONF, el comando NEWCONF sólo activa los datos de máquina que producen modificaciones para el canal del programa de pieza.

Nota

Para asegurar que se activan todas las modificaciones, el comando NEWCONF se tiene que ejecutar en cada canal en el que se calculan actualmente los ejes o funciones afectados por los datos de máquina modificados.

En el caso de NEWCONF no se activa ningún dato de máquina por eje.

Para ejes controlados por el PLC debe realizarse un RESET por eje.

Ejemplo

Fresado: mecanizar posición de taladrado con diferentes tecnologías

Código de programa	**Comentarios**
N10 $MA_CONTOUR_TOL[AX]=1.0	; Modificar dato de máquina.
N20 NEWCONF	; Activar datos de máquina.
...	

3.23.2 Comprobar el repertorio del lenguaje CN disponible (STRINGIS)

La función "STRINGIS(...)" permite comprobar si la cadena indicada está disponible como elemento del repertorio actual del lenguaje de programación CN.

Definición

```
INT STRINGIS(STRING <Nombre>)
```

Sintaxis

```
STRINGIS(<Nombre>)
```

Descripción

STRINGIS:	Función con valor de retorno
<Nombre>:	Nombre del elemento del lenguaje de programación CN que debe comprobarse
Valor de retorno:	El formato del valor de retorno es yxx (decimal).

Elementos del lenguaje de programación CN

Pueden comprobarse los siguientes elementos del lenguaje de programación CN:

- Comandos G de todos los grupos G existentes, p. ej., "G0", "INVCW", "POLY", "ROT", "KONT", "SOFT", "CUT2D", "CDON", "RMBBL", "SPATH"
- Direcciones DIN o CN, como, p. ej., "ADIS", "RNDM", "SPN", "SR", "MEAS"
- Funciones, como, p. ej., "TANG(...)" o "GETMDACT"
- Procedimientos, p. ej., "SBLOF"
- Palabras reservadas, p. ej., "ACN", "DEFINE" o "SETMS"
- Datos de sistema, p. ej., datos de máquina $M..., datos de operador $S... o datos de opción $O...
- Variables de sistema $A..., $V..., $P...
- Parámetros de cálculo R...
- Nombres de ciclos activados
- Variables GUD y LUD
- Nombres de macro
- Nombres de label

Valor de retorno

El valor de retorno solo es relevante en los 3 primeros números decimales. El formato del valor de retorno es yxx, siendo y = información básica y xx = información detallada.

Valor de retorno	Descripción	
000	La cadena 'nombre' no se conoce en el sistema actual [1)]	
100	La cadena 'nombre' es un elemento del lenguaje de programación CN, pero actualmente no se puede programar (opción/función inactiva)	
2xx	La cadena 'nombre' es un elemento programable del lenguaje de programación CN (opción/función activa). La información detallada xx contiene más datos sobre el tipo de elemento:	
	xx	Descripción
	01	Dirección DIN o dirección CN [2)]
	02	Comando G (p. ej., G04, INVCW)
	03	Función con valor de retorno
	04	Función sin valor de retorno
	05	Palabra reservada (p. ej., DEFINE)
	06	Dato de máquina ($M...), de operador ($S...) o de opción ($O...)
	07	Parámetros de sistema, p. ej., variables de sistema ($...) o parámetros de cálculo (R...)
	08	Ciclo (el ciclo debe estar cargado en el CN y los programas de ciclo deben estar activos [3)])
	09	Variables GUD (la variable GUD debe estar definida en el fichero de definición GUD y la variable GUD debe estar activada)
	10	Nombres de macro (la macro debe estar definida en el fichero de definición de macros y la macro debe estar activada) [4)]
	11	Variables LUD del programa de pieza actual
	12	Comando G ISO (debe estar activado el modo de lenguaje ISO)
400	La cadena 'nombre' es una dirección CN que no se ha reconocido como xx == 01 o bien xx == 10 y tampoco es G o R [2)]	
y00	Sin posibilidad de asignación específica	

1) En determinadas circunstancias, en función del control solo se conoce un subconjunto de los comandos de lenguaje CN de Siemens, p. ej. SINUMERIK 802D sl. En estos controles, para las cadenas que son en principio comandos de lenguaje CN de Siemens, se devuelve el valor 0. Este comportamiento puede modificarse con MD10711 $MN_NC_LANGUAGE_CONFIGURATION. Con MD10711 = 1 se devuelve entonces siempre el valor 100 para los comandos de lenguaje de programación CN de Siemens.

2) Son direcciones CN las siguientes letras: A, B, C, E, I, J, K, Q, U, V, W, X, Y, Z. Estas direcciones CN también pueden programarse con una ampliación de dirección. La ampliación de la dirección puede indicarse en la comprobación con STRINGIS. Ejemplo: 201 == STRINGIS("A1").
Las letras: D, F, H, L, M, N, O, P, S, T son direcciones CN o funciones auxiliares definidas por el usuario. Para ellas siempre se devuelve el valor 400. Ejemplo: 400 == STRINGIS("D"). Estas direcciones CN no pueden indicarse con ampliación de dirección en la comprobación con STRINGIS.
Ejemplo: 000 == STRINGIS("M02"), pero 400 == STRINGIS("M").

3) Los nombres de parámetros de ciclos no pueden comprobarse con STRINGIS.
4) Dirección definida como macro; p. ej., G, H, M y L se identifican como macros.

Ejemplos

En los ejemplos siguientes se supone que los elementos de programación CN indicados como cadena, si no se indica especialmente, pueden programarse en principio en el control.

1. La cadena "T" está definida como función auxiliar:
   ```
   400 == STRINGIS("T")
   000 == STRINGIS ("T3")
   ```
2. La cadena "X" está definida como eje:
   ```
   201 == STRINGIS("X")
   201 == STRINGIS("X1")
   ```
3. La cadena "A2" está definida como dirección CN con ampliación:
   ```
   201 == STRINGIS("A")
   201 == STRINGIS("A2")
   ```
4. La cadena "INVCW" está definida como comando G denominado:
   ```
   202 == STRINGIS("INVCW")
   ```
5. La cadena "$MC_GCODE_RESET_VALUES" está definida como dato de máquina:
   ```
   206 == STRINGIS("$MC_GCODE_RESET_VALUES")
   ```
6. La cadena "GETMDACT" es una función de lenguaje CN:
   ```
   203 == STRINGIS("GETMDACT")
   ```
7. La cadena "DEFINE" es una palabra reservada:
   ```
   205 == STRINGIS("DEFINE")
   ```
8. La cadena "$TC_DP3" es un parámetro de sistema (componente de longitud de herramienta):
   ```
   207 == STRINGIS("$TC_DP3")
   ```
9. La cadena "$TC_TP4" es un parámetro de sistema (tamaño de herramienta):
   ```
   207 == STRINGIS("$TC_TP4")
   ```
10. La cadena "$TC_MPP4" es un parámetro de sistema (estado de puesto de almacén):
    - La gestión de almacén de herramientas está activa: `207 == STRINGIS("$TC_MPP4") ;`
    - La gestión de almacén de herramientas no está activa: `000 == STRINGIS("$TC_MPP4")`

    Ver el apartado: Gestión de almacén de herramientas.
11. La cadena "MACHINERY_NAME" está definida como variable GUD:
    ```
    209 == STRINGIS("MACHINERY_NAME")
    ```
12. La cadena "LONGMACRO" está definida como macro:
    ```
    210 == STRINGIS("LONGMACRO")
    ```
13. La cadena "MYVAR" está definida como variable LUD:
    ```
    211 == STRINGIS("MYVAR")
    ```
14. La cadena "XYZ" no es un comando, variable GUD, nombre de macro ni nombre de ciclo conocido en el CN:
    ```
    000 == STRINGIS("XYZ")
    ```

Gestión de almacén de herramientas

Si no está activa la función de gestión de almacén de herramientas, para los parámetros de sistema de la gestión de almacén de herramientas, STRINGIS suministra, con independencia del dato de máquina

- MD10711 $MN_NC_LANGUAGE_CONFIGURATION,

siempre el valor 000.

Modo ISO

Si está activa la función "Modo ISO":

- MD18800 $MN_MM_EXTERN_LANGUAGE (Activación de lenguajes externos del CN)
- MD10880 $MN_ MM_EXTERN_CNC_SYSTEM (Sistema de control que se debe adaptar)

STRINGIS comprueba primero la cadena indicada como comando G de SINUMERIK. Si la cadena no es un comando G de SINUMERIK, se comprueba seguidamente si es comando G ISO.

Las conmutaciones programadas (`G290` (modo SINUMERIK), `G291` (modo ISO)) no tienen ninguna repercusión en STRINGIS.

Ejemplo

Los datos de máquina relevantes para la función STRINGIS(...) tienen los valores siguientes:

- MD10711 $MN_NC_LANGUAGE_CONFIGURATION = 2 (se consideran conocidos únicamente los comandos de lenguaje CN cuyas opciones están activadas)
- MD19410 $ON_TRAFO_TYPE_MASK = 'H0' (opción: transformadas)
- MD10700 $MN_PREPROCESSING_LEVEL='H43' (Preprocesamiento para ciclos activo)

El siguiente programa de ejemplo se ejecuta sin aviso de error:

Código de programa	**Comentarios**
`N1 R1=STRINGIS("TRACYL")`	`; R1 == 0, ya que TRACYL se considera "no conocido" debido a que falta la opción de transformada`
`N2 IF STRINGIS("TRACYL") == 204`	
`N3 TRACYL(1,2,3)`	`; N3 se ignora`
`N4 ELSE`	
`N5 G00`	`; y se ejecuta en su lugar N5`
`N6 ENDIF`	
`N7 M30`	

3.23.3 Llamar interactivamente la ventana del programa de pieza (MMC)

Con el subprograma predefinido MMC(...) pueden mostrarse cuadros de diálogo específicos de usuario en la interfaz de usuario desde un programa CN.

Pueden configurarse los siguientes tipos de cuadros de diálogo:

- Run MyScreens
- Easy XML
- XML de usuario

Más información en:

- Manual de programación Run MyScreens
- Manual de programación Easy XML

Sintaxis

```
MMC("<ADDRESS>,<COMMAND>,<FILE>,<DIALOG>","<QUIT>")
```

Descripción

`MMC(...)`:	Identificador de subprograma Los parámetros se indican con codificación de puesto y separados por comas en dos cadenas, la cadena de comando y la cadena de confirmación.	
Parámetro dentro de la cadena de comando:		
`<ADDRESS>`:	Campo de manejo en el cual se ejecutan los cuadros de diálogo de usuario configurados	
	Función	**Campos de manejo**
	Cuadro de diálogo de usuario "Run MyScreens"	`CYCLES`
	Cuadro de diálogo de usuario "Easy XML"	`CYCLES`
	XML de usuario	`XML`
	Ventana emergente "Run MyScreens"	`POPUPDLG`
	Ventana emergente "Easy XML"	`POPUPDLG`
`<COMMAND>`:	Comando para ejecutar	
	Función	**Comandos**
	Cuadro de diálogo de usuario "Run MyScreens"	`PICTURE_ON`, `PICTURE_OFF`
	Cuadro de diálogo de usuario "Easy XML"	`PICTURE_ON`, `PICTURE_OFF`
	XML de usuario	`XML_ON`, `XML_OFF`
	Ventana emergente "Run MyScreens"	`PICTURE_ON`, `PICTURE_OFF`
	Ventana emergente "Easy XML"	`PICTURE_ON`, `PICTURE_OFF`

<FILE>:	Nombre del fichero en el que se ha programado el cuadro de diálogo para mostrar	
	Función	**Ficheros**
	Cuadro de diálogo de usuario "Run MyScreens"	<name>.com
	Cuadro de diálogo de usuario "Easy XML"	<name>.xml
	XML de usuario	<name>.xml
	Ventana emergente "Run MyScreens"	<name>.com
	Ventana emergente "Easy XML"	<name>.xml
	Ventana emergente "Easy XML" con configuración directa en el programa CN (ver ejemplo 2)	xmldial_emb.xml
<DIALOG>:	Nombre del cuadro de diálogo que se mostrará	
	Función	**Nombre del cuadro de diálogo**
	Todas las funciones excepto ventana emergente "Easy XML" con configuración directa en el programa CN	Nombre del cuadro de diálogo configurado en el fichero <FILE>
	Ventana emergente "Easy XML" con configuración directa en el programa CN (ver ejemplo 3)	main
Parámetros dentro de la cadena de confirmación:		
<QUIT>:	Tipo de confirmación	
	N:	Sin confirmación. La ejecución del programa continúa cuando se ha transmitido el comando. No se recibe respuesta si el comando no se ha ejecutado correctamente. **Nota** El tipo de confirmación "N" debe utilizarse si se programa un tiempo de visualización (tiempo de espera) en el programa CN (ver ejemplo 2 siguiente)
	A:	Confirmación asíncrona El programa sigue ejecutándose después de transmitirse el comando. El valor de retorno se guarda en una variable de confirmación específica de usuario (variable GUD) definida en la configuración de diálogo y puede leerse en el programa CN.

Condiciones

- Los ficheros de definición *.com de los cuadros de diálogo deben guardarse en la carpeta "proj".
- Los ficheros de definición Easy XML *.xml de los cuadros de diálogo deben guardarse en la carpeta "appl".
  Si los ficheros de definición se guardan en un directorio diferente, la ruta de acceso deberá indicarse indirectamente comenzando desde el directorio "appl".
- Los cuadros de diálogo definidos por el usuario no pueden mostrarse desde distintos canales simultáneamente.
- En la simulación no se soporta la funcionalidad MMC.

Ejemplos

Ejemplo 1

Visualización de un cuadro de diálogo y reacción a la acción del usuario en un programa CN.

```
; La variable de confirmación QUIT ya se creó como variable global de usuario (GUD)
; del tipo STRING al configurar el cuadro de diálogo:
; DEF NCK STRING[20] QUIT
QUIT = "XXX"                                ; Inicializar variable de confirmación
G4 F5
MMC("CYCLES,PICTU-                          ; Mostrar cuadro de diálogo
RE_ON,test.com,test1","A")                  ; - Campo de manejo: CYCLES
                                            ;  - Estado de imagen: PICTURE_ON (mos-
                                            trar)
                                            ; - Fichero de imagen de cuadro de diálo-
                                            go: test.com
                                            ; - Imagen de cuadro de diálogo: test1
INPUT:                                      ; Esperar acción del usuario
   STOPRE                                   ;  Parada de decodificación previa
   IF MATCH (QUIT,"RUN") >= 0 GOTOF WORK    ;    Pulsador de menú "RUN"
   IF MATCH (QUIT,"CHK") >= 0 GOTOF CHECK   ;    Pulsador de menú "CHK"
GOTOB INPUT                                 ; => Esperar

WORK:                                       ; Pulsador de menú "RUN" pulsado
MSG ("Continuar mecanizado -> Marcha CN")   ; Emitir aviso
MMC("CYCLES,PICTURE_OFF","N")               ; Cerrar cuadro de diálogo
M0                                          ; Esperar a inicio de CN
GOTOF END                                   ; => Fin del programa

CHECK:                                      ; Pulsador de menú "CHK" pulsado
MSG ("Desplazar a posición -> Marcha CN")   ; Emitir aviso
MMC("CYCLES,PICTURE_OFF","N")               ; Cerrar cuadro de diálogo
M0                                          ; Esperar a inicio de CN
GOTOF END                                   ; => Fin del programa

END:
...
```

Código de programa | **Comentarios**

Ejemplo 2

El tiempo de visualización de un cuadro de diálogo en el programa CN se define, p. ej., mediante un tiempo de espera.

```
F1000 G94
...
```

Código de programa | **Comentarios**

Código de programa	**Comentarios**
MMC("POPUPDLG,PICTURE_**ON**,xmldial_emb.xml,main","N")	; Mostrar cuadro de diálogo
X200	
Z40	
MMC("POPUPDLG,PICTURE_**OFF**","N")	; Cerrar cuadro de diálogo

Ejemplo 3

Incorporación de un script emergente en un programa CN y su utilización.

Código de programa

```
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>
```

Código de programa

```
;          <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>
; ========================= Sección de programa ==========================
...
G94 F100
MMC("POPUPDLG,PICTURE_ON,xmldial_emb.xml,main","N")
G4 F4
X200
MMC("POPUPDLG,PICTURE_OFF","N")
G4 F2
X0
...
```

3.23.4 Process DataShare - Salida a un equipo/fichero externo (EXTOPEN, WRITE, EXTCLOSE):

La escritura de datos de un programa de pieza en un equipo/fichero externo se realiza en tres pasos:

1. Abrir equipo/fichero externo
   Con el comando EXTOPEN se abre el equipo/fichero externo para el canal en el cual se va a escribir.
2. Escribir datos
   El dato de salida puede prepararse con las funciones de string del lenguaje CN, p. ej., SPRINT. La escritura en sí se realiza mediante el comando WRITE.
3. Cerrar equipo/fichero externo
   Con el comando EXTCLOSE o al alcanzar el final del programa (M30), así como al resetear el canal, se vuelve a habilitar el equipo/fichero externo asignado al canal.

Sintaxis

```
DEF INT <Result>
DEF STRING[<n>] <Output>
…
EXTOPEN(<Result>,<ExtDev>,<SyncMode>,<AccessMode>,<WriteMode>)
…
<Output>="Salida de datos"
WRITE(<Result>,<ExtDev>,<Output>)
…
EXTCLOSE(<Result>,<ExtDev>)
```

Descripción

EXTOPEN:	Procedimiento predefinido para abrir un equipo/fichero externo		
<Result>:	**Parámetro 1:** Variable de resultado Con ayuda del valor de variable de resultado se evalúa en el programa si la operación se ha realizado correctamente y este puede continuarse en función de los resultados.		
	Tipo:	INT	
	Valores:	0	Sin errores
		1	El equipo externo no puede abrirse
		2	El equipo externo no está configurado
		3	Equipo externo configurado con ruta no válida
		4	Acceso no autorizado al equipo externo
		5	Modo de uso: Equipo externo ya asignado en modo "exclusivo"
		6	Modo de uso: Equipo externo ya asignado en modo "compartido"
		7	Tamaño de fichero superior a LOCAL_DRIVE_MAX_FILESIZE
		8	Número máximo de equipos externos superado
		9	Opción para LOCAL_DRIVE no activada
		11	Reservado
		12	Modo de escritura: Indicación contradictoria con extdev.ini
		16	Ruta externa programada no válida
		22	Equipo externo no montado

<ExtDev>:	**Parámetro 2:** Identificador simbólico para el equipo/fichero externo que se debe abrir	
	Tipo:	STRING
	El identificador simbólico está compuesto por: 1. el nombre lógico del equipo; 2. dado el caso, seguido de una ruta de fichero (unida con "/"). Están definidos los siguientes **nombres lógicos del equipo**:	
	"LOCAL_DRIVE":	Tarjeta CompactFlash local (predefinida)
	"CYC_DRIVE":	Indicación de unidad reservada para el uso en ciclos de SIEMENS (predefinida)
	"/dev/ext/1" ... "/dev/ext/9":	Unidades de red disponibles **Nota:** Configuración necesaria en el fichero extdev.ini.
	"/dev/cyc/1", "/dev/cyc/2":	Indicaciones de unidades reservadas para el uso en ciclos de SIEMENS **Nota:** Configuración necesaria en el fichero extdev.ini.
	Ruta de fichero: • Para "LOCAL_DRIVE" y "CYC_DRIVE" debe indicarse una ruta de fichero, p. ej.: "LOCAL_DRIVE/my_dir/my_file.txt" • Los nombres lógicos de equipo "/dev/ext/1...9" y "/dev/cyc/1...2" pueden, en cada configuración: – remitir ya a un fichero (en ese caso, debe indicarse solamente el nombre lógico del equipo, p. ej.: "/dev/ext/4") – o bien a un directorio (en ese caso debe indicarse una ruta de fichero, p. ej.: "/dev/ext/5/my_dir/my_file.txt")	
	Nota: Para los nombres lógicos del equipo "/dev/ext/1...9", "/dev/v24" y "/dev/cyc/1...2" no se diferencia entre mayúsculas y minúsculas, mientras que en la indicación de la ruta de un fichero, la diferencia entre mayúsculas y minúsculas es significativa. Para "LOCAL_DRIVE" y "CYC_DRIVE" solo se admiten mayúsculas.	

<SyncMode>:	**Parámetro 3:** Modo de edición para los comandos WRITE de este equipo/fichero		
	Tipo:	STRING	
	Valores:	"SYN":	Escritura sincrónica La ejecución del programa se detiene hasta que haya finalizado el proceso de escritura. Evaluando las variables de error del comando WRITE puede comprobarse si la escritura sincrónica ha finalizado correctamente.
		"ASYN":	Escritura asincrónica La ejecución del programa no se interrumpe mediante el comando WRITE. **Nota:** La variable de error del comando WRITE no tiene valor informativo en este modo y siempre tiene el valor 0 (no hay error). En este modo no existe ninguna certeza de que el comando WRITE haya funcionado correctamente.
<AccessMode>:	**Parámetro 4:** Modo de uso para este equipo/fichero		
	Tipo:	STRING	
	Valores:	"SHARED":	se solicita el acceso al equipo/fichero en modo compartido. Otros canales pueden utilizar el equipo; es decir, abrirlo también en este modo.
		"EXCL":	el equipo/fichero se utiliza en el canal de manera exclusiva; ningún otro canal puede usar el equipo.
<WriteMode>:	**Parámetro 5:** Modo de escritura para los comandos WRITE de este equipo/fichero (opcional)		
	Tipo:	STRING	
	Valores:	"APP":	Añadir El contenido del fichero se conserva; las peticiones de escritura se añaden al final.
		"OVR":	Sobrescribir El contenido del fichero se borra y se genera de nuevo mediante las peticiones de escritura siguientes.
	Nota: Este parámetro no permite sobrescribir el modo de escritura configurado en el fichero extdev.ini. En caso de conflicto, la llamada EXTOPEN se confirma con un error.		

WRITE:	Procedimiento predefinido para escribir los datos de salida

EXTCLOSE:	Procedimiento predefinido para cerrar un equipo/fichero externo abierto		
<Result>:	**Parámetro 1:** Variable de resultado		
	Tipo:	INT	
	Valores:	0	Sin errores
		16	Ruta externa programada no válida
		21	Error al cerrar el equipo externo
<ExtDev>:	**Parámetro 2:** Identificador simbólico para el equipo/fichero externo que se debe cerrar. Ver descripción en EXTOPEN. **Nota:** El identificador debe ser exactamente igual al indicado en la llamada EXTOPEN.		

Ejemplo

Código de programa

```
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("Error en EXTOPEN:" << RESULT)
N80       ELSE
N90          EXTDEVICE=TRUE
N100      ENDIF
…
N200      PHASE=4
N210      IF EXTDEVICE
N220         OUTPUT=SPRINT("Fin fase: %D",PHASE)
N230         WRITE(RESULT,"LOCAL_DRIVE/my_file.txt",OUTPUT)
N240      ENDIF
…
```

Consulte también

Operaciones con cadenas de caracteres (Página 461)

Escribir fichero (WRITE) (Página 582)

3.23.5 Alarmas (SETAL)

Es posible fijar alarmas en un programa CN. Las alarmas se visualizan en la interfaz de usuario dentro de un campo especial. Una alarma provoca una reacción en el control numérico que depende de la categoría a la que ha sido asociada.

Sintaxis

```
SETAL(<N.°>[,<String>])
```

Descripción

SETAL(...):	Procedimiento predefinido para activar una alarma SETAL se debe programar en una secuencia CN propia.		
<Nr>:	Número de alarma		
	Tipo de dato:	INT	
	Rango de valores:	60000 ... 64999 (reservado)	Alarmas para ciclos SIEMENS
		65000 ... 69999	Alarmas para ciclos de usuario
<String>:	Cadena de caracteres (opcional)		
	Tipo de dato:	STRING	
	Durante la programación de alarmas de ciclos de usuario se puede introducir una cadena de caracteres con un máximo de 4 parámetros. En estos parámetros se pueden definir textos de usuario variables. También están disponibles los siguientes parámetros predefinidos:		
	%1	Número de canal	
	%2	número de secuencia, lábel	
	%3	Índice de texto para alarmas de ciclo	
	%4	Parámetro de alarma adicional	

Nota

Los textos de alarma deben configurarse en la interfaz de usuario.

Nota

Si una alarma debe emitirse en el idioma activo en la interfaz de usuario, el usuario necesitará información sobre el idioma ajustado actualmente en el HMI. Esta información se puede consultar en el programa de pieza y en acciones síncronas, a través de la variable de sistema $AN_LANGUAGE_ON_HMI (Página 1355).

Ejemplo

```
Código de programa                  Comentarios
...
N100 SETAL (65000)                  ; Activar la alarma n.° 65000
...
```

Información adicional

Reacción y confirmación de alarma

Las alarmas de ciclos de usuario están asignadas a rangos de números que se diferencian en reacción y confirmación de alarma:

Rango de números	Reacción por la alarma	Confirmación de alarma
65000 - 65499	Visualización, bloqueo Marcha CN	Reset
65500 - 65999	Visualización, enclavamiento arranque CN (no para los ASUP con MD20194 establecido)	Reset
66000 - 66999	Visualización, enclavamiento arranque CN, parada del movimiento tras ejecución de las secuencias predecodificadas	Reset
67000 - 67999	Visualización	Cancelar
68000 - 68999	Visualización, bloqueo Marcha CN, parada inmediata del interpolador	Reset
69000 - 69999	Visualización, bloqueo Marcha CN, parada en próximo fin de secuencia	Reset

3.23.6 Definir pieza en bruto (WORKPIECE)

El controlador debe conocer la forma y el tamaño de la pieza en bruto para poder mostrarla en la simulación gráfica. El usuario tiene así la posibilidad de definir piezas en bruto en la interfaz de usuario o directamente en el programa CN. Las definiciones de las piezas en bruto se conservan aunque se ejecute un reset (de fin de programa/de canal/de GMO). Se borran automáticamente con el siguiente arranque del control.

Sintaxis

```
WORKPIECE("<WP>", "<RefP>", "<ZeroOffset>", "<Type>", <Par5>,
<Par6>, ..., <Par12>)
```

Descripción

`WORKPIECE(...)`:		Procedimiento predefinido para definir una pieza en bruto	
		Parada de decodificación previa:	Sí
		Único elemento de la secuencia:	Sí
Parámetros:			
1	`"<WP>"`:	Nombre de la pieza (opcional)	
		Tipo de dato:	STRING
		Solo hay que indicar un dato si puede haber varias piezas en un canal. Si no se indica nada, se toma automáticamente "WORKP<n>", siendo <n> el número del canal declarado.	

2	"<RefP>":	Amarre (opcional; solo para fresadoras)		
		Tipo de dato:	STRING	
		Rango de valores:	"Mesa"	Amarre sobre la mesa fija
			"A"	Amarre en el eje giratorio A
			"B"	Amarre en el eje giratorio B
			"C"	Amarre en el eje giratorio C
		Requisito: La mesa y el eje giratorio deben haberse habilitado para el amarre de la pieza en bruto mediante el dato de máquina correspondiente (ver Manual de puesta en marcha SINUMERIK Operate).		
3	"<ZeroOffset>":	Decalaje de origen ajustable para posicionar la pieza en bruto (no programable) La selección de un decalaje de origen ajustable para posicionar la pieza en bruto está disponible solo si la pieza en bruto se define a través de la interfaz de usuario. Si la pieza en bruto se define directamente en el programa de pieza, la pieza se referirá siempre al decalaje de origen actual.		
4	"<Type>":	Forma de la pieza en bruto		
		Tipo de dato:	STRING	
		Rango de valores:	"CYLINDER":	Cilindro
			"PIPE":	Tubo
			"RECTANGLE":	Paralelepípedo centrado
			"BOX":	Paralelepípedo
			"N_CORNER":	Polígono
5 ... 12	<Par5> ... <Par12>:	Parámetros para la descripción de la forma de la pieza en bruto		
		Tipo de dato:	REAL	
		El número de parámetros necesarios y su significado dependen de la forma de cada pieza en bruto y del valor del parámetro de bit. Ver: • Tabla "Parámetros para la descripción de la forma de la pieza en bruto" • Tabla "Parámetros de bit"		
WORKPIECE():		Una llamada de WORKPIECE sin parámetros borra todas las definiciones de piezas en bruto.		
WORKPIECE(<WP>):		Una llamada de WORKPIECE con un nombre de pieza borra solo la definición de la pieza en bruto correspondiente.		

Tabla 3-6 Parámetros para la descripción de la forma de la pieza en bruto

Forma de la pieza en bruto	Parámetros							
	<Par5>	<Par6>	<Par7>	<Par8>	<Par9>	<Par10>	<Par11>	<Par12>
Cilindro	Parámetros de bit Valor real que se interpreta como número entero codificado en bits. Los bits definen el significado de los siguientes parámetros (ver tabla "Parámetros de bit").	Punto de referencia Z_0	Longitud Z_1	Cota de mecanizado Z_B	Diámetro exterior d_0	-	Giro alrededor del eje giratorio	-
Tubo		Punto de referencia Z_0	Longitud Z_1	Cota de mecanizado Z_B	Diámetro exterior d_0	Espesor de pared (inc)/ diámetro interior d_1 (abs)	Giro alrededor del eje giratorio	-
Paralelepípedo centrado		Punto de referencia Z_0	Longitud Z_1	Cota de mecanizado Z_B	Anchura W	Longitud L	Giro alrededor del eje giratorio	-
Paralelepípedo		Punto de referencia Z_0	Longitud Z_1	Cota de mecanizado Z_B	X_0	Y_0	X_1	Y_1
Polígono		Punto de referencia Z_0	Longitud Z_1	Cota de mecanizado Z_B	Número esquinas	Ancho de llave	Giro alrededor del eje giratorio	-

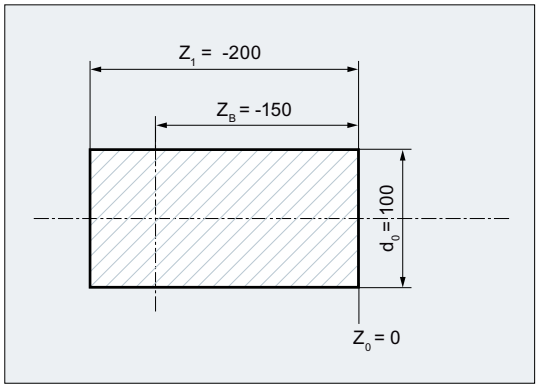
Tabla 3-7 Parámetros de bit

Bit	Descripción	
4 (0x0010)	Paralelepípedo: X_1	
	= 0	inc
	= 1	abs
5 (0x0020)	Paralelepípedo: Y_1	
	= 0	inc
	= 1	abs
6 (0x0040)	Longitud Z_1 (cota final)	
	= 0	inc
	= 1	abs
Bit 7 (0x0080)	Cota de mecanizado Z_B	
	= 0	inc
	= 1	abs
Bit 8 (0x0100)	Tubo: Espesor de pared/diámetro interno	
	= 0	inc
	= 1	abs
9 (0x0200)	Polígono	
	= 0	Ancho de llave
	= 1	Longitud del borde

Bit	Descripción	
12 (0x1000)	Amarre para tornos	
	= 0	Cabezal principal
	= 1	Contracabezal
13 (0x2000)	Contracabezal	
	= 0	con simetría especular
	= 1	sin simetría especular

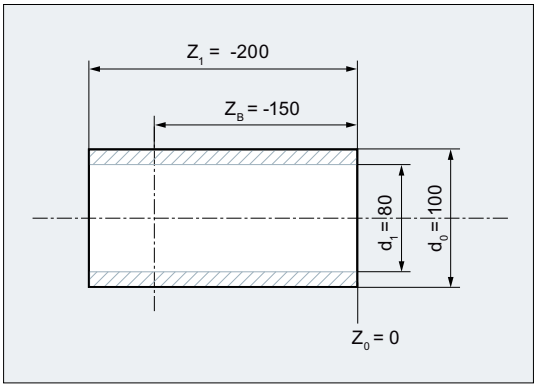
Ejemplos

Ejemplo 1: Pieza en bruto cilíndrica en un torno



Código de programa	Comentarios
...	
WORKPIECE(,,,"CYLINDER",0,0,-200,-150,100)	**; Definición de pieza en bruto:**
	; Forma de la pieza en bruto: Cilindro
	; Parámetro de bit=0(sin bit activado) → Los valores de longitud y cota de mecanizado son incrementales; pieza en bruto en el cabezal principal
	; Punto de referencia(Z0)=0
	; Longitud(Z1)=-200
	; Cota de mecanizado(ZB)=-150
	; Diámetro exterior(d0)=100
...	

Ejemplo 2: Pieza en bruto tubular en un torno



Código de programa	Comentarios
...	
WORKPIECE(,,,"PIPE",256,0,-200,-150,100,80)	; **Definición de pieza en bruto:** ; Forma de la pieza en bruto: Tubo ; Parámetro de bit=256(bit8=1) → El diámetro interior es absoluto, la longitud y cota de mecanizado son incrementales; pieza en bruto en el cabezal principal ; Punto de referencia(Z0)=0 ; Longitud(Z1)=-200 ; Cota de mecanizado(ZB)=-150 ; Diámetro exterior(d0)=100 ; Diámetro interior(d1)=80
...	

3.23.7 Cambiar modo de lenguaje (G290, G291)

El controlador permite leer y procesar programas de pieza de sistemas CNC externos. El requisito es haber definido el modo de lenguaje CN (dialecto ISO) pertinente en la puesta en marcha.

El modo de dialecto ISO puede activarse por separado para cada canal. El modo de dialecto ISO puede estar activo, por ejemplo, en el canal 1 mientras en el canal 2 se utiliza el modo SINUMERIK.

El cambio entre el modo SINUMERIK y el modo de dialecto ISO en el programa CN se realiza mediante los comandos del grupo G 47. La herramienta activa, lss correcciones de herramienta y los decalajes de origen no se ven afectados por la conmutación.

Sintaxis

```
G291
```

```
...
G290
```

Descripción

G290:	Activación del modo de lenguaje SINUMERIK	
	Único elemento de la secuencia:	Sí
	Efecto:	modal
G291:	Activación del modo de lenguaje ISO	
	Único elemento de la secuencia:	Sí
	Efecto:	modal

Condiciones

Modo SINUMERIK

- El ajuste previo de los comandos G se puede establecer para cada canal a través de un dato de máquina.
- En el modo SINUMERIK no se pueden programar comandos de lenguaje de los dialectos ISO.

Modo de dialecto ISO

- El modo de dialecto ISO como ajuste básico del controlador se puede definir con datos de máquina. Después el control arrancará de forma predeterminada en el modo dialecto ISO.
- Solo se pueden programar comandos G del dialecto ISO. No es posible programar funciones SINUMERIK en el modo de dialecto ISO.
- No se puede mezclar el dialecto ISO y el lenguaje SINUMERIK en la misma secuencia CN.
- No es posible cambiar entre el dialecto ISO M (fresado) y el dialecto ISO T (torneado) utilizando comandos G.
- Se pueden hacer llamadas a subprogramas que estén programados en el modo SINUMERIK.
- Si se utilizan funciones SINUMERIK, debe cambiarse previamente al modo SINUMERIK (ver ejemplo).

Ejemplo

Compresión de secuencias lineales en el modo de dialecto ISO

Código de programa	**Comentarios**
N5 G290	; Activación del modo de lenguaje SINUME-RIK.

Código de programa	**Comentarios**
N10 COMPCAD	; COMPCAD es un comando del lenguaje Siemens que activa una función de compresor que sustituye secuencias lineales consecutivas por secuencias polinómicas con longitudes de trayectoria lo mayor posible.
N15 G291	; Activación del modo de lenguaje ISO.
N20 G01 X100 Y100 F1000	; Puesto que COMPCAD se ha activado en el modo SINUMERIK, con esta función también se pueden comprimir secuencias lineales en el modo de dialecto ISO.
...	

Información adicional

Manual de funciones Dialectos ISO

3.24 Programas de desbaste propios

3.24.1 Funciones de apoyo para el desbaste

Para el desbaste se ofrecen ciclos de ejecución terminados. Además tiene la posibilidad de crear programas de desbaste propios con las funciones indicadas a continuación:

- Creación de tabla para el contorno (CONTPRON)
- Creación de tabla para el contorno codificada (CONTDCON)
- Desactivación de la preparación del contorno (EXECUTE)
- Determinación del punto de intersección entre dos elementos de contorno (INTERSEC) (Sólo para tablas elaboradas mediante CONTPRON).
- Ejecución de los elementos de contorno de una tabla secuencia a secuencia (EXECTAB) (Sólo para tablas elaboradas mediante CONTPRON).
- Cálculo de los datos de circunferencia (CALCDAT)

Nota

Puede utilizar estas funciones de forma universal y no sólo para el desbaste.

Requisitos

Antes de la llamada a las funciones CONTPRON o CONTDCON debe:

- Realizarse un desplazamiento a un punto inicial que permita un mecanizado sin colisiones.
- Estar desactivada la compensación del radio de filo mediante `G40`.

3.24.2 Creación de tabla para el contorno (CONTPRON)

`CONTPRON` permite activar la preparación de contornos. Las secuencias CN que se van a llamar a continuación no se ejecutan, sino que se dividen en desplazamientos individuales y se almacenan en la tabla para el contorno. Cada uno de los elementos del contorno corresponde a una fila en un campo bidimensional de la tabla de contornos. Se devuelve el número de puntos de destalonado (cambio de sentido en eje x) calculados.

Sintaxis

Activar la preparación del contorno:

```
CONTPRON(<Tabla para el contorno>,<Tipo de
mecanizado>,<Destalonados>,
<Sentido de mecanizado>)
```

Desactivación de la preparación del contorno y conmutación al modo de ejecución normal de programa:

```
EXECUTE(<ERROR>)
```

Ver "Desactivación de la preparación del contorno (EXECUTE) (Página 1058)"

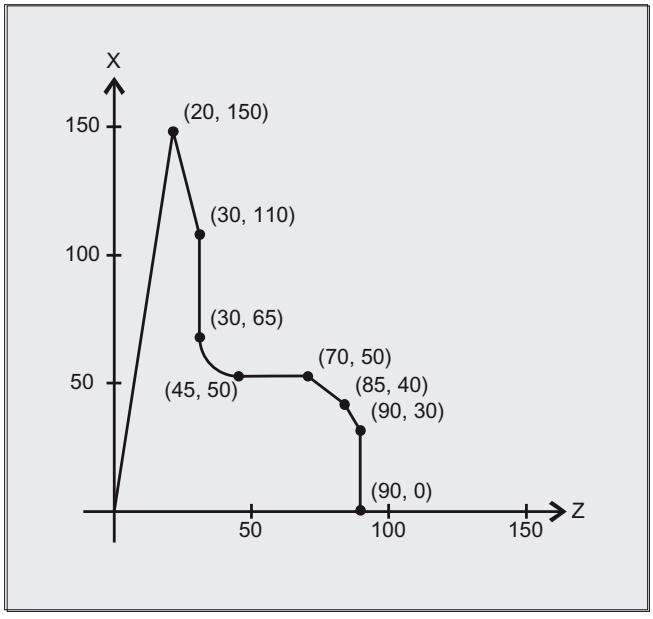
Descripción

CONTPRON:	Procedimiento predefinido para activar la preparación del contorno para crear una tabla para el contorno		
<Tabla para el contorno>:	Nombre de la tabla para el contorno		
<Tipo de mecanizado>:	Parámetros del tipo de mecanizado		
	Tipo:	CHAR	
	Valor:	"G":	Torneado longitudinal: Mecanizado interior
		"L":	Torneado longitudinal: Mecanizado exterior
		"N":	Refrentado: Mecanizado interior
		"P":	Refrentado: Mecanizado exterior
<Destalonados>:	Variable de resultado para el número de elementos de destalonado presentes		
	Tipo:	INT	
<Sentido de mecanizado>:	Parámetros del sentido de mecanizado		
	Tipo:	INT	
	Valor:	0	Preparación del contorno hacia delante (valor estándar)
		1	Preparación del contorno en ambos sentidos

Ejemplo 1

Creación de una tabla para el contorno con:

- nombres "KTAB",
- 30 elementos de contorno como máximo (arcos, rectas),
- una variable para el número de elementos de destalonado presentes,
- una variable para avisos de error.



Programa CN:

Código de programa	**Comentarios**
N10 DEF REAL KTAB[30,11]	; Tabla para el contorno con nombre KTAB y 30 elementos de contorno como máximo, el valor de parámetro 11 (número de columnas de la tabla) es un valor fijo.
N20 DEF INT CORTES	; Variable para el número de elementos de destalonado con nombre CORTES.
N30 DEF INT ERROR	; Variable para señalización de error (0=ningún error, 1=error).
N40 G18	
N50 CONTPRON(KTAB,"G",CORTES)	; Activar la preparación del contorno.
N60 G1 X150 Z20	; N60 a N120: descripción del contorno
N70 X110 Z30	
N80 X50 RND=15	
N90 Z70	
N100 X40 Z85	
N110 X30 Z90	
N120 X0	
N130 EXECUTE(ERROR)	; Finalización de la introducción de datos en la tabla para el contorno, conmutación a la ejecución normal del programa.
N140 …	; Continuación del procesamiento de la tabla.

Tabla para el contorno KTAB:

Index Línea	Columna									
(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

Explicación del contenido de las columnas:

(0) Puntero en el siguiente elemento de contorno (en el número de línea del mismo)

(1) Puntero en el elemento de contorno anterior

(2) Codificación del modo de contorno para el movimiento

Posibles valores para 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) Punto inicial de los elementos de contorno

(3) = Abscisa, (4) = Ordenada en el plano actual

(5), (6) Punto final de los elementos de contorno

(5) = Abscisa, (6) = Ordenada en el plano actual

(7) Indicador máx/mín: caracteriza los máximos y mínimos locales en el contorno

(8) Valor máximo entre el elemento de contorno y la abscisa (en el mecanizado longitudinal), o la ordenada (en el refrentado). El ángulo depende del modo de mecanizado programado.

(9), (10) Coordenadas del punto central de un elemento del contorno en el caso de que sea un
arco de circunferencia.

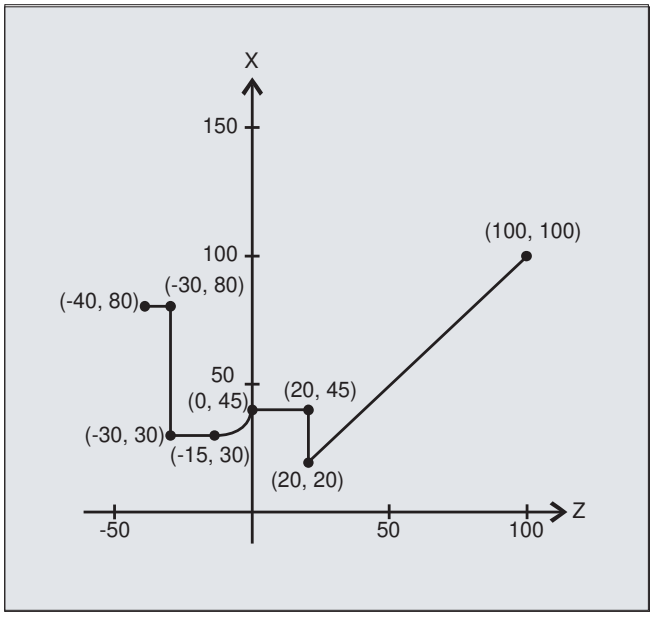
(9) = Abscisa, (10) = Ordenada

Ejemplo 2

Creación de una tabla para el contorno con

- nombres KTAB,
- 92 elementos de contorno como máximo (arcos, rectas),

- Modo de operación: torneado longitudinal, mecanizado exterior,
- preparación hacia delante y atrás.



Programa CN:

Código de programa	**Comentarios**
N10 DEF REAL KTAB[92,11]	; Tabla para el contorno con nombre KTAB y como máx. 92 elementos de contorno; el valor de parámetro 11 es un valor fijo.
N20 DEF CHAR BT="L"	; Modo de operación para CONTPRON: torneado longitudinal, mecanizado exterior,
N30 DEF INT HE=0	; Número de elementos de destalonado=0
N40 DEF INT MODE=1	; Preparación hacia delante y atrás
N50 DEF INT ERR=0	; Señalización de error
...	
N100 G18 X100 Z100 F1000	
N105 CONTPRON(KTAB,BT,HE,MODE)	; Activar la preparación del contorno.
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)	; Finalización de la introducción de datos en la tabla para el contorno, conmutación a la ejecución normal del programa.
...	

Tabla para el contorno KTAB:

Tras finalizar la preparación del contorno, éste se encuentra disponible en ambos sentidos.

Index	Columna										
Línea	(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

Explicación del contenido de las columnas y de las notas de las filas 0, 1, 6, 8, 83, 85 y 91

Son válidas las explicaciones del contenido de las columnas citadas en el ejemplo 1.

Siempre en la línea de tabla 0:

1) El. anterior: A la línea n se le asigna el extremo de contorno hacia delante

2) El. posterior: La línea n es el final de la tabla para el contorno

En cada caso una vez dentro de los elementos de contorno hacia delante:

3) El. anterior: Comienzo de contorno (hacia delante)

4) El. posterior: Fin de contorno (hacia delante)

Siempre en la línea del final de la tabla para el contorno (hacia delante) +1:

5) El. anterior: Cantidad de puntos de destalonado hacia delante

6) El. posterior: Cantidad de puntos de destalonado hacia atrás

En cada caso una vez dentro de los elementos de contorno hacia atrás:

7) El. posterior: Extremo de contorno (hacia atrás)

8) El. anterior: Comienzo de contorno (hacia atrás)

Siempre en la última línea de tabla:

9) El. anterior: La línea n es el comienzo de la tabla para el contorno (hacia atrás)

10) El. posterior: La línea n contiene el comienzo del contorno (hacia atrás)

Información adicional

Órdenes de desplazamiento permitidas, sistema de coordenadas

Para la programación de contornos se admiten los siguientes comandos G:

- Grupo G 1: `G0`, `G1`, `G2`, `G3`

También son posibles:

- redondeo y chaflán,
- Programación del círculo mediante `CIP` y `CT`

Las funciones Spline, Polinomio y Rosca producen errores.

No se pueden realizar cambios en el sistema de coordenadas mediante la activación de un frame entre las instrucciones `CONTPRON` y `EXECUTE`. Lo mismo rige para un cambio entre `G70` y `G71` o entre `G700` y `G710`.

Cualquier cambio de los ejes geométricos con `GEOAX` durante la preparación de la tabla para el contorno origina una alarma.

Elementos de destalonado

La descripción del contorno para el cálculo de los puntos de destalonado (cambio de sentido en eje x) se puede realizar en un subprograma o bien en secuencias individuales.

Desbaste independiente de la dirección de contorno programada

La preparación del contorno con `CONTPRON` se ha ampliado de forma que, tras llamarla, la tabla para el contorno estará disponible independientemente del sentido programado.

3.24.3 Creación de tabla para el contorno codificada (CONTDCON)

Cuando la preparación del contorno se activa con `CONTDCON`, las secuencias CN siguientes se almacenan de forma ventajosa para la memoria en una tabla para el contorno de 6 columnas. Cada uno de los elementos del contorno corresponde a una fila en la tabla de contornos. Conociendo las reglas de codificación abajo indicadas, se pueden confeccionar programas en código DIN, p. ej., para ciclos a partir de las líneas de la tabla. En la línea de la tabla con el número 0 se almacenan los datos del punto de partida.

Sintaxis

Activar la preparación del contorno:
`CONTDCON(<Tabla para el contorno>,<Sentido de mecanizado>)`

Desactivación de la preparación del contorno y conmutación al modo de ejecución normal de programa:
`EXECUTE(<ERROR>)`

Ver "Desactivación de la preparación del contorno (EXECUTE) (Página 1058)"

Descripción

CONTDCON:	Procedimiento predefinido para activar la preparación del contorno para crear una tabla codificada para el contorno		
<Tabla para el contorno>:	Nombre de la tabla para el contorno		
<Sentido de mecanizado>:	Parámetro para el sentido de mecanizado		
	Tipo:	INT	
	Valor:	0	Preparación del contorno según el orden de las secuencias del contorno (valor estándar)
		1	inadmisible

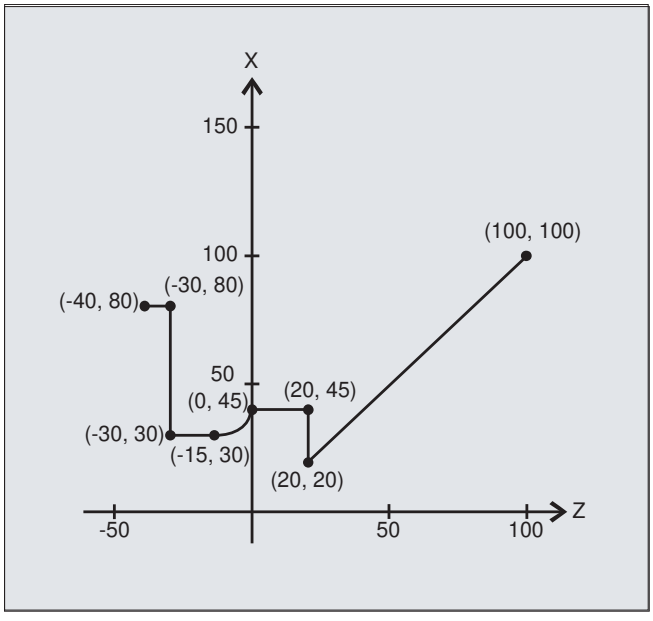
Nota

Los comandos G admitidos para CONTDCON en la parte de programa que se va a tabular son más amplios que para CONTPRON. Además se guardarán también los avances y el tipo de avance por cada sección del contorno.

Ejemplo

Creación de una tabla para el contorno con:

- nombres "KTAB",
- Elementos de contorno (arcos, rectas)
- Modo de operación: Torneado
- Dirección de mecanizado: Adelante



Programa CN:

Código de programa	**Comentarios**
N10 DEF REAL KTAB[9,6]	; Tabla para el contorno con nombres KTAB y 9 líneas de tabla. Éstas permiten 8 secuencias de contornos. El valor de parámetro 6 (número de columnas de la tabla) es un valor fijo.
N20 DEF INT MODE = 0	; Variable para el sentido de mecanizado. Valor por defecto 0: sólo en el sentido de contorno programado.
N30 DEF INT ERROR = 0	; Variable para señalización de error.
...	
N100 G18 G64 G90 G94 G710	
N101 G1 Z100 X100 F1000	
N105 CONTDCON (KTAB, MODE)	; Llamada de la preparación del contorno (MODE se puede omitir).
N110 G1 Z20 X20 F200	; Descripción del contorno.
N120 G9 X45 F300	
N130 Z0 F400	
N140 G2 Z-15 X30 K=AC(-15) I=AC(45)F100	
N150 G64 Z-30 F600	
N160 X80 F700	
N170 Z-40 F800	
N180 EXECUTE(ERROR)	; Finalización de la introducción de datos en la tabla para el contorno, conmutación a la ejecución normal del programa.
...	

Tabla para el contorno KTAB:

	Índice de columna					
	0	1	2	3	4	5
Índice de líneas	Modo de contorno	Punto final, abscisa	Punto final, ordenada	Centro, abscisa	Centro, ordenada	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

Explicación del contenido de las columnas:

Línea 0: Codificaciones para el **punto inicial:**

Columna 0:	10^0 (unidades): G0 = 0
	10^1 (decenas): G70 = 0, G71 = 1, G700 = 2, G710 = 3
Columna 1:	Punto inicial, abscisa
Columna 2:	Punto inicial ordenada
Columna 3-4:	0
Columna 5:	Índice de línea de la última sección del contorno en la tabla

Líneas 1-n: Entradas de las **secciones del contorno**

Columna 0:	10^0 (unidades): G0 = 0, G1 = 1, G2 = 2, G3 = 3
	10^1 (decenas): G70 = 0, G71 = 1, G700 = 2, G710 = 3
	10^2 (centenas): G90 = 0, G91 = 1
	10^3 (millares): G93 = 0, G94 = 1, G95 = 2, G96 = 3
	10^4 (decenas de millar): G60 = 0, G44 = 1, G641 = 2, G642 = 3
	10^5 (centenas de millar): G9 = 1
Columna 1:	Punto final, abscisa
Columna 2:	Punto final ordenada
Columna 3:	Centro abscisa en la interpolación circular
Columna 4:	Centro ordenada en la interpolación circular
Columna 5:	Avance

Información adicional

Órdenes de desplazamiento permitidas, sistema de coordenadas

Para la programación de contornos se admiten los siguientes grupos y comandos G:

Grupo G 1:	`G0`, `G1`, `G2`, `G3`
Grupo G 10:	`G60`, `G64`, `G641`, `G642`
Grupo G 11:	`G9`
Grupo G 13:	`G70`, `G71`, `G700`, `G710`
Grupo G 14:	`G90`, `G91`
Grupo G 15:	`G93`, `G94`, `G95`, `G96`, `G961`

También son posibles:

- redondeo y chaflán,
- Programación del círculo mediante `CIP` y `CT`

Las funciones Spline, Polinomio y Rosca producen errores.

No se pueden realizar cambios en el sistema de coordenadas mediante la activación de un frame entre las instrucciones `CONTDCON` y `EXECUTE`. Lo mismo rige para un cambio entre `G70` y `G71` o entre `G700` y `G710`.

Cualquier cambio de los ejes geométricos con GEOAX durante la preparación de la tabla para el contorno origina una alarma.

Sentido de mecanizado

La tabla para el contorno generada con CONTDCON está pensada para el desbaste en la dirección programada del contorno.

3.24.4 Determinación del punto de intersección entre dos elementos de contorno (INTERSEC)

INTERSEC calcula el punto de intersección o corte de dos elementos normalizados del contorno, obtenidos de tablas para el contorno generadas mediante CONTPRON.

Sintaxis

```
<Estado>=INTERSEC(<Tabla para el contorno_1>[<Elemento de
contorno_1>],
<Tabla para el contorno_2>[<Elemento de contorno_2>],<Punto de
intersección>,<Tipo de mecanizado>)
```

Descripción

INTERSEC:	Función predefinida para determinar el punto de intersección de dos elementos de contorno obtenidos de tablas para el contorno generadas mediante CONTPRON.		
<Estado>:	Variable para el estado del punto de intersección		
	Tipo:	BOOL	
	Valor:	TRUE	Se ha encontrado el punto de intersección
		FALSE	no se ha encontrado punto de intersección
<Tabla para el contorno_1>:	Nombre de la primera tabla para el contorno		
<Elemento de contorno_1>:	Número del elemento de contorno de la primera tabla para el contorno		
<Tabla para el contorno_2>:	Nombre de la segunda tabla para el contorno		
<Elemento de contorno_2>:	Número del elemento de contorno de la segunda tabla para el contorno		
<Punto de intersección>:	Coordenadas del punto de intersección en el plano activo (G17/G18/G19)		
	Tipo:	REAL	

<Tipo de mecanizado>:	Parámetros del tipo de mecanizado		
	Tipo:	INT	
	Valor:	0	Cálculo del punto de intersección en el plano activo con el parámetro 2 (valor estándar)
		1	Cálculo del punto de intersección independiente del plano transferido

Nota

Tenga en cuenta que las variables deben estar definidas antes de su utilización.

La transferencia de los contornos requiere el cumplimiento de los valores definidos con CONTPRON:

Parámetros	Descripción
2	Codificación del modo de contorno para el movimiento
3	Punto inicial de contorno, abscisa
4	Punto inicial de contorno, ordenada
5	Punto final de contorno, abscisa
6	Punto final de contorno, ordenada
9	Coordenada central de la abscisa (sólo en contornos circulares)
10	Coordenada central de la ordenada (sólo en contornos circulares)

Ejemplo

Cálculo del punto de corte del elemento del contorno 3 de la tabla TABNAME1 y del elemento del contorno 7 de la tabla TABNAME2. Las coordenadas del punto de intersección en el plano activo se almacenan en la variable ISCOORD (1.er elemento = abscisa, 2.º elemento = ordenada). En el caso de no existir un punto de intersección se realiza un salto a NOCORTA (no existe punto de intersección).

Código de programa	Comentarios
DEF REAL TABNAME1[12,11]	; Tabla para el contorno 1
DEF REAL TABNAME2[10,11]	; Tabla para el contorno 2
DEF REAL ISCOORD [2]	; Variable para las coordenadas del punto de intersección.
DEF BOOL ISPOINT	; Variable para el estado del punto de intersección.
DEF INT MODE	; Variable para el tipo de mecanizado.
…	
MODE=1	; Cálculo con independencia del plano activo.
N10 ISPOINT=INTERSEC(TABNAME1[3],TABNAME2[7], ISCOORD,MODE)	; Llamada del punto de intersección de los elementos de contorno.
N20 IF ISPOINT==FALSE GOTOF NOCORTE	; Salto a NOCORTE.
…	

3.24.5 Retirada de los elementos de contorno de una tabla secuencia a secuencia (EXECTAB)

EXECTAB permite retirar secuencia a secuencia los elementos de contorno de una tabla creada, p. ej., con CONTPRON.

Sintaxis

```
EXECTAB(<Tabla para el contorno>[<Elemento de contorno>])
```

Descripción

EXECTAB:	Procedimiento predefinido para retirar un elemento de contorno
<Tabla para el contorno>:	Nombre de la tabla para el contorno
<Elemento de contorno>:	Número del elemento de contorno

Ejemplo

Los elementos de contorno 0 a 2 de la tabla KTAB deben retirarse secuencia a secuencia.

Código de programa	**Comentarios**
N10 EXECTAB(KTAB[0])	; Desplazar elemento 0 de la tabla KTAB.
N20 EXECTAB(KTAB[1])	; Desplazar elemento 1 de la tabla KTAB.
N30 EXECTAB(KTAB[2])	; Desplazar elemento 2 de la tabla KTAB.

3.24.6 Cálculo de los datos de circunferencia (CALCDAT)

CALCDAT permite calcular el radio y las coordenadas del centro de una circunferencia a partir de tres o cuatro puntos conocidos de esta. Los puntos indicados deben de ser diferentes.

Para el cálculo utilizando 4 puntos que no se encuentran exactamente sobre la circunferencia, se calcula un valor medio para el centro y el radio de la circunferencia.

Nota

Instrucciones para el cálculo del valor medio

Se calculan 4 arcos de circunferencia:

1. con centro 1, 2, 3
2. con centro 1, 2, 4
3. con centro 1, 3, 4
4. con centro 2, 3, 4

Los valores de abscisa y de ordenada para la coordenada del centro de la circunferencia se calculan sumando los valores de abscisas o de ordenadas de los cuatro arcos de circunferencia calculados y dividiendo entre 4.

El radio se calcula tomando la raíz cuadrada de la suma de los cuatro radios de los arcos de circunferencia calculados y multiplicando el resultado por 0,5.

Sintaxis

```
<Estado>=CALCDAT(<Puntos de
circunferencia>[<Cantidad>,<Tipo>],<Cantidad>,<Resultado>)
```

Descripción

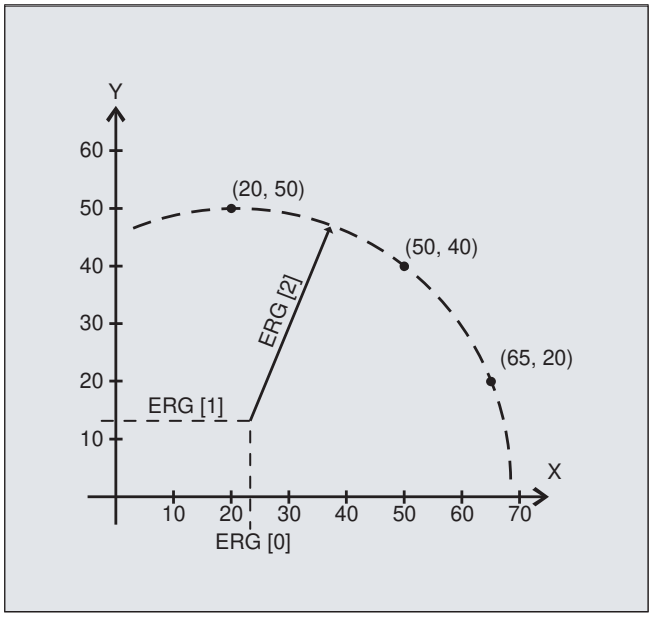
`CALCDAT`:	Función predefinida para calcular el radio y las coordenadas del centro de una circunferencia a partir de 3 o 4 puntos.		
`<Estado>`:	Variable para el estado del cálculo de circunferencia		
	Tipo:	BOOL	
	Valor:	TRUE	Los puntos indicados se encuentran en una circunferencia.
		FALSE	Los puntos indicados **no** se encuentran en una circunferencia.
`<Puntos de circunferencia>[]`:	Variable para indicar los puntos de circunferencia con los parámetros:		
	`<Cantidad>`:	Cantidad de puntos de circunferencia (3 ó 4)	
	`<Tipo>`:	Tipo del dato de coordenadas, p. ej., 2 para coordenadas de 2 puntos	
`<Cantidad>`:	Parámetro para la cantidad de puntos utilizados para el cálculo (3 ó 4)		
`<Resultado>[3]`:	Variable para el resultado: Indicación de coordenadas de centro de una circunferencia y radio		
	0	Coordenadas del centro de una circunferencia: valor de abscisas	
	1	Coordenadas del centro de una circunferencia: valor de ordenadas	
	2	Radio	

Nota

Tenga en cuenta que las variables deben estar definidas antes de su utilización.

Ejemplo

A partir de tres puntos debe determinarse si éstos se encuentran en un segmento circular.



Código de programa	Comentarios
N10 DEF REAL PKT[3,2]=(20,50,50,40,65,20)	; Variable para la indicación de puntos de circunferencia.
N20 DEF REAL RESULT[3]	; Variable para resultado.
N30 DEF BOOL STATUS	; Variable para el estado.
N40 STATUS=CALCDAT(PKT,3,RESULT)	; Llamada de los datos de circunferencia calculados.
N50 IF STATUS == FALSE GOTOF ERROR	; Salto a error.

3.24.7 Desactivación de la preparación del contorno (EXECUTE)

EXECUTE desactiva la preparación del contorno y conmuta al mismo tiempo al modo de ejecución normal de programa.

Sintaxis

EXECUTE(<ERROR>)

Descripción

EXECUTE:	Procedimiento predefinido para finalizar la preparación del contorno	
<ERROR>:	Variable para señalización de error	
	Tipo:	INT
	El valor de las variables indica si el contorno se ha preparado sin errores:	
	0	Error
	1	sin errores

Ejemplo

Código de programa

```
...
N30 CONTPRON(...)
N40 G1 X... Z...
...
N100 EXECUTE(...)
...
```

3.25 Programar ciclos externamente

3.25.1 Ciclos tecnológicos

3.25.1.1 Introducción

Índice

Este capítulo contiene la descripción de los ciclos para las tecnologías de torneado, fresado y rectificado.

Estructura

La descripción de un ciclo tiene la siguiente estructura:

- **Sintaxis**
  Nombre de ciclo y secuencia de llamada de los parámetros de transferencia
- **Parámetros**
  Tabla con explicación de cada uno de los parámetros

Descripción de los parámetros

En la tabla se indican los siguientes datos para un parámetro: nombre, descripción, rango de valores y dependencias respecto de otros parámetros.

La columna con referencias a los parámetros de la máscara sirve para encontrar fácilmente valores programados en el control al decompilar llamadas de ciclos generadas externamente.

Parámetros "sólo para la interfaz"

En la tabla, algunos parámetros están identificados con el texto "sólo para la interfaz". Estos parámetros no son relevantes para la función del ciclo. Sólo son necesarios para poder decompilar llamadas de ciclos por completo. Aunque no estén programados, el ciclo todavía puede decompilarse; en ese caso, los campos correspondientes se marcan en color y deben rellenarse en la máscara.

Parámetro "reservado"

Los parámetros descritos como "reservado" deben programarse con el valor 0 o bien dejando una coma para que los parámetros de llamada siguientes estén correctamente asignados a los parámetros de ciclo internos. Excepción: en parámetros de cadena con valor "" o donde se deja una coma.

Repetir ciclos en patrones de posiciones

Los ciclos de taladrado y fresado pueden repetirse en patrones de posiciones (llamadas modales). En ese caso, antes del ciclo debe escribirse `MCALL` en la misma línea, p. ej. `MCALL CYCLE83(...)`.

Nota

Si determinados parámetros de transferencia (p. ej., `<_VARI>`, `<_GMODE>`, `<_DMODE>`, `<_AMODE>`) están programados indirectamente como parámetros, al decompilar se abre la máscara de entrada, pero no se puede guardar porque no existe una asignación unívoca para ciertos campos de selección.

3.25.1.2 Vista general específica de tecnología

En la siguiente tabla sinóptica se incluyen todos los ciclos tecnológicos programables externamente disponibles y se asignan a su correspondiente tecnología:

Tecnología	Ciclo tecnológico
Taladrado	• CYCLE81, taladrado, punteado (Página 1102) • CYCLE82, taladrado, avellanado (Página 1103) • CYCLE85, escariado (Página 1112) • CYCLE86, mandrinar (Página 1113) • CYCLE83, taladrado profundo 1 (Página 1106) • CYCLE830, taladrado profundo 2 (Página 1140) • CYCLE84, roscado sin mandril de compensación (Página 1109) • CYCLE840, roscado con mandril de compensación (Página 1150) • CYCLE78, fresado de roscas (Página 1098) • CYCLE802, posiciones libres (Página 1137) • HOLES1, patrón de posiciones línea (Página 1063) • CYCLE801, patrón de posiciones rejilla o marco (Página 1136) • HOLES2, patrón de posiciones arco circular o arco (Página 1064)
Torneado	• CYCLE951, desbaste (Página 1160) • CYCLE930, ranurado (Página 1155) • CYCLE940, garganta forma E y F/garganta rosca (Página 1157) • CYCLE99, tallado de roscas (Página 1122) • CYCLE98, cadena de roscas (Página 1118) • CYCLE92, tronzar (Página 1115)
Torneado de contorno	• CYCLE62, llamada de contorno (Página 1083) • CYCLE952, desbastar/desbastar resto/ranurar/ranurar resto/ranurar derecha-izquierda/ranurar derecha-izquierda resto (Página 1163)

Tecnología	Ciclo tecnológico
Fresado	• CYCLE61, planear (Página 1081) • POCKET3, caja rectangular (Página 1066) • POCKET4, caja circular (Página 1068) • CYCLE76, saliente rectangular (Página 1094) • CYCLE77, saliente circular (Página 1096) • CYCLE79, poliedro (Página 1100) • SLOT1, ranura longitudinal (Página 1071) • SLOT2, ranura circular (Página 1074) • CYCLE899, ranura abierta (Página 1152) • LONGHOLE, agujero rasgado (Página 1076) • CYCLE70, fresado de roscas (Página 1088) • CYCLE60, grabado (Página 1078)
Fresado del contorno	• CYCLE62, llamada de contorno (Página 1083) • CYCLE72, fresado en contorneado (Página 1090) • CYCLE63, fresar caja de contorno/material sobrante caja de contorno/fresar saliente de contorno/material sobrante saliente de contorno (Página 1084) • CYCLE64, pretaladrar caja de contorno (Página 1086)
Rectificado	• CYCLE495, perfilar (Página 1127) • CYCLE435 - Definir el sistema de coordenadas del diamante (Página 1127) • CYCLE4071 - Rectificado longitudinal con penetración en el punto de inversión (Página 1169) • CYCLE4072 - Rectificado longitudinal con penetración en el punto de inversión y señal de cancelación (Página 1171) • CYCLE4073 - Rectificado longitudinal con penetración continua (Página 1175) • CYCLE4074 - Rectificado longitudinal con penetración continua y señal de cancelación (Página 1176) • CYCLE4075 - Rectificado plano con penetración en el punto de inversión (Página 1179) • CYCLE4077 - Rectificado plano con penetración en el punto de inversión y señal de cancelación (Página 1182) • CYCLE4078 - Rectificado plano con penetración continua (Página 1186) • CYCLE4079 - Rectificado plano con penetración intermitente (Página 1188)

Tecnología	Ciclo tecnológico
Otros	• CYCLE782: Adaptar a la carga (Página 1129) • CYCLE800, orientar plano/orientar herramienta/alinear herramienta (Página 1132) • CYCLE832, High Speed Settings (Página 1146)
Todos	• GROUP_BEGIN - Inicio bloque de programa (Página 1191) • GROUP_END - Fin bloque de programa (Página 1191) • GROUP_ADDEND - Fin adición aprox. (Página 1192)

3.25.1.3 HOLES1, patrón de posiciones línea

Sintaxis

```
HOLES1(<SPCA>, <SPCO>, <STA1>, <FDIS>, <DBH>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	X0	<SPCA>	REAL	Punto de referencia para fila de agujeros en el 1.er eje (abs)		
2	Y0	<SPCO>	REAL	Punto de referencia para fila de agujeros en el 2.º eje (abs)		
3	α0	<STA1>	REAL	Ángulo de giro básico (ángulo respecto al 1.er eje)		
4	L0	<FDIS>	REAL	Dist. entre el 1.er taladro y el punto de ref.		
5	L	<DBH>	REAL	Distancia entre los taladros		
6	N	<NUM>	INT	Número de taladros		
7		<_VARI>	INT	reservado		
8		<_UMODE>	INT	reservado		
9		<_HIDE>	STRING [200]	Posiciones ocultas • Máximo 198 caracteres • Indicación del número de posición consecutivo, p. ej., "1,3" (las posiciones 1 y 3 no se ejecutan)		
10		<_NSP>	INT	reservado		
11		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)

3.25.1.4 HOLES2, patrón de posiciones arco circular o arco

Sintaxis

```
HOLES2(<CPA>, <CPO>, <RAD>, <STA1>, <INDA>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetros interno	Tipo de datos	Descripción			
1	X0	<CPA>	REAL	Centro de agujeros en círculo en el 1.er eje (abs) Punto de referencia en el 1.er eje			(para XY) (para XA, YB, ZC)
2	Y0	<CPO>	REAL	Centro de agujeros en círculo en el 2.er eje (abs) Punto de referencia en el 2.º eje			(para XY) (para XA, YB, ZC)
3	R	<RAD>	REAL	Radio de los agujeros en círculo			(para XY)
4	α0	<STA1>	REAL	Ángulo inicial o 1.ª posición de eje giratorio			(para XY) (para XA, YB, ZC)
5	α1	<INDA>	REAL	Ángulo de incremento (sólo en el arco)			(para XY, XA, YB, ZC)
					< 0 =	Sentido horario	
					> 0 =	Sentido antihorario	
6	N	<NUM>	INT	Número de posiciones			

N.°	Máscara de parámetros	Parámetros interno	Tipo de datos	Descripción			
7		<_VARI>	INT	Tipo de mecanizado			
				UNIDADES:	reservado		
				DECENAS:	Tipo de posicionamiento		
					0 =	Desplazar a posición, linealmente	
					1 =	Desplazar a posición, en arco	
				CENTENAS:	reservado		
				MILLARES:	Patrón circular		
					0 =	Modo de compatibilidad; si INDA = 0, entonces círculo; si INDA <> 0, entonces arco	
					1 =	Círculo	
					2 =	Arco	
				DECENAS DE MILLAR:	Modelo de posición con eje giratorio		
					0 =	XY (sin eje giratorio)	(para XY)
					1 =	XA (eje X y eje giratorio alrededor de X)	(solo para XA)
					2 =	YB (eje Y y eje giratorio alrededor de Y)	(solo para YB)
					3 =	ZC (eje Z y eje giratorio alrededor de C)	(solo para ZC)
				UNIDADES DE MILLÓN + CENTENAS DE MILLAR:	Offset (para varios ejes giratorios alrededor del mismo eje; si el índice es demasiado grande, 1.er eje)		
					00 =	1.er eje A, B o C	
					01 =	2.º eje A, B o C	
					...		
					10 =	20.º eje A, B o C	
8		<_UMODE>	INT	reservado			
9		<_HIDE>	STRING [200]	reservado			
10		<_NSP>	INT	reservado			
11		<_DMODE>	INT	Modo de visualización			
				UNIDADES:	Plano de mecanizado G17/18/19		
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo	
					1 =	G17 (solo activo en el ciclo)	
					2 =	G18 (solo activo en el ciclo)	
					3 =	G19 (solo activo en el ciclo)	

3.25.1.5 POCKET3, caja rectangular

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<_RTP>	REAL	Plano de retirada (abs)		
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)		
3	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<_DP>	REAL	Profundidad de caja (abs/inc), ver <_AMODE>		
5	L	<_LENG>	REAL	Longitud de caja (inc, se introduce con signo)		
6	W	<_WID>	REAL	Ancho de caja (inc, se introduce con signo)		
7	R	<_CRAD>	REAL	Radio de esquina de la caja		
8	X0	<_PA>	REAL	Punto de referencia, 1.er eje (abs)		
9	YO	<_PO>	REAL	Punto de referencia, 2.º eje (abs)		
10	α0	<_STA>	REAL	Ángulo de giro, ángulo entre el eje longitudinal (L) y el 1.er eje		
11	DZ	<_MID>	REAL	Penetración máx. en profundidad		
12	UXY	<_FAL>	REAL	Creces para acabado en el plano		
13	UZ	<_FALD>	REAL	Creces para acabado profundidad		
14	F	<_FFP1>	REAL	Avance en el plano		
15	FZ	<_FFD>	REAL	Avance de penetración profundidad		
16		<_CDIR>	INT	Sentido de fresado:	0 =	En concordancia
					1 =	En oposición
17		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:		
					1 =	Desbaste
					2 =	Acabado
					4 =	Acabado borde
					5 =	Achaflanado
				DECENAS:		
					0 =	Pretaladrado, penetración con G0
					1 =	Perpendicular, penetración con G1
					2 =	Helicoidal
					3 =	Movimiento oscilante en el eje longitudinal de la caja
				CENTENAS:	reservado	

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
18	DXY	<_MIDA>	REAL	Penetración máxima del plano; unidad, ver <_AMODE>		
19	L1	<_AP1>	REAL	Longitud del mecanizado previo (inc)		
20	W1	<_AP2>	REAL	Anchura del mecanizado previo (inc)		
21	AZ	<_AD>	REAL	Profundidad del mecanizado previo (inc)		
22	ER	<_RAD1>	REAL	Radio de la trayectoria helicoidal en la penetración helicoidal		
	EW			Máximo ángulo de penetración para movimiento oscilante		
23	EP	<_DP1>	REAL	Paso de hélice en la penetración helicoidal		
24		<_UMODE>	INT	reservado		
25	FS	<_FS>	REAL	Ancho del chaflán (inc)		
26	ZFS	<_ZFS>	REAL	Profundidad de penetración (punta de la herramienta) en el achaflanado (abs/inc), ver <_AMODE>		
27		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	selección del mecanizado/sólo cálculo del punto inicial	
					0 =	Modo de compatibilidad
					1 =	Mecanizado normal
				MILLARES:	acotado a través de centro/esquina	
					0 =	Modo de compatibilidad
					1 =	Acotado a través de centro
					2 =	Acotado esquina, posición de la caja +LENG/+WID
					3 =	Acotado esquina, posición de la caja -LENG/+WID
					4 =	Acotado esquina, posición de la caja +LENG/-WID
					5 =	Acotado esquina, posición de la caja -LENG/-WID
				DECENAS DE MILLAR:	Mecanizado completo/repaso	
					0 =	Modo de compatibilidad (tratar <_AP1>, <_AP2> y <_AD> como hasta ahora)
					1 =	Mecanizado completo
					2 =	Repaso

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
28		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	Tipo de avance: grupo G (G94/G95) para avance superficial y avance en profundidad	
					0 =	Modo de compatibilidad
					1 =	Comando G como antes de la llamada de ciclo. G94/G95 iguales para avance superficial y avance en profundidad
				CENTENAS:	---	reservado
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple
29		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad de la caja (Z1)	
					0 =	Absoluta (modo de compatibilidad)
					1 =	Incremental
				DECENAS:	unidad para la penetración del plano (DXY)	
					0 =	mm
					1 =	% del diámetro de la herramienta
				CENTENAS:	Profundidad de penetración en el achaflanado (ZFS)	
					0 =	Absoluta
					1 =	Incremental

3.25.1.6 POCKET4, caja circular

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<_RTP>	REAL	Plano de retirada (abs)		
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)		
3	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<_DP>	REAL	Profundidad de caja (abs/inc), ver <_AMODE>		
5	∅	<_CDIAM>	REAL	Diámetro o radio de la caja, ver <_DMODE>		
6	X0	<_PA>	REAL	Punto de referencia 1.er eje (abs)		
7	Y0	<_PO>	REAL	Punto de referencia 2.º eje (abs)		
8	DZ	<_MID>	REAL	Penetración máxima en profundidad, ver <_VARI> = por planos Paso helicoidal máximo, ver <_VARI> = helicoidal		
9	UXY	<_FAL>	REAL	Creces para acabado en el plano		
10	UZ	<_FALD>	REAL	Creces para acabado profundidad		
11	F	<_FFP1>	REAL	Avance para mecanizado de superficie		
12	FZ	<_FFD>	REAL	Avance de penetración profundidad		
13		<_CDIR>	INT	Sentido de fresado	0 =	En concordancia
					1 =	En oposición
14		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	Mecanizado	
					1 =	Desbaste
					2 =	Acabado
					4 =	Acabado borde
					5 =	Achaflanado
				DECENAS:	clase de penetración (desbaste y acabado)	
					0 =	Pretaladrado, penetración con G0 (caja preelaborada)
					1 =	Perpendicular, penetración con G1
					2 =	Helicoidal
				CENTENAS:	reservado	
				MILLARES:		
					0 =	Por planos
					1 =	Helicoidal
15	DXY	<_MIDA>	REAL	Penetración máxima del plano, ver <_AMODE>, 0 = 0,8 x diámetro HERR		
16	∅	<_AP1>	REAL	Diámetro/radio del mecanizado previo (inc)		
17	AZ	<_AD>	REAL	Profundidad del mecanizado previo (inc)		
18	ER	<_RAD1>	REAL	Radio de la trayectoria helicoidal en la penetración helicoidal		
19	EP	<_DP1>	REAL	Paso de hélice en la penetración en la trayectoria helicoidal		
20		<_UMODE>	INT	reservado		
21	FS	<_FS>	REAL	Ancho del chaflán (inc)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
22	ZFS	<_ZFS>	REAL	Profundidad de penetración (punta de la herramienta) en el achaflanado (abs/inc), ver <_AMODE>		
23		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	mecanizado/cálculo del punto inicial	
					0 =	Modo de compatibilidad
					1 =	Mecanizado normal
				MILLARES:	reservado	
				DECENAS DE MILLAR:	Mecanizado completo/repaso	
					0 =	Modo de compatibilidad (tratar <_AP1> y <_AD> como hasta ahora)
					1 =	Mecanizado completo
					2 =	Repaso
24		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	Tipo de avance: grupo G (G94/G95) para avance superficial y avance en profundidad	
					0 =	Modo de compatibilidad
					1 =	Comando G como antes de la llamada de ciclo. G94/G95 iguales para avance superficial y avance en profundidad
				CENTENAS:		
					0 =	Modo de compatibilidad (<_CDIAM>/<_AP1> introducido como radio)
					1 =	<_CDIAM>/<_AP1> introducido como diámetro
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
25		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad de la caja (Z1)	
					0 =	Absoluta (modo de compatibilidad)
					1 =	Incremental
				DECENAS:	unidad para el ancho de penetración (DXY)	
					0 =	mm
					1 =	% del diámetro de la herramienta
				CENTENAS:	Profundidad de penetración en el achaflanado (ZFS)	
					0 =	Absoluta
					1 =	Incremental

3.25.1.7 SLOT1, ranura longitudinal

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	RP	<RTP>	REAL	Plano de retirada (abs)
2	Z0	<RFP>	REAL	Punto de referencia del eje de herramienta (abs)
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)
4	Z1	<_DP>	REAL	Profundidad de ranura (abs)
5		<_DPR>	REAL	Profundidad de ranura (inc), referida a Z0 (se introduce sin signo)
6		<NUM>	INT	Número de ranuras = 1
7	L	<LENG>	REAL	Longitud de la ranura
8	W	<WID>	REAL	Anchura de la ranura
9	X0	<_CPA>	REAL	Punto de referencia, 1.er eje del plano
10	Y0	<_CPO>	REAL	Punto de referencia, 2.er eje del plano
11		<RAD>	REAL	reservado
12	α	<STA1>	REAL	Ángulo de giro
13		<INDA>	REAL	reservado
14	FZ	<FFD>	REAL	Avance de penetración profundidad
15	F	<FFP1>	REAL	Avance

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
16	DZ	<_MID>	REAL	Penetración máx. en profundidad		
17		<CDIR>	INT	Sentido de fresado	0 =	En concordancia
					1 =	En oposición
18	UXY	<_FAL>	REAL	Creces para acabado plano o borde de la ranura		
19		<VARI>	INT	Tipo de mecanizado		
				UNIDADES:		
					0 =	reservado
					1 =	Desbaste
					2 =	Acabado
					4 =	Acabado borde (mecanizar sólo el borde)
					5 =	Achaflanado
				DECENAS:	aproximación	
					0 =	Pretaladrado, penetración con G0 (ranura premecanizada)
					1 =	Perpendicular, penetración con G1
					2 =	Helicoidal
					3 =	Oscilante
				CENTENAS:	reservado	
20	DZF	<_MIDF>	REAL	reservado		
21	FF	<FFP2>	REAL	reservado		
22	SF	<SSF>	REAL	reservado		
23	UZ	<_FALD>	REAL	Creces para acabado profundidad		
24	ER	<_STA2>	REAL	Radio de la trayectoria helicoidal en la penetración helicoidal		
	EW			Máximo ángulo de penetración para movimiento oscilante		
25	EP	<_DP1>	REAL	Profundidad de penetración por vuelta para hélice		
26		<_UMODE>	INT	reservado		
27	FS	<_FS>	REAL	Ancho del chaflán (inc) en achaflanado		
28	ZFS	<_ZFS>	REAL	Profundidad de penetración (punta de la herramienta) en el achaflanado (abs/inc), ver <_AMODE>		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
29		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	Selección del mecanizado o solo cálculo del punto inicial	
					1 =	Mecanizado normal
				MILLARES:	acotado del punto de referencia, posición de la ranura	
					0 =	Centro
					1 =	Izquierda, interior +L
					2 =	Derecha, interior -L
					3 =	Borde izquierdo +L
					4 =	Borde derecho -L
30		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:	Identificación de la versión de software	
					1 =	Ampliación de función SLOT1
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple
31		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad final Z1 (abs/inc)	
					0 =	Compatibilidad
					1 =	Z1 (inc)
					2 =	Z1 (abs)
				DECENAS:	reservado	
				CENTENAS:	Profundidad de penetración en el achaflanado ZFS	
					0 =	ZFS (abs)
					1 =	ZFS (inc)

Nota

A diferencia de las versiones de SW anteriores, el ciclo está equipado con nuevas funciones. En consecuencia, ya no se muestran determinados parámetros en la máscara de entrada (<NUM>, <RAD>, <INDA>). Se pueden programar varias ranuras en un patrón de posiciones mediante "MCALL" y llamando el patrón de posiciones deseado, p. ej. HOLES2.

3.25.1.8 SLOT2, ranura circular

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<RTP>	REAL	Plano de retirada (abs)		
2	Z0	<RFP>	REAL	Punto de referencia del eje de herramienta (abs)		
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<_DP>	REAL	Profundidad de ranura (abs)		
5		<_DPR>	REAL	Profundidad de ranura (inc), referida a Z0 (se introduce sin signo)		
6	N	<NUM>	INT	Número de ranuras		
7	α1	<AFSL>	REAL	Ángulo en el vértice de la ranura		
8	W	<WID>	REAL	Anchura de la ranura		
9	X0	<_CPA>	REAL	Punto de referencia = centro del círculo, 1.er eje del plano		
10	Y0	<_CPO>	REAL	Punto de referencia = centro del círculo, 2.er eje del plano		
11	R	<RAD>	REAL	Radio del círculo		
12	α0	<STA1>	REAL	Ángulo inicial		
13	α2	<INDA>	REAL	Ángulo de incremento		
14	FZ	<FFD>	REAL	Avance de penetración profundidad		
15	F	<FFP1>	REAL	Avance		
16	DZ	<_MID>	REAL	Penetración máx. en profundidad		
17		<CDIR>	INT	Sentido de fresado	0 =	En concordancia
					1 =	En oposición
18	UXY	<_FAL>	REAL	Creces para acabado plano o borde de la ranura		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
19		<VARI>	INT	Tipo de mecanizado		
				UNIDADES:		
					0 =	Mecanizado completo
					1 =	Desbaste
					2 =	Acabado
					3 =	Acabado borde
					5 =	Achaflanado
				DECENAS:		
					0 =	Posicionamiento intermedio con recta G0
					1 =	Posicionamiento intermedio en arco
				CENTENAS:	reservado	
				MILLARES:		
					0 =	Modo de compatibilidad; si <INDA> = 0, entonces círculo; si <INDA> **<>** 0, entonces arco
					1 =	Círculo
					2 =	Arco
20	DZF	<_MIDF>	REAL	reservado		
21		<FFP2>	REAL	reservado		
22		<SSF>	REAL	reservado		
23	FF	<_FFCP>	REAL	reservado		
24		<_UMODE>	INT	reservado		
25	FS	<_FS>	REAL	Ancho del chaflán (inc)		
26	ZFS	<_ZFS>	REAL	Profundidad de penetración (punta de la herramienta) en el achaflanado (abs/inc), ver <_AMODE>		
27		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	Selección del mecanizado o solo cálculo del punto inicial	
					0 =	Modo de compatibilidad
					1 =	Mecanizado normal

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
28		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:	Identificación de la versión de software	
					1 =	Funciones SLOT2 a partir de la versión de SW 2.5
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple
29		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad final Z1 (abs/inc)	
					0 =	Compatibilidad
					1 =	Z1 (inc)
					2 =	Z1 (abs)
				DECENAS:	reservado	
				CENTENAS:	Profundidad de penetración en el achaflanado ZFS	
					0 =	ZFS (abs)
					1 =	ZFS (inc)

3.25.1.9 LONGHOLE, agujero rasgado

Sintaxis

```
LONGHOLE(<RTP>, <RFP>, <SDIS>, <_DP>, <_DPR>, <NUM>, <LENG>, <_CPA>,
<_CPO>, <RAD>, <STA1>, <INDA>, <FFD>, <FFP1>, <MID>, <_VARI>,
<_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	RP	<RTP>	REAL	Plano de retirada (abs)
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<_DP>	REAL	Profundidad de agujero rasgado (abs)		
5		<_DPR>	REAL	Profundidad de agujero rasgado (inc), referida a Z0 (se introduce sin signo)		
6		<NUM>	INT	Número de agujeros rasgados = 1		
7	L	<LENG>	REAL	Longitud del agujero rasgado		
8	X0	<_CPA>	REAL	Punto de referencia, 1.er eje del plano		
9	Y0	<_CPO>	REAL	Punto de referencia, 2.º eje del plano		
10		<RAD>	REAL	reservado		
11	α0	<STA1>	REAL	Ángulo de giro		
12		<INDA>	REAL	reservado		
13	FZ	<FFD>	REAL	Avance de penetración profundidad		
14	F	<FFP1>	REAL	Avance		
15	DZ	<MID>	REAL	Penetración máx. en profundidad		
16		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	tipo de penetración	
					1 =	perpendicular con G1
					3 =	oscilante
				CENTENAS:	reservado	
17		<_UMODE>	INT	reservado		
18		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	selección del mecanizado o solo cálculo del punto inicial	
					0 =	modo de compatibilidad
					1 =	mecanizado normal
				MILLARES:	acotado del punto de referencia, posición de la ranura	
					0 =	centro
					1 =	izquierda, interior +L
					2 =	derecha, interior -L
					3 =	borde izquierdo +L
					4 =	borde derecho -L

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
19		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	Tipo de avance: grupo G (G94/G95) para avance superficial y avance en profundidad	
					0 =	modo de compatibilidad
					1 =	Comando G como antes de la llamada de ciclo. G94/G95 iguales para avance superficial y avance en profundidad
				CENTENAS:	reservado	
				MILLARES:	identificación de la versión de software	
					1 =	ampliación de función LONGHOLE (acotado del punto de referencia)
20		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad final Z1 (abs/inc)	
					0 =	Compatibilidad
					1 =	Z1 (inc)
					2 =	Z1 (abs)

Nota

A diferencia de las versiones de SW anteriores, el ciclo está equipado con nuevas funciones. En consecuencia, ya no se muestran determinados parámetros en la máscara de entrada (<NUM>, <RAD>, <INDA>). Se pueden programar varias ranuras en un patrón de posiciones mediante "MCALL" y llamando el patrón de posiciones deseado, p. ej., HOLES2.

3.25.1.10 CYCLE60, grabado

Sintaxis

```
CYCLE60(<_TEXT>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_PA>,
<_PO>, <_STA>, <_CP1>, <_CP2>, <_WID>, <_DF>, <_FFD>, <_FFP1>,
<_VARI>, <_CODEP>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1		<_TEXT>	STRING [200]	Texto a grabar (máx. 100 caracteres)
2	RP	<_RTP>	REAL	Plano de retirada (abs)
3	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)
4	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al plano de referencia, se introduce sin signo)
5	Z1	<_DP>	REAL	Profundidad (abs), ver <_AMODE>
6	Z1	<_DPR>	REAL	Profundidad (inc), ver <_AMODE>
7	X0	<_PA>	REAL	Punto de referencia, 1.er Eje del plano (abs), perpendicular, ver <_VARI>
	R			Punto de referencia, longitud (radio), polar, ver <_VARI>
8	Y0	<_PO>	REAL	Punto de referencia, 2.º Eje del plano (abs), perpendicular, ver <_VARI>
	α0			Punto de referencia, ángulo referido al 1.er eje, polar, ver <_VARI>
9	α1	<_STA>	REAL	Dirección del texto, ángulo de la línea de texto referido al 1.er eje), ver <_VARI>
10	XM	<_CP1>	REAL	Centro del círculo de texto, 1.er eje del plano (abs), perpendicular, ver <_VARI>
	LM			Centro del círculo de texto, longitud (radio) referida a WNP, polar, ver <_VARI>
11	YM	<_CP2>	REAL	Centro del círculo de texto, 2.º eje del plano (abs), perpendicular, ver <_VARI>
	αM			Centro del círculo de texto, ángulo referido al 1.er eje, polar, ver <_VARI>
12	W	<_WID>	REAL	Altura de los caracteres (se introduce sin signo)
13	DX1 DX2	<_DF>	REAL	Distancia entre caracteres/ancho total, ver <_VARI>
	α2			Ángulo en el vértice, ver <_VARI>
14	FZ	<_FFD>	REAL	Avance de penetración profundidad, ver <_DMODE>
15	F	<_FFP1>	REAL	Avance para mecanizado de superficie

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
16		<_VARI>	INT	Mecanizado (alineación y punto de referencia del texto grabado)		
				UNIDADES:	Punto de referencia	
					0 =	perpendicular
					1 =	Polar
				DECENAS:	alineación del texto	
					0 =	texto en una línea
					1 =	texto en un arco circunferencial arriba
					2 =	texto en un arco circunferencial abajo
				CENTENAS:	reservado	
				MILLARES:	punto de referencia del texto horizontal	
					0 =	izquierda
					1 =	Centrado
					2 =	derecha
				DECENAS DE MILLAR:	punto de referencia del texto vertical	
					0 =	abajo
					1 =	Centrado
					2 =	arriba
				CENTENAS DE MILLAR:	longitud del texto	
					0 =	distancia entre caracteres
					1 =	ancho total del texto (sólo con texto lineal)
					2 =	ángulo en el vértice (sólo con texto en un arco)
				UNIDADES DE MILLÓN:	centro del círculo	
					0 =	perpendicular (cartesiano)
					1 =	Polar
				DECENAS DE MILLÓN:	escritura especular	
					0 =	Compatibilidad
					1 =	Escritura especular CON
					2 =	Escritura especular DES
17		<_CODEP>	INT	Número de la página de código para la letra (por el momento sólo 1252)		
18		<_UMODE>	INT	reservado		
19		_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	selección del mecanizado/sólo cálculo del punto inicial	
					0 =	Modo de compatibilidad
					1 =	Mecanizado normal

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
20		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	Tipo de avance: grupo G (G94/G95) para avance superficial y avance en profundidad	
					0 =	Modo de compatibilidad
					1 =	Comando G como antes de la llamada de ciclo. G94/G95 iguales para avance superficial y avance en profundidad
21		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad final (<_DP>, <_DPR>)	
					0 =	Compatibilidad
					1 =	Incremental (<_DPR>)
					2 =	Absoluta (<_DP>)

3.25.1.11 CYCLE61, planear

Sintaxis

```
CYCLE61(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_PA>, <_PO>, <_LENG>,
<_WID>, <_MID>, <_MIDA>, <_FALD>, <_FFP1>, <_VARI>, <_LIM>,
<_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	RP	<_RTP>	REAL	Plano de retirada (abs)
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta, altura de la pieza en bruto (abs)
3	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)
4	Z1	<_DP>	REAL	Altura de la pieza acabada (abs/inc), ver <_AMODE>
5	X0	<_PA>	REAL	Esquina 1 en el 1.er eje (abs)
6	Y0	<_PO>	REAL	Esquina 2 en el 1.er eje (abs)
7	X1	<_LENG>	REAL	Esquina 2 en el 1.er eje (abs/inc), ver <_AMODE>
8	Y1	<_WID>	REAL	Esquina 2 en el 2.º eje (abs/inc), ver <_AMODE>

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
9	DZ	<_MID>	REAL	Penetración máx. en profundidad		
10	DXY	<_MIDA>	REAL	Penetración máxima del plano (unidad, ver <_AMODE>)		
11	UZ	<_FALD>	REAL	Creces para acabado profundidad		
12	F	<_FFP1>	REAL	Avance de mecanizado		
13		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	Mecanizado	
					1 =	Desbaste
					2 =	Acabado
				DECENAS:	Sentido de mecanizado	
					1 =	Paralelo al 1.er eje, un sentido
					2 =	Paralelo al 2.º eje, un sentido
					3 =	Paralelo al 1.er eje, sentido cambiante
					4 =	Paralelo al 2.º eje, sentido cambiante
14		<_LIM>	INT	Limitaciones		
				UNIDADES:	limitación 1.er eje neg.	
					0 =	no
					1 =	sí
				DECENAS:	Limitación 1.er eje pos.	
					0 =	no
					1 =	sí
				CENTENAS:	limitación 2.er eje neg.	
					0 =	no
					1 =	sí
				MILLARES:	Limitación 2.er eje pos.	
					0 =	no
					1 =	sí
15		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
16		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad final (<_DP>)	
					0 =	Absoluta
					1 =	Incremental
				DECENAS:	Unidad para la penetración del plano (<_MIDA>)	
					0 =	mm
					1 =	% del diámetro de la herramienta
				CENTENAS:	reservado	
				MILLARES:	longitud de la superficie	
					0 =	Incremental
					1 =	Absoluta
				DECENAS DE MILLAR:	anchura de la superficie	
					0 =	Incremental
					1 =	Absoluta

3.25.1.12 CYCLE62, llamada de contorno

Sintaxis

```
CYCLE62(<_KNAME>,<_TYPE>, <_LAB1>, <_LAB2>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	PRG/CON	<_KNAME>	STRING [140]	Nombre de contorno o nombre de subprograma, no es necesario programarlo con _TYPE = 2		
2		<_TYPE>	INT	Determinación de la introducción del contorno		
					0 =	Subprograma
					1 =	Nombre de contorno
					2 =	Labels
					3 =	Labels en subprograma
3	LAB1	<_LAB1>	STRING[32]	Label 1, inicio del contorno		
4	LAB2	<_LAB2>	STRING[32]	Label 2, final del contorno		

3.25.1.13 CYCLE63, fresar caja de contorno/material sobrante caja de contorno/fresar saliente de contorno/material sobrante saliente de contorno

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	PRG	<_PRG>	STRING [100]	Nombre del programa de vaciado		
2		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	mecanizado tecnológico	
					1 =	Desbaste
					3 =	Acabado fondo
					4 =	Acabado borde
					5 =	Achaflanado
				DECENAS:	clase de penetración	
					0 =	Penetración central
					1 =	Penetración helicoidal
					2 =	Penetración oscilante
				CENTENAS:	reservado	
				MILLARES:	Modo de retirada	
					0 =	Retirada al plano de retirada
					1 =	Retirada al punto de referencia + distancia de seguridad
				DECENAS DE MILLAR:	punto inicial en desbaste y acabado de fondo	
					0 =	Auto
					1 =	Manual
3	RP	<_RP>	REAL	Plano de retirada (abs)		
4	Z0	<_Z0>	REAL	Punto de referencia del eje de herramienta (abs)		
5	SC	<_SC>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
6	Z1	<_Z1>	REAL	Profundidad final (ver <_AMODE> UNIDADES)		
7	F	<_F>	REAL	Avance en el plano durante el desbaste/acabado		
8	FZ	<_FZ>	REAL	Avance de penetración profundidad		
9	DXY	<_DXY>	REAL	Penetración en el plano, unidad (ver <_AMODE> DECENAS)		
10	DZ	<_DZ>	REAL	Penetración profundidad		
11	UXY	<_UXY>	REAL	Creces para acabado en el plano		
12	UZ	<_UZ>	REAL	Creces para acabado profundidad		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
13		<_CDIR>	INT	Sentido de fresado	0 =	En concordancia
					1 =	En oposición
14	XS	<_XS>	REAL	Punto inicial X, absoluto		
15	YS	<_YS>	REAL	Punto inicial Y, absoluto		
16	ER	<_ER>	REAL	Penetración helicoidal: radio		
17	EP	<_EP>	REAL	Penetración helicoidal: Paso		
18	EW	<_EW>	REAL	Penetración oscilante: ángulo máximo de penetración		
19	FS	<_FS>	REAL	Ancho del chaflán (inc) en achaflanado		
20	ZFS	<_ZFS>	REAL	Profundidad de penetración de la punta de la herramienta en el achaflanado (ver <_AMODE> CENTENAS)		
21	TR	<_TR>	STRING[32]	Nombre de la herramienta de referencia con mecanizado de material sobrante		
22	DR	<_DR>	INT	Número D de la herramienta de referencia con mecanizado de material sobrante		
23		<_UMODE>	INT	reservado		
24		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	Selección del mecanizado/sólo cálculo del punto inicial	
					0 =	Mecanizado normal (no se necesita el modo de compatibilidad)
					1 =	Mecanizado normal
					2 =	reservado

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
25		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	reservado	
				CENTENAS:	modo tecnológico	
					1 =	Caja
					2 =	Saliente
				MILLARES:	mecanizar material sobrante	
					0 =	no
					1 =	sí
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple
				CENTENAS DE MILLAR:	Nombre de programa automático	
					0 =	no
					1 =	sí
26		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad final (Z1)	
					0 =	Absoluta (modo de compatibilidad)
					1 =	Incremental
				DECENAS:	Unidad para la penetración del plano (DXY)	
					0 =	mm
					1 =	% del diámetro de la herramienta
				CENTENAS:	Profundidad de penetración en el achaflanado (ZFS)	
					0 =	Absoluto
					1 =	Incremental
				MILLARES:	---	reservado

3.25.1.14 CYCLE64, pretaladrar caja de contorno

Sintaxis

```
CYCLE64(<_PRG>, <_VARI>, <_RP>, <_Z0>, <_SC>, <_Z1>, <_F>, <_DXY>,
<_UXY>, <_UZ>, <_CDIR>, <_TR>, <_DR>, <_UMODE>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	PRG	<_PRG>	STRING [100]	Nombre del programa de taladrado/punteado		
2		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:	Modo de retirada	
					0 =	Retirada al plano de retirada
					1 =	Retirada al punto de referencia + distancia de seguridad
3	RP	<_RP>	REAL	Plano de retirada (abs)		
4	Z0	<_Z0>	REAL	Punto de referencia (abs)		
5	SC	<_SC>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
6	Z1	<_Z1>	REAL	Profundidad de taladrado/punteado (ver <_AMODE> UNIDADES)		
7	F	<_F>	REAL	Avance de taladrado/punteado		
8	DXY	<_DXY>	REAL	Penetración en el plano, unidad (ver <_AMODE> DECENAS)		
9	UXY	<_UXY>	REAL	Creces para acabado en el plano		
10	UZ	<_UZ>	REAL	Creces para acabado profundidad		
11		<_CDIR>	INT	Sentido de fresado	0 =	En concordancia
					1 =	En oposición
12	TR	<_TR>	STRING[20]	Nombre de la herramienta de referencia		
13	DR	<_DR>	INT	Número D de la herramienta de referencia		
14		<_UMODE>	INT	reservado		
15		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	Selección del mecanizado/sólo cálculo del punto inicial	
					0 =	Mecanizado normal (no se necesita el modo de compatibilidad)
					1 =	Mecanizado normal
					2 =	reservado

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
25		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	modo tecnológico	
					1 =	Pretaladrado
					2 =	Puntear
				CENTENAS:	---	reservado
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	---	reservado
				CENTENAS DE MILLAR:	Nombre de programa automático	
					0 =	no
					1 =	sí
26		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad de taladrado/punteado Z1	
					0 =	Absoluta (modo de compatibilidad)
					1 =	Incremental
				DECENAS:	Unidad para la penetración del plano (DXY)	
					0 =	mm
					1 =	% del diámetro de la herramienta

3.25.1.15 CYCLE70, fresado de roscas

Sintaxis

```
CYCLE70(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DIATH>, <_H1>, <_FAL>,
<_PIT>, <_NT>, <_MID>, <_FFR>, <_TYPTH>, <_PA>, <_PO>, <_NSP>,
<_VARI>, <_PITA>, <_PITM>, <_PTAB>, <_PTABA>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	RP	<_RTP>	REAL	Plano de retirada (abs)
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
3	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<_DP>	REAL	Longitud de la rosca (abs, inc), ver <_AMODE> Tener en cuenta la salida en el fondo del taladro (mínimo: medio paso)		
5	∅	<_DIATH>	REAL	Diámetro nominal de la rosca		
6	H1	<_H1>	REAL	Profundidad de rosca		
7	U	<_FAL>	REAL	Creces para acabado		
8	P	<_PIT>	REAL	Paso de rosca (selección de <_PITA>: mm, pulgadas, MÓDULO, filetes/pulgada)		
9	NT	<_NT>	INT	Número de dientes en la plaquita de corte ¡Longitud de herramienta referida siempre al diente inferior!		
10	DXY	<_MID>	REAL	Penetración máxima por corte <_MID> **>** <_H1>: todo con un corte		
11	F	<_FFR>	REAL	Avance de fresado		
12		<_TYPTH>	INT	Tipo de rosca	0 =	Rosca interior
					1 =	Rosca exterior
13	X0	<_PA>	REAL	Centro del círculo, 1.er eje (abs)		
14	Y0	<_PO>	REAL	Centro del círculo, 2.er eje (abs)		
15	αS	<_NSP>	REAL	Ángulo inicial (rosca de varias entradas)		
16		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:		
					1 =	Desbaste
					2 =	Acabado
				DECENAS:		
					1 =	De arriba hacia abajo
					2 =	De abajo hacia arriba
				CENTENAS:		
					0 =	Rosca a derechas
					1 =	Rosca a izquierdas
17		<_PITA>	INT	Evaluación del paso de rosca		
					0 =	Modo de compatibilidad
					1 =	Paso en mm
					2 =	Paso en filetes por pulgada (TPI)
					3 =	Paso en pulgadas
					4 =	Paso como MÓDULO
18		<_PITM>	STRING[15]	Cadena de caracteres como marca para la introducción del paso de rosca (sólo para la interfaz)		
19		<_PTAB>	STRING[20]	Cadena de caracteres para la tabla de roscas ("", "ISO", "BSW", "BSP", "UNC") (sólo para la interfaz)		
20		<_PTABA>	STRING[20]	Cadena de caracteres para la selección en la tabla de roscas (p. ej. "M 10", "M 12"...) (sólo para la interfaz)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
21		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	mecanizado/cálculo del punto inicial	
					0 =	Modo de compatibilidad
					1 =	Mecanizado normal
22		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
23		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Longitud de la rosca (<_DP>)	
					0 =	Absoluta
					1 =	Incremental

3.25.1.16 CYCLE72, fresado en contorneado

Sintaxis

```
CYCLE72(<_KNAME>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_MID>, <_FAL>,
<_FALD>, <_FFP1>, <_FFD>, <_VARI>, <_RL>, <_AS1>, <_LP1>, <_FF3>,
<_AS2>, <_LP2>,<_UMODE>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1		<_KNAME>	STRING [141]	Nombre del subprograma del contorno
2	RP	<_RTP>	REAL	Plano de retirada (abs)
3	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)
4	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)
5	Z1	<_DP>	REAL	Punto final, profundidad final (abs/inc), ver <_AMODE>
6	DZ	<_MID>	REAL	Máx. penetración en profundidad (inc, se introduce sin signo)
7	UXY	<_FAL>	REAL	Creces para acabado en el plano (inc), creces en el contorno del borde
8	UZ	<_FALD>	REAL	Creces para acabado en profundidad (inc), creces en el fondo (se introduce sin signo)
9	FX	<_FFP1>	REAL	Avance en el contorno

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
10	FZ	<_FFD>	REAL	Avance para penetración en profundidad (o penetración espacial)		
11		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	Mecanizado	
					1 =	Desbaste
					2 =	Acabado
					5 =	Achaflanado
				DECENAS:		
					0 =	Recorridos intermedios con G0
					1 =	Recorridos intermedios con G1
				CENTENAS:	Retirada al final del contorno	
					0 =	Retirada al final del contorno hasta punto de referencia
					1 =	Retirada al final del contorno hasta punto de referencia + <_SDIS>
					2 =	Retirada al final del contorno en <_SDIS>
					3 =	Sin retirada al final del contorno; el siguiente punto inicial se alcanzará con el avance de contorno
				MILLARES:	reservado	
				DECENAS DE MILLAR:	Mecanizar contorno	
					0 =	Mecanizar contorno hacia delante
					1 =	Mecanizar contorno hacia atrás Limitaciones con marcha hacia atrás: • máx. 170 elementos de contorno (incluidos chaflanes o redondeos) • sólo se evalúan valores en el plano (X/Y) y F
12		<_RL>	INT	Sentido de mecanizado		
					40 =	Centro del contorno (G40, aproximación y retirada: recta o vertical)
					41 =	Izquierda del contorno (G41, aproximación y retirada: recta o arco)
					42 =	Derecha del contorno (G42, aproximación y retirada: recta o arco)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
13		<_AS1>	INT	Desplazamiento de aproximación del contorno		
				UNIDADES:		
					1 =	Recta
					2 =	Cuadrante
					3 =	Semicírculo
					4 =	Aproximación y retirada vertical
				DECENAS:		
					0 =	Último movimiento, en el plano
					1 =	Último movimiento, espacial
14	L1	<_LP1>	REAL	Recorrido de aproximación o radio de aproximación (inc, se introduce sin signo)		
15	FZ	<_FF3>	REAL	Avance para recorridos intermedios (G94/G95 como en el contorno)		
16		<_AS2>	INT	Movimiento de retirada de contorno (no con aproximación/retirada vertical)		
				UNIDADES:		
					1 =	Recta
					2 =	Cuadrante
					3 =	Semicírculo
				DECENAS:		
					0 =	Último movimiento, en el plano
					1 =	Último movimiento, espacial
17	L2	<_LP2>	REAL	Recorrido de retirada o radio de retirada (inc, se introduce sin signo)		
18		<_UMODE>	INT	reservado		
19	FS	<_FS>	REAL	Ancho del chaflán (inc)		
20	ZFS	<_ZFS>	REAL	Profundidad de penetración (punta de la herramienta) en el achaflanado (abs/inc), ver <_AMODE>		
21		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	Selección del mecanizado/sólo cálculo del punto inicial	
					0 =	Modo de compatibilidad
					1 =	Mecanizado normal

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
22		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	Tipo de avance: grupo G (G94/G95) para avance superficial y avance en profundidad	
					0 =	Modo de compatibilidad
					1 =	Comando G como antes de la llamada de ciclo. G94/G95 iguales para avance superficial y avance en profundidad
				MILLARES:		
					0 =	Modo de compatibilidad: el nombre de contorno figura en <_KNAME>
					1 =	El nombre de contorno se programa en CYCLE62 y se transfiere a _SC_CONT_NAME
23		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Punto final Z1 (<_DP>)	
					0 =	Absoluto (modo de compatibilidad)
					1 =	Incremental
				DECENAS:	unidad para la penetración del plano	
					0 =	mm, pulgadas
					1 =	reservado
				CENTENAS:	Profundidad de penetración en el achaflanado (<_ZFS>)	
					0 =	Absoluta
					1 =	Incremental

Nota

Si los siguientes parámetros de transferencia se han programado indirectamente (como parámetros), la máscara de entrada no se decompila.

<_VARI>, <_RL>, <_AS1>, <_AS2>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>

3.25.1.17 CYCLE76, saliente rectangular

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<_RTP>	REAL	Plano de retirada (abs)		
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)		
3	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<_DP>	REAL	Profundidad del saliente (abs)		
5		<_DPR>	REAL	Profundidad del saliente (inc), referida a Z0 (se introduce sin signo)		
6	L	<_LENG>	REAL	Longitud del saliente, ver <_GMODE> (se introduce sin signo)		
7	W	<_WID>	REAL	Anchura del saliente, ver <_GMODE> (se introduce sin signo)		
8	R	<_CRAD>	REAL	Radio de esquina del saliente (se introduce sin signo)		
9	X0	<_PA>	REAL	Punto de referencia del saliente, 1.er eje del plano (abs)		
10	Y0	<_PO>	REAL	Punto de referencia del saliente, 2.er eje del plano (abs)		
11	α0	<_STA>	REAL	Ángulo de giro, ángulo entre el eje longitudinal (L) y el 1.er eje del plano		
12	DZ	<_MID>	REAL	Máx. penetración en profundidad (inc, se introduce sin signo)		
13	UXY	<_FAL>	REAL	Creces para acabado en el plano (inc), creces en el contorno del borde		
14	UZ	<_FALD>	REAL	Creces para acabado en profundidad (inc), creces en el fondo (se introduce sin signo)		
15	FX	<_FFP1>	REAL	Avance en el contorno		
16	FZ	<_FFD>	REAL	Avance de penetración profundidad		
17		<_CDIR>	INT	Sentido de fresado (se introduce sin signo)		
				UNIDADES:		
					0 =	En concordancia
					1 =	En oposición
18		<_VARI>	INT	Mecanizado		
				UNIDADES:		
					1 =	Desbaste
					2 =	Acabado
					5 =	Achaflanado
19	L1	<_AP1>	REAL	Longitud del saliente bruto		
20	W1	<_AP2>	REAL	Anchura del saliente bruto		
21	FS	<_FS>	REAL	Ancho del chaflán (inc)		
22	ZFS	<_ZFS>	REAL	Profundidad de penetración (punta de la herramienta) en el achaflanado (abs, inc), ver <_AMODE>		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
23		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	selección del mecanizado o sólo cálculo del punto inicial	
					0 =	Modo de compatibilidad
					1 =	Mecanizado normal
				MILLARES:	acotado del saliente a través de centro o esquina	
					0 =	Modo de compatibilidad
					1 =	Acotado a través de centro
					2 =	Acotado de esquina, saliente +L +W
					3 =	Acotado de esquina, saliente -L +W
					4 =	Acotado de esquina, saliente +L -W
					5 =	Acotado de esquina, saliente -L -W
				DECENAS DE MILLAR:	mecanizado completo o repaso	
					0 =	Modo de compatibilidad
					1 =	Mecanizado completo
					2 =	Repaso
24		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	---	reservado
				CENTENAS:	---	reservado
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
25		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad final Z1 (DP)	
					0 =	Compatibilidad
					1 =	Incremental
					2 =	Absoluta
				DECENAS:	reservado	
				CENTENAS:	Profundidad de penetración en el achaflanado (ZFS)	
					0 =	Absoluta
					1 =	Incremental

3.25.1.18 CYCLE77, saliente circular

Sintaxis

```
CYCLE77(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_CDIAM>, <_PA>,
<_PO>, <_MID>, <_FAL>, <_FALD>, <_FFP1>, <_FFD>, <_CDIR>, <_VARI>,
<_AP1>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<_RTP>	REAL	Plano de retirada (abs)		
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)		
3	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<_DP>	REAL	Profundidad del saliente (abs)		
5		<_DPR>	REAL	Profundidad del saliente (inc), referida a Z0 (se introduce sin signo)		
6	∅	<_CDIAM>	REAL	Diámetro del saliente (se introduce sin signo)		
7	X0	<_PA>	REAL	Punto de referencia del saliente, 1.er eje del plano (abs)		
8	Y0	<_PO>	REAL	Punto de referencia del saliente, 2.er eje del plano (abs)		
9	DZ	<_MID>	REAL	Máx. penetración en profundidad (inc, se introduce sin signo)		
10	UXY	<_FAL>	REAL	Creces para acabado en el plano (inc), creces en el contorno del borde		
11	UZ	<_FALD>	REAL	Creces para acabado en profundidad (inc), creces en el fondo (se introduce sin signo)		
12	FX	<_FFP1>	REAL	Avance en el contorno		
13	FZ	<_FFD>	REAL	Avance de penetración profundidad		
14		<_CDIR>	INT	Sentido de fresado (se introduce sin signo)		
				UNIDADES:		
					0 =	En concordancia
					1 =	En oposición

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
15		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	Mecanizado	
					1 =	Desbaste hasta creces para acabado
					2 =	Acabado (creces X/Y/Z = 0)
					5 =	Achaflanado
16	∅1	<_AP1>	REAL	Diámetro del saliente de la pieza en bruto		
17	FS	<_FS>	REAL	Ancho del chaflán (inc)		
18	ZFS	<_ZFS>	REAL	Profundidad de penetración (punta de la herramienta) en el achaflanado (abs/inc), ver <_AMODE>		
19		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	selección del mecanizado/sólo cálculo del punto inicial	
					0 =	Modo de compatibilidad
					1 =	Mecanizado normal
				MILLARES:	reservado	
				DECENAS DE MILLAR:	mecanizado completo/repaso	
					0 =	Modo de compatibilidad (tratar <_AP1> como hasta ahora)
					1 =	Mecanizado completo
					2 =	Repaso
20		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	---	reservado
				CENTENAS:	---	reservado
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
21		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad final Z1 (DP)	
					0 =	Absoluta (modo de compatibilidad)
					1 =	Incremental
					2 =	Absoluta
				DECENAS:	reservado	
				CENTENAS:	Profundidad de penetración en el achaflanado (ZFS)	
					0 =	Absoluta
					1 =	Incremental

3.25.1.19 CYCLE78, fresado de roscas

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	RP	<_RTP>	REAL	Plano de retirada (abs)
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)
3	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)
4	Z1	<_DP>	REAL	Profundidad de taladrado final (abs/inc), ver <_AMODE>
5		<_ADPR>	REAL	Profundidad de punteado con avance de taladrado reducido (inc), activo con <_VARI> DECENAS DE MILLAR
6	D	<_FDPR>	REAL	Penetración máxima en profundidad (inc) D ≥ Z1 ⇒ una penetración hasta profundidad de taladrado final D < Z1 ⇒ ciclo de taladrado profundo con varias penetraciones y extracción de virutas
7	ZR	<_LDPR>	REAL	Profundidad de taladrado residual en taladro pasante (inc), con avance FR
8	∅	<_DIAM>	REAL	Diámetro nominal de la rosca
9	P	<_PIT>	REAL	Paso de rosca como valor numérico

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
10		<_PITA>	INT	Evaluación del paso de rosca P		
					1 =	paso en mm/vuelta
					2 =	paso en filetes/pulgada
					3 =	paso en pulg/vuelta
					4 =	paso como MÓDULO
11	DF	<_DAM>	REAL	Valor absoluto/porcentaje para cada penetración adicional (degresión), ver <_AMODE>		
12	V1	<_MDEP>	REAL	Penetración mínima (inc), sólo activa con degresión		
13		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	reservado	
				DECENAS:	Sacar virutas antes de fresar rosca	
					0 =	sin sacar virutas antes del fresado de roscas (activo solo en profundidad de taladrado final)
					1 =	sacar virutas antes del fresado de roscas (activo solo en profundidad de taladrado final)
				CENTENAS:	Roscado a derechas/izquierdas	
					0 =	Rosca a derechas
					1 =	Rosca a izquierdas
				MILLARES:	Profundidad de taladrado residual con avance de taladrado	
					0 =	sin profundidad de taladrado residual con avance de taladrado FR
					1 =	profundidad de taladrado residual con avance de taladrado FR
				DECENAS DE MILLAR:	Puntear con avance reducido	
					0 =	sin puntear con avance reducido
					1 =	puntear con avance reducido Avance de punteado = 0,3 F1 si F1 < 0,15 mm/vuelta Avance de punteado = 0,1 mm/vuelta si F1 ≥0,15 mm/vuelta
14		<_CDIR>	INT	Sentido de fresado	0 =	En concordancia
					1 =	En oposición
					4 =	En oposición + en concordancia (combinación de desbaste + acabado)
15	Z2	<_GE>	REAL	Valor de retirada antes de fresado de roscas (inc)		
16	F1	<_FFD>	REAL	Avance de taladrado (mm/min o pulg/min, o bien mm/vuelta)		
17	FR	<_FRDP>	REAL	Avance de taladrado para profundidad de taladrado residual (mm/min o mm/vuelta)		
18	F2	<_FFR>	REAL	Avance para fresado de roscas (mm/min o mm/diente)		
19	FS	<_FFP2>	REAL	Avance de acabado para <_CDIR> = 4 (mm/min o mm/diente)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
20		<_FFA>	INT	Evaluación de avances		
				UNIDADES:	avance de taladrado F1	
				DECENAS:	avance de taladrado para profundidad de taladrado residual FR	
				CENTENAS:	avance para fresado de roscas F2	
				MILLARES:	avance de acabado FS	
21		<_PITM>	STRING[15]	Cadena de caracteres como marca para la introducción del paso de rosca (sólo para la interfaz)[1)]		
22		<_PTAB>	STRING[20]	Cadena de caracteres para la tabla de roscas ("", "ISO", "BSW", "BSP", "UNC") (sólo para la interfaz)[1)]		
23		<_PTABA>	STRING[20]	Cadena de caracteres para la selección en la tabla de roscas (p. ej. "M 10", "M 12"...) (sólo para la interfaz)[1)]		
24		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados), reservado		
25		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
26		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad de taladrado = profundidad de taladrado final Z1 abs/inc	
					0 =	Absoluto
					1 =	Incremental
				DECENAS:	valor/porcentaje DF para cada penetración adicional (degresión)	
					0 =	Valor absoluto
					1 =	Porcentaje (0,001 hasta 100%)

Nota

[1)] Los parámetros 21, 22 y 23 se utilizan solamente al seleccionar la rosca en las tablas de roscas de la máscara de entrada. No se puede acceder a las tablas de roscas mediante la definición de ciclo en el tiempo de ejecución del ciclo.

3.25.1.20 CYCLE79, poliedro

Sintaxis

```
CYCLE79(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_NUM>, <_SWL>, <_PA>,
<_PO>, <_STA>, <_RC>, <_AP1>, <_MIDA>, <_MID>, <_FAL>, <_FALD>,
```

```
<_FFP1>, <_CDIR>, <_VARI>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<_RTP>	REAL	Plano de retirada (abs)		
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)		
3	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<_DP>	REAL	Profundidad de poliedro (abs/inc), ver <_AMODE>		
5	N	<_NUM>	INT	Número de bordes (1...n)		
6	SW/L	<_SWL>	REAL	Ancho de llave o longitud de borde (según <_VARI>) ("SW" con ancho de llave, "L" con longitud de borde) Ancho de llave sólo en caso de un número par de bordes y borde único		
7	X0	<_PA>	REAL	Punto de referencia del saliente, 1.er eje (abs)		
8	Y0	<_PO>	REAL	Punto de referencia del saliente, 2.er eje (abs)		
9	α0	<_STA>	REAL	Ángulo de giro, centro del borde con respecto al 1.er eje (eje X)		
10	R1/FS1	<_RC>	REAL	Redondeado de bordes con <_NUM> > 2 (radio/chaflán, ver <_AMODE>) (inc, se introduce sin signo) ("R1" con radio, "FS1" con chaflán)		
11	∅	<_AP1>	REAL	Diámetro en bruto del saliente		
12	DXY	<_MIDA>	REAL	Ancho máximo de penetración (unidad, ver <_AMODE>)		
13	DZ	<_MID>	REAL	Penetración máx. en profundidad		
14	UXY	<_FAL>	REAL	Creces para acabado en el plano		
15	UZ	<_FALD>	REAL	Creces para acabado profundidad		
16	F	<_FFP1>	REAL	Avance de mecanizado		
17		<_CDIR>	INT	Sentido de fresado	0 =	En concordancia
					1 =	En oposición
18		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	Mecanizado	
					1 =	Desbaste
					2 =	Acabado
					3 =	Acabado borde
					5 =	Achaflanado
				DECENAS:	Ancho de llave o longitud de borde	
					0 =	Ancho de llave
					1 =	Longitud del borde
19	FS	<_FS>	REAL	Ancho del chaflán (inc)		
20	ZFS	<_ZFS>	REAL	Profundidad de penetración (punta de la herramienta) en el achaflanado (abs/inc), ver <_AMODE>		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
21		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	Selección del mecanizado o solo cálculo del punto inicial	
					1 =	Mecanizado normal
22		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/18/19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	---	reservado
				CENTENAS:	---	reservado
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple
23		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad final (<_DP>)	
					0 =	Absoluta
					1 =	Incremental
				DECENAS:	Unidad para la penetración del plano (<_MIDA>)	
					0 =	mm
					1 =	% del diámetro de la herramienta
				CENTENAS:	Profundidad de penetración en el achaflanado (<_ZFS>)	
					0 =	Absoluta
					1 =	Incremental
				MILLARES:	Redondeado de bordes (<_RC>)	
					0 =	Radio
					1 =	Chaflán

3.25.1.21 CYCLE81, taladrado, punteado

Sintaxis

```
CYCLE81(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<RTP>	REAL	Plano de retirada (abs)		
2	Z0	<RFP>	REAL	Punto de referencia (abs)		
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1/∅	<DP>	REAL	Profundidad de taladrado (abs)/diámetro del centrado (abs), ver <_GMODE>		
5	Z1	<DPR>	REAL	Profundidad de taladrado (inc)		
6	DT	<DTB>	REAL	Tiempo de espera en la profundidad de taladrado final, ver <_AMODE>		
7		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	centrado relativo a la profundidad/el diámetro	
					0 =	Compatibilidad, profundidad
					1 =	Diámetro
8		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
9		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad de taladrado Z1 (abs/inc)	
					0 =	Compatibilidad, a partir de la programación de DP/DPR
					1 =	Incremental
					2 =	Absoluto
				DECENAS:	Tiempo de espera en la profundidad de taladrado final DT en segundos/vueltas	
					0 =	Compatibilidad, a partir del signo de DTB (> 0 segundos o < 0 vueltas)
					1 =	En segundos
					2 =	En vueltas

3.25.1.22 CYCLE82, taladrado, avellanado

Sintaxis

```
CYCLE82(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>, <_VARI>, <S_ZA>, <S_FA>, <S_ZD>, <S_FD>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<RTP>	REAL	Plano de retirada (abs)		
2	Z0	<RFP>	REAL	Punto de referencia (abs)		
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<DP>	REAL	Profundidad de taladrado (abs), ver <_AMODE>		
5	Z1	<DPR>	REAL	Profundidad de taladrado (inc), ver <_AMODE>		
6	DT	<DTB>	REAL	Tiempo de espera en la profundidad de taladrado final, ver <_AMODE>		
7		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	Profundidad de taladrado relativa a punta/vástago	
					0 =	Compatibilidad, punta
					1 =	Vástago
8		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:	reservado	
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: Completa
					1 =	Entrada: Simple

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
9		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad de taladrado Z1 (abs/inc)	
					0 =	Compatibilidad a partir de la programación de DP/DPR
					1 =	Incremental
					2 =	Absoluto
				DECENAS:	Tiempo de espera DT en la profundidad de taladrado final en segundos/vueltas	
					0 =	Compatibilidad, a partir del signo de DT (> 0 segundos/< 0 vueltas)
					1 =	En segundos
					2 =	En vueltas
				CENTENAS:	profundidad de punteado ZA abs/inc	
					0 =	Incremental
					1 =	Absoluto
				MILLARES:	Evaluación avance de punteado	
					0 =	En % del avance de taladrado
					1 =	F/min
					2 =	F/vuelta
				DECENAS DE MILLAR:	profundidad de taladrado residual ZD abs/inc	
					0 =	Incremental
					1 =	Absoluto
				CENTENAS DE MILLAR:	Evaluación profundidad de taladrado residual	
					0 =	En % del avance de taladrado
					1 =	F/min
					2 =	F/vuelta
10		<_VARI>	INT	Tipo de mecanizado punteado/taladro pasante		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:	Taladro pasante	
					0 =	Taladro pasante "No"
					1 =	Taladro pasante "Sí"
				DECENAS DE MILLAR:	Puntear	
					0 =	Punteado "No"
					1 =	Punteado "Sí"
11	ZA	<S_ZA>	REAL	Profundidad de punteado incremental relativa a punto de referencia, o absoluta (ver <_AMODE> CENTENAS)		
12	FA	<S_FA>	REAL	Avance de punteado como valor o en % (en combinación con <_AMODE> MILLARES)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
13	ZD	<S_ZD>	REAL	Profundidad de taladrado residual incremental referida a profundidad de taladrado final o absoluta (en combinación con <_AMODE> DECENAS DE MILLAR)
14	FD	<S_FD>	REAL	Avance de taladrado residual como valor o en % (en combinación con <_AMODE> CENTENAS DE MILLAR)

3.25.1.23 CYCLE83, taladrado profundo 1

Sintaxis

```
CYCLE83(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <FDEP>, <FDPR>, <_DAM>,
<DTB>, <DTS>, <FRF>, <VARI>, <_AXN>, <_MDEP>, <_VRT>, <_DTD>,
<_DIS1>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<RTP>	REAL	Plano de retirada (abs)		
2	Z0	<RFP>	REAL	Punto de referencia (abs)		
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<DP>	REAL	Profundidad de taladrado final (abs), ver <_AMODE>		
5	Z1	<DPR>	REAL	Profundidad de taladrado final (inc), ver <_AMODE>		
6	D	<FDEP>	REAL	1.ª profundidad de taladrado (abs), ver <_AMODE>		
7	D	<FDPR>	REAL	1.ª profundidad de taladrado (inc), ver <_AMODE>		
8	DF	<_DAM>	REAL	Valor de degresión/porcentaje para cada penetración adicional, ver <_AMODE>		
9	DTB	<DTB>	REAL	Tiempo de espera en la profundidad de taladrado, ver <_AMODE>		
10	DTS	<DTS>	REAL	Tiempo de espera en el punto inicial (solo con Sacar virutas), ver <_AMODE>		
11	FD1	<FRF>	REAL	Porcentaje de avance en la primera penetración, ver <_AMODE>		
12		<VARI>	INT	Tipo de mecanizado		
				UNIDADES:	Romper/sacar virutas	
					0 =	Romper virutas
					1 =	Sacar virutas
13		<_AXN>	INT	Eje de herramienta		
					0 =	3.er eje geométrico
					1 =	1.er eje geométrico
					2 =	2.º eje geométrico
					> 2	3.er eje geométrico
14	V1	<_MDEP>	REAL	Penetración mínima (sólo en porcentaje para degresión)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
15	V2	<_VRT>	REAL	Valor de retirada tras cada mecanizado (sólo con Romper virutas)		
					> 0	Valor de retirada variable
					0 =	Valor estándar 1 mm
16	DT	<_DTD>	REAL	Tiempo de espera en la profundidad de taladrado final, ver <_AMODE>		
17	V3	<_DIS1>	REAL	Distancia de parada previa (solo con Sacar virutas), ver <_AMODE>		
18		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	Profundidad de taladrado relativa a punta/vástago	
					0 =	Punta
					1 =	Vástago
19		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	---	reservado
				CENTENAS:	---	reservado
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
20		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad de taladrado = profundidad de taladrado final Z1 (abs/inc)	
					0 =	Compatibilidad, a partir de programación <DP>/<DPR>
					1 =	Incremental
					2 =	Absoluto
				DECENAS:	Tiempo de espera en la profundidad de taladrado DTB en segundos/vueltas	
					0 =	Compatibilidad a partir del signo de DTB (> 0 segundos o < 0 vueltas)
					1 =	En segundos
					2 =	En vueltas
				CENTENAS:	tiempo de espera en el punto inicial DTS en segundos/vueltas	
					0 =	Compatibilidad a partir del signo de DTS (> 0 segundos o < 0 vueltas)
					1 =	En segundos
					2 =	En vueltas
				MILLARES:	Tiempo de espera en la profundidad de taladrado final DTD en segundos/vueltas	
					0 =	Compatibilidad a partir del signo de DTD (> 0 segundos o < 0 vueltas)
					1 =	En segundos
					2 =	En vueltas
				DECENAS DE MILLAR:	1.ª profundidad de taladrado D (abs/inc)	
					0 =	Compatibilidad, a partir de programación <FDEPF>/<DPR>
					1 =	Incremental
					2 =	Absoluto
				CENTENAS DE MILLAR:	Valor de degresión/porcentaje <_DAM> para cada penetración adicional	
					0 =	Compatibilidad, a partir del signo de <_DAM> (> 0 valor de degresión o < 0 factor 0,001 hasta 1,0)
					1 =	Valor de degresión
					2 =	Porcentaje (0,001 hasta 100%)
				UNIDADES DE MILLÓN:	distancia de parada previa V3 automática/manual	
					0 =	Compatibilidad a partir del signo de <_DIS1> (= 0 automática o > 0 manual)
					1 =	Automática (se calcula en el ciclo)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
					2 =	Manual (valor programado)
				DECENAS DE MILLÓN:	Factor de avance para la primera penetración <FRF> como factor/porcentaje	
					0 =	Compatibilidad, como factor (0,001 a 1,0, FRF = 0 significa 100%)
					1 =	Porcentaje (0,001 hasta 999,999%)

3.25.1.24 CYCLE84, roscado sin mandril de compensación

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<RTP>	REAL	Plano de retirada (abs)		
2	Z0	<RFP>	REAL	Punto de referencia (abs)		
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<DP>	REAL	Profundidad de taladrado = profundidad de taladrado final (abs), ver <_AMODE>		
5	Z1	<DPR>	REAL	Profundidad de taladrado = profundidad de taladrado final (inc), ver <_AMODE>		
6	DT	<DTB>	REAL	Tiempo de espera en la profundidad de taladrado, en segundos		
7	SDE	<SDAC>	INT	Sentido de giro al final del ciclo		
8		<MPIT>	REAL	Tamaño de rosca sólo para "ISO métrica" (el paso se calcula internamente durante el tiempo de ejecución)		
9	P	<PIT>	REAL	Paso de rosca como valor; unidad de medida, ver <_PITA>		
10	αS[1)]	<POSS>	REAL	Posición del cabezal para parada de cabezal orientada		
11	S	<SST>	REAL	Velocidad de giro del cabezal para roscado con macho		
12	SR	<SST1>	REAL	Velocidad de giro del cabezal para la retirada		
13		<_AXN>	INT	Eje de taladrado	0 =	3. eje geométrico
					1 =	1. eje geométrico
					2 =	2. eje geométrico
					≥ 3 =	3. eje geométrico

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción			
14		<_PITA>	INT	Unidad de medida del paso de rosca (evaluación <PIT> y <MPIT>)			
					0 =	Paso en mm	- Evaluación <MPIT>/<PIT>
					1 =	Paso en mm	- Evaluación <PIT>
					2 =	Paso en TPI	- Evaluación <PIT> (filetes de rosca por pulgada)
					3 =	Paso en pulgadas	- Evaluación <PIT>
					4 =	MÓDULO	- Evaluación <PIT>
15		<_TECHNO>	INT	Tecnología[1)]			
				UNIDADES:	Comportamiento de parada precisa		
					0 =	Comportamiento de parada precisa como antes de la llamada activa de ciclo	
					1 =	Parada precisa G601	
					2 =	Parada precisa G602	
					3 =	Parada precisa G603	
				DECENAS:	control anticipativo		
					0 =	Con/sin control anticipativo como antes de la llamada activa de ciclo	
					1 =	Con control anticipativo FFWON	
					2 =	Sin control anticipativo FFWOF	
				CENTENAS:	aceleración		
					0 =	SOFT/BRISK/DRIVE como antes de la llamada activa de ciclo	
					1 =	Con limitación de tirones SOFT	
					2 =	Sin limitación de tirones BRISK	
					3 =	Aceleración reducida DRIVE	
				MILLARES:	modo de cabezal en MCALL		
					0 =	con MCALL, reactivar modo de cabezal	
					1 =	con MCALL, permanecer en regulación de posición	

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
16		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	0 =	1 corte
					1 =	Romper virutas (roscado de agujero profundo)
					2 =	Sacar virutas (roscado de agujero profundo)
				MILLARES:	Modo ISO/SIEMENS para máscara de entrada no relevante	
					0 =	Llamada desde compatibilidad ISO
					1 =	Llamada desde contexto SIEMENS
17	D	<_DAM>	REAL	Penetración máxima en profundidad (sólo con Sacar/Romper virutas)		
18	V2	<_VRT>	REAL	Valor de retirada tras cada mecanizado (solo con Romper virutas), ver <_AMODE>		
19		<_PITM>	STRING[15]	Cadena de caracteres como marca para la introducción del paso de rosca[2)]		
20		<_PTAB>	STRING[5]	Cadena de caracteres para la tabla de roscas ("", "ISO", "BSW", "BSP", "UNC")[2)]		
21		<_PTABA>	STRING[20]	Cadena de caracteres para la selección en la tabla de roscas (p. ej. "M 10", "M 12"...)[2)]		
22		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
23		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:	Modo de compatibilidad (solo para la máscara de entrada que se decompilará) si MD 52216 bit 0 = 1[1)]	
					0 =	se muestran los parámetros tecnológicos (compatibilidad): los parámetros TECHNO están activos
					1 =	no se muestran los parámetros tecnológicos: está activa la tecnología "como programado antes de la llamada de ciclo"
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
24		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad de taladrado = profundidad de taladrado final Z1 (abs/inc)	
					0 =	Compatibilidad, a partir de programación <DP>/<DPR>
					1 =	Incremental
					2 =	Absoluto
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:	sentido del tallado de rosca derecha/izquierda	
					0 =	compatibilidad, a partir del signo de PIT/MPIT
					1 =	dcha.
					2 =	izda.
				DECENAS DE MILLAR:	reservado	
				CENTENAS DE MILLAR:	reservado	
				UNIDADES DE MILLÓN:	valor de retirada tras cada mecanizado V2 manual/automático	
					0 =	compatibilidad, a partir de programación de <_VRT> (> 0 valor variable o ≤ 0 valor estándar 1 mm/0,0394 pulgadas)
					1 =	automático (valor estándar 1 mm/0,0394 pulgadas)
					2 =	manual (como programado en V2)

[1] Los campos de tecnología pueden estar ocultos en función del dato de operador SD52216 $MCS_FUNCTION_MASK_DRILL

[2] Los parámetros 19, 20 y 21 se utilizan solamente al seleccionar la rosca en las tablas de roscas de la máscara de entrada. No se puede acceder a las tablas de roscas mediante la definición de ciclo en el tiempo de ejecución del ciclo.

3.25.1.25 CYCLE85, escariado

Sintaxis

```
CYCLE85(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <FFR>, <RFF>,
<_GMODE>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	RP	<RTP>	REAL	Plano de retirada (abs)
2	Z0	<RFP>	REAL	Punto de referencia (abs)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<DP>	REAL	Profundidad de taladrado (abs), ver <_AMODE>		
5	Z1	<DPR>	REAL	Profundidad de taladrado (inc), ver <_AMODE>		
6	DT	<DTB>	REAL	Tiempo de espera en la profundidad de taladrado final, ver <_AMODE>		
7	F	<FFR>	REAL	Avance		
8	FR	<RFF>	REAL	Avance en la retirada		
9		<_GMODE>	INT	reservado		
10		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
11		<_AMODE>	INT	Modo alternativo (taladrado)		
				UNIDADES:	profundidad de taladrado Z1 (abs/inc)	
					0 =	Compatibilidad, a partir de la programación de DP/DPR
					1 =	Incremental
					2 =	Absoluto
				DECENAS:	Tiempo de espera DT en la profundidad de taladrado final en segundos/vueltas	
					0 =	Compatibilidad, a partir del signo de DT (> 0 segundos o < 0 vueltas)
					1 =	En segundos
					2 =	En vueltas

3.25.1.26 CYCLE86, mandrinar

Sintaxis

```
CYCLE86(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDIR>, <RPA>,
<RPO>, <RPAP>, <POSS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	RP	<RTP>	REAL	Plano de retirada (abs)
2	Z0	<RFP>	REAL	Punto de referencia (abs)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<DP>	REAL	Profundidad de taladrado (abs), ver <_AMODE>		
5	Z1	<DPR>	REAL	Profundidad de taladrado (inc), ver <_AMODE>		
6	DT	<DTB>	REAL	Tiempo de espera en la profundidad de taladrado final, ver <_AMODE>		
7	DIR	<SDIR>	INT	Sentido de giro del cabezal	3 =	M3
					4 =	M4
8	DX	<RPA>	REAL	Cuantía de retirada en dirección X		
9	DY	<RPO>	REAL	Cuantía de retirada en dirección Y		
10	DZ	<RPAP>	REAL	Cuantía de retirada en dirección Z		
11	SPOS	<POSS>	REAL	Posición del cabezal para la retirada (para parada de cabezal orientada, en grados)		
12		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	modo de retirada	
					0 =	Retirar, compatibilidad
					1 =	No levantar
13		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
14		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad de taladrado Z1 (abs/inc)	
					0 =	Compatibilidad, a partir de programación <DP>/<DPR>
					1 =	Incremental
					2 =	Absoluto
				DECENAS:	Tiempo de espera en la profundidad de taladrado final DT en segundos/vueltas	
					0 =	Compatibilidad, a partir del signo de DT (> 0 segundos o < 0 vueltas)
					1 =	En segundos
					2 =	En vueltas

3.25.1.27 CYCLE92, tronzar

Sintaxis

```
CYCLE92(<_SPD>, <_SPL>, <_DIAG1>, <_DIAG2>, <_RC>, <_SDIS>, <_SV1>,
<_SV2>, <_SDAC>, <_FF1>, <_FF2>, <_SS2>, <_DIAGM>, <_VARI>, <_DN>,
<_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	X0	<_SPD>	REAL	Punto de referencia (abs, siempre diámetro)		
2	Y0	<_SPL>	REAL	Punto de referencia (abs)		
3	X1	<_DIAG1>	REAL	Profundidad para la reducción de la velocidad de giro, ver <_AMODE> (UNIDADES)		
4	X2	<_DIAG2>	REAL	Profundidad final, ver <_AMODE> (DECENAS)		
5	R/FS	<_RC>	REAL	Radio de redondeo o ancho del chaflán, ver <_AMODE> (MILLARES)		
6	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
7	S	<_SV1>	REAL	Velocidad de giro constante del cabezal, ver <_AMODE> (DECENAS DE MILLAR)		
	V			Velocidad de corte constante		
8	SV	<_SV2>	REAL	Velocidad de giro máxima a velocidad de corte constante		
9	DIR	<_SDAC>	INT	Sentido de giro del cabezal	3 =	Para M3
					4 =	Para M4
10	F	<_FF1>	REAL	Avance hasta profundidad para reducción de velocidad de giro		
11	FR	<_FF2>	REAL	Avance reducido hasta profundidad final		
12	SR	<_SS2>	REAL	Velocidad de giro reducida hasta profundidad final		
13	XM	<_DIAGM>	REAL	Salir de profundidad bandeja de piezas (abs, siempre diámetro)		
14		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	Retirada	
					0 =	Retirada a <_SPD> + <_SDIS>
					1 =	Sin retirada al final
				DECENAS:	bandeja de piezas	
					0 =	No, no ejecutar comandos M
					1 =	Sí, llamada de CUST_TECHCYC(101)- abrir cajón, CUST_TECHCYC(102)- cerrar cajón
15		<_DN>	INT	Número D para el 2.º filo si no está programado ⇒ D+1		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
20		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
21		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Profundidad para reducción de velocidad de giro (<_DIAG1>)	
					0 =	Absoluta, valor del eje de refrentado en el diámetro
					1 =	Incremental, valor del eje de refrentado en el radio
				DECENAS:	Profundidad final (<_DIAG2>)	
					0 =	Absoluta, valor del eje de refrentado en el diámetro
					1 =	Incremental, valor del eje de refrentado en el radio
				CENTENAS:	reservado	
				MILLARES:	Radio/chaflán (<_RC>)	
					0 =	Radio
					1 =	Chaflán
				DECENAS DE MILLAR:	Velocidad de giro del cabezal/velocidad de corte (<_SV1>)	
					0 =	Velocidad de giro constante del cabezal
					1 =	Velocidad de corte constante

3.25.1.28 CYCLE95, mecanizado de contorno

Sintaxis

```
CYCLE95(<NPP>, <MID>, <FALZ>, <FALX>, <FAL>, <FF1>, <FF2>, <FF3>,
<_VARI>, <DT>, <DAM>, <_VRT>, <_GMODE>, <_DMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	CON	<NPP>	STRING [140]	Nombre de contorno
2	D	<MID>	REAL	Máxima penetración en profundidad para desbaste, ver <_GMODE>

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
3	UZ	<FALZ>	REAL	Creces para acabado en Z		
4	UX	<FALX>	REAL	Creces para acabado en X		
5	U	<FAL>	REAL	Creces para acabado paralelas al contorno (válido para los dos ejes)		
6	F	<FF1>	REAL	Avance para desbaste		
7	FY	<FF2>	REAL	Avance de penetración destalonados		
8	FS	<FF3>	REAL	Avance de acabado		
9		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES y DECENAS:		
					1 =	Desbaste, longitudinal, exterior
					2 =	Desbaste, transversal, exterior
					3 =	Desbaste, longitudinal, interior
					4 =	Desbaste, transversal, interior
					5 =	Acabado, longitudinal, exterior
					6 =	Acabado, transversal, exterior
					7 =	Acabado, longitudinal, interior
					8 =	Acabado, transversal, interior
					9 =	Mecanizado completo, longitudinal, exterior
					10 =	Mecanizado completo, transversal, exterior
					11 =	Mecanizado completo, longitudinal, interior
					12 =	Mecanizado completo, transversal, interior
				CENTENAS:		
					0 =	Con repasado en el contorno, sin esquinas residuales
					1 =	Sin repasado en el contorno
					2 =	Repasado solo hasta el punto de intersección anterior, pueden producirse esquinas residuales
10	DT	<DT>	REAL	Tiempo de espera en interrupción de avance		
11	DI	<DAM>	REAL	Distancia para interrupciones del avance		
12	VRT	<_VRT>	REAL	Trayecto de retirada del contorno		
					0 =	Internamente se utiliza un trayecto de retirada de 1 mm independientemente del sistema de medida (pulgadas/métrico)
					> 0 =	Trayecto de retirada

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
13		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	Evaluación de la profundidad de penetración	
					0 =	La profundidad de penetración se calcula de acuerdo con el grupo G DIAMON/DIAMOF
					1 =	La profundidad de penetración actúa como valor de radio (con independencia de DIAMON/ DIAMOF)
14		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				MILLARES:		
					0 =	Modo de compatibilidad: el nombre de contorno figura en NPP
					1 =	El nombre de contorno se programa en CYCLE62 y se transfiere a _SC_CONT_NAME

3.25.1.29 CYCLE98, cadena de roscas

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	Z0	<_PO1>	REAL	Punto de referencia en Z (abs)
2	X0	<_DM1>	REAL	Punto de referencia en X (abs), en el diámetro
3	Z1	<_PO2>	REAL	Punto intermedio 1 en Z (abs/inc), ver <_AMODE> (UNIDADES)
4	X1	<_DM2>	REAL	Punto intermedio 1 en X (abs/inc), ver <_AMODE> (DECENAS) o bien
	X1α			Rosca oblicua 1 (de -90° a 90°) abs siempre diámetro, inc siempre radio
5	Z2	<_PO3>	REAL	Punto intermedio 2 en Z (abs/inc), ver <_AMODE> (CENTENAS)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
6	X2	<_DM3>	REAL	Punto intermedio 2 en X (abs/inc), ver <_AMODE> (MILLARES) o bien		
	X2α			Rosca oblicua 2 (de -90° a 90°) abs siempre diámetro, inc siempre radio		
7	Z3	<_PO4>	REAL	Punto final en Z (abs/inc), ver <_AMODE> (DECENAS DE MILLAR)		
8	X3	<_DM4>	REAL	Punto final en X (abs/inc), ver <_AMODE> (CENTENAS DE MILLAR) o bien		
	X3α			Rosca oblicua 3 (de -90° a 90°) abs siempre diámetro, inc siempre radio		
9	LW	<APP>	REAL	Trayecto previo de rosca (inc, se introduce sin signo)		
10	LR	<ROP>	REAL	Trayecto de salida de rosca (inc, se introduce sin signo)		
11	H1	<TDEP>	REAL	Profundidad de rosca (inc, se introduce sin signo)		
12	U	<FAL>	REAL	Creces para acabado en X y Z		
13	DP	<_IANG>	REAL	Penetración oblicua como distancia o ángulo, ver <_AMODE> (UNIDADES DE MILLÓN)		
	αP			La penetración oblicua actúa conforme al ajuste del parámetro <_VARI> (CENTENAS).		
					Definición para <_VARI CENTENAS = 0, modo de compatibilidad:	
					> 0 =	Penetración en un flanco
					0 =	Penetración perpendicular en la rosca
					< 0 =	Penetración de flancos con flancos alternantes
					Definición para _VARI CENTENAS <> 0:	
					> 0 =	Penetración en el flanco positivo
					0 =	Penetración central
					< 0 =	Penetración en el flanco negativo
14	α0	<NSP>	REAL	Decalaje del ángulo inicial para la 1.a entrada de rosca		
15		<NRC>	INT	Número de pasadas de desbaste, ver <_VARI> (DECENAS DE MILLAR)		
16	NN	<NID>	INT	Número de cortes en vacío		
17	P0	<_PP1>	REAL	Paso de rosca de la 1.ª sección de rosca, ver <_PITA>		
18	P1	<_PP2>	REAL	Paso de rosca de la 2.ª sección de rosca, ver <_PITA>		
19	P2	<_PP3>	REAL	Paso de rosca de la 3.ª sección de rosca, ver <_PITA>		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
20		<_VARI>	INT	Mecanizado		
				UNIDADES:	Tecnología	
					1 =	Rosca exterior con penetración lineal
					2 =	Rosca interior con penetración lineal
					3 =	Rosca exterior con penetración degresiva, sección de viruta constante
					4 =	Rosca interior con penetración degresiva, sección de viruta constante
				DECENAS:	reservado	
				CENTENAS:	clase de penetración	
					0 =	Modo de compatibilidad para <_IANG>
					1 =	Penetración por un lado
					2 =	Penetración cambiante
				MILLARES:	reservado	
				DECENAS DE MILLAR:	penetración alternativa en profundidad	
					0 =	Compatibilidad, especificación del número de pasadas de desbaste (<_NRC>)
					1 =	Especificación del valor para la 1.ª penetración (<_MID>)
				CENTENAS DE MILLAR:	Tipo de mecanizado	
					0 =	Compatibilidad (desbaste y acabado)
					1 =	Desbaste
					2 =	Acabado
					3 =	Desbaste y acabado
				UNIDADES DE MILLÓN:	orden de mecanizado con roscas de varias entradas	
					0 =	Orden ascendente de los filetes
					1 =	Orden inverso de los filetes
21	N	<_NUMTH>	INT	Número de entradas de rosca		
22		<_VRT>	REAL	Distancia de retroceso (inc)		
					0 =	Internamente se utiliza un trayecto de retirada de 1 mm independientemente del sistema de medida (pulgadas/métrico)
					> 0 =	Trayecto de retirada
23	D1	<_MID>	REAL	Primera penetración, ver <_VARI> (DECENAS DE MILLAR)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
24	DA	<_GDEP>	REAL	Profundidad intermedia (activo sólo con "varias entradas")		
					0 =	No tener en cuenta profundidad intermedia
					> 0 =	Tener en cuenta profundidad intermedia
25		<_IFLANK>	REAL	Penetración oblicua como anchura (sólo para la interfaz)		
26		<_PITA>	INT	Evaluación del paso de rosca		
					0 =	Modo de compatibilidad para el paso de rosca, evaluación de <_PP1> a <_PP3> como hasta ahora conforme al sistema activo (métrico/pulgadas)
					1 =	Paso en mm
					2 =	Paso en TPI (filetes de rosca por pulgada)
					3 =	Paso en pulgadas
					4 =	MÓDULO
27		<_PITM1>	STRING[15]	Cadena de caracteres como marca para la introducción del paso de rosca (sólo para la interfaz)		
28		<_PITM2>	STRING[15]	Cadena de caracteres como marca para la introducción del paso de rosca (sólo para la interfaz)		
29		<_PITM3>	STRING[15]	Cadena de caracteres como marca para la introducción del paso de rosca (sólo para la interfaz)		
30		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	---	reservado
				CENTENAS:	---	reservado
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
31		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	1. 1.er punto intermedio en Z (Z1)	
					0 =	Absoluto
					1 =	Incremental
				DECENAS:	1. 1.er punto intermedio en X (X1)	
					0 =	Absoluto
					1 =	Incremental
					2 =	α
				CENTENAS:	2. 2.º punto intermedio en Z (Z2)	
					0 =	Absoluto
					1 =	Incremental
				MILLARES:	2. 2.º punto intermedio en X (X2)	
					0 =	Absoluto
					1 =	Incremental
					2 =	α
				DECENAS DE MILLAR:	punto final en Z (Z3)	
					0 =	Absoluto
					1 =	Incremental
				CENTENAS DE MILLAR:	punto final en X (X3)	
					0 =	Absoluto
					1 =	Incremental
					2 =	α
				UNIDADES DE MILLÓN:	selección de la penetración oblicua como ángulo o anchura	
					0 =	Ángulo de penetración <_IANG>
					1 =	Penetración oblicua <_IFLANK>
				DECENAS DE MILLÓN:	una entrada/varias entradas	
					0 =	Modo de compatibilidad (se evalúa el ángulo inicial <_NSP>)
					1 =	Un filete (con decalaje de ángulo inicial <_NSP>)
					2 =	Varios filetes

3.25.1.30 CYCLE99, tallado de roscas

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	Z0	<_SPL>	REAL	Punto de referencia (abs)		
2	X0	<_SPD>	REAL	Punto de referencia (abs, siempre diámetro)		
3	Z1	<_FPL>	REAL	Punto final en combinación con <_AMODE> (UNIDADES)		
4	X1	<_FPD>	REAL	Punto final, en combinación con <_AMODE> (DECENAS)		
5	LW/LW2	<_APP>	REAL	Trayecto previo de rosca en combinación con <_AMODE> (CENTENAS) o bien Trayecto previo de rosca = salida de rosca en combinación con <_AMODE> (CENTENAS)		
6	LR	<_ROP>	REAL	Salida de rosca		
7	H1	<_TDEP>	REAL	Profundidad de rosca		
8	U	<_FAL>	REAL	Creces para acabado en X y Z		
9	DP	<_IANG>	REAL	Penetración oblicua como distancia o ángulo, en combinación con <_AMODE> (MILLARES)		
	αP				> 0 =	Penetración en el flanco positivo
					< 0 =	Penetración en el flanco negativo
					0 =	Penetración central
10	α0	<_NSP>	REAL	Decalaje del ángulo inicial (activo sólo con "una entrada")		
11	ND	<_NRC>	INT	N.º de pasadas de desbaste, en combinación con <_VARI> (DECENAS DE MILLAR)		
12	NN	<_NID>	INT	Número de cortes en vacío		
13	P	<_PIT>	REAL	Paso de rosca como valor, en combinación con <_PITA>		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
14		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	Tecnología	
					1 =	Rosca exterior con penetración lineal
					2 =	Rosca interior con penetración lineal
					3 =	Rosca exterior con penetración degresiva, sección de viruta constante
					4 =	Rosca interior con penetración degresiva, sección de viruta constante
				DECENAS:	reservado	
				CENTENAS:	clase de penetración	
					1 =	Penetración por un lado
					2 =	Penetración cambiante
				MILLARES:	reservado	
				DECENAS DE MILLAR:	penetración alternativa en profundidad	
					0 =	Especificación del número de pasadas de desbaste (<_NRC>)
					1 =	Especificación del valor para la 1.ª penetración (<_MID>)
				CENTENAS DE MILLAR:	Tipo de mecanizado	
					1 =	Desbaste
					2 =	Acabado
					3 =	Desbaste y acabado
				UNIDADES DE MILLÓN:	orden de mecanizado con roscas de varias entradas	
					0 =	Orden ascendente de los filetes
					1 =	Orden inverso de los filetes
15	N	<_NUMTH>	INT	Número de entradas de rosca		
16	VR	<_SDIS>	REAL	Distancia de retroceso, inc		
17	D1	<_MID>	REAL	Primera profundidad de penetración, en combinación con <_VARI> (DECENAS DE MILLAR)		
18	DA	<_GDEP>	REAL	Profundidad intermedia (activo sólo con "varios filetes")		
					0 =	No tener en cuenta profundidad intermedia
					> 0 =	Tener en cuenta profundidad intermedia
19	G	<_PIT1>	REAL	Modificación del paso por vuelta		
					0 =	El paso de rosca es constante (G33)
					> 0 =	El paso de rosca aumenta (G34)
					< 0 =	El paso de rosca disminuye (G35)
20		<_FDEP>	REAL	Profundidad de penetración (se introduce sin signo)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
21	N1	<_GST>	INT	Filete inicial N1 = 1...N, en combinación con <_AMODE> (CENTENAS DE MILLAR)		
22		<_GUD>	INT	reservado		
23		<_IFLANK>	REAL	Penetración oblicua como anchura (sólo para la interfaz)		
24		<_PITA>	INT	Unidad de medida del paso de rosca (evaluación PIT y/o MPIT)		
					0 =	Paso en mm, evaluación MPIT/PIT
					1 =	Paso en mm, evaluación PIT
					2 =	Paso en TPI, evaluación PIT (filetes de rosca por pulgada)
					3 =	Paso en mm, evaluación PIT
					4 =	MÓDULO, evaluación PIT
25		<_PITM>	STRING[15]	Cadena de caracteres como marca para la introducción del paso de rosca (sólo para la interfaz)[1)]		
26		<_PTAB>	STRING[20]	Cadena de caracteres para la tabla de roscas (sólo para la interfaz)[1)]		
27		<_PTABA>	STRING[20]	Cadena de caracteres para la selección en la tabla de roscas (sólo para la interfaz)[1)]		
28		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	tipo de rosca	
					0 =	Rosca longitudinal
					1 =	Rosca transversal
					2 =	Rosca cónica
				CENTENAS:	---	reservado
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: Simple

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
29		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	longitud de rosca en Z	
					0 =	Absoluto
					1 =	Incremental
				DECENAS:	longitud de rosca en X	
					0 =	Absoluta, valor del eje de refrentado en el diámetro
					1 =	Incremental, valor del eje de refrentado en el radio
					2 =	α
				CENTENAS:	Evaluación del trayecto previo/de entrada <_APP>	
					0 =	Trayecto previo de rosca <_APP>
					1 =	Entrada en rosca = salida de rosca <_APP> = -<_ROP>
					2 =	Especificar entrada en rosca <_APP> = -<_APP>
				MILLARES:	selección de la penetración oblicua como ángulo o anchura	
					0 =	Ángulo de penetración <_IANG>
					1 =	Penetración oblicua <_IFLANK>
				DECENAS DE MILLAR:	una entrada/varias entradas	
					0 =	Un filete (con decalaje de ángulo inicial <_NSP>)
					1 =	Varios filetes
				CENTENAS DE MILLAR:	Filete inicial <_GST>	
					0 =	Mecanizado completo
					1 =	Iniciar el mecanizado a partir de este filete
					2 =	Mecanizar sólo este filete
				UNIDADES DE MILLÓN:	Compensación de flexión con rosca longitudinal	
					0 =	Altura de segmento rosca convexa XS
					1 =	Radio rosca convexa RS
30	XS/RS	<_S_XRS>	REAL	Compensación de flexión con rosca longitudinal en combinación con <_AMODE>: UNIDADES DE MILLÓN		

Nota

[1] Los parámetros <_PITM>, <_PTAB> y <_PTABA> se utilizan solamente al seleccionar la rosca en las tablas de roscas de la máscara de entrada.
No se puede acceder a las tablas de roscas mediante la definición de ciclo en el tiempo de ejecución del ciclo.

3.25.1.31 CYCLE435 - Definir el sistema de coordenadas del diamante

Sintaxis

```
CYCLE435(<_T>, <_DD>, <S_TA>, <S_DA>, <S_AD>, <S_AL>, <S_PVD>,
<S_PVL>, <S_PD>, <S_PL>, <_AMODE>)
```

Parámetro

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1		<_T>	STRING[32]	Nombre de herramienta de la muela rectificadora		
2		<_DD>	INT	Número de filo de la muela rectificadora		
3		<S_TA>	STRING[32]	Punto de referencia diamante - Nombre del diamante		
4		<S_DA>	INT	Número de filo del diamante		
5		<S_AD>	REAL	Valor de diamantado diámetro		
6		<S_AL>	REAL	Valor de diamantado plano		
7		<S_PVD>	REAL	Decalaje del perfilado diámetro		
8		<S_PVL>	REAL	Decalaje del perfilado plano		
9		<S_PD>	REAL	Sobredimensionamiento del perfilado diámetro		
10		<S_PL>	REAL	Sobredimensionamiento del perfilado plano		
11		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Herramienta activa al final del ciclo	
					0 =	Diamante activo
					1 =	Muela activa

3.25.1.32 CYCLE495, perfilar

Sintaxis

```
CYCLE495(<_T>, <_DD>, <_SC>, <_F>, <_VARI>, <_D>, <_DX>, <_DZ>,
<S_PA>, <S_N>, <_DMODE>, <_AMODE>, <S_FW>, <S_HW>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1		<_T>	STRING[20]	Nombre de herramienta de la muela rectificadora
2		<_DD>	INT	Número de filo de la muela rectificadora
3		<_SC>	REAL	Cuantía de retirada para esquivar obstáculos, incremental
4		<_F>	REAL	Avance perfilado

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
5		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	Tipo de perfilado	
					1 =	Paralelo al eje
					2 =	Paralelo al contorno
				DECENAS:	Sentido de mecanizado	
					0 =	Axial posible con las posiciones de filo 1 a 4
					1 =	Radial posible con las posiciones de filo 1 a 4
					2 =	Cambiante posible con las posiciones de filo 1 a 8
					3 =	Inicio → final posible con las posiciones de filo 1 a 8
					4 =	Final → inicio posible con las posiciones de filo 1 a 8
				CENTENAS:	Dirección de penetración	
					1 =	Penetración X- con G18 o bien Y- con G19
					2 =	Penetración X+ con G18 o bien Y + con G19
					3 =	Penetración Z- con G18 y con G19
					4 =	Penetración Z+ con G18 y con G19
6		<_D>	REAL	Valor de diamantado con tipo de perfilado en paralelo al eje		
7		<_DX>	REAL	Valor de diamantado X con G18 o bien Y con G19 con tipo de perfilado en paralelo al contorno		
8		<_DZ>	REAL	Valor de diamantado Z con G18 y G19 con tipo de perfilado en paralelo al contorno		
9		<S_PA>	REAL	Sobredim. perfilado		
10		<S_N>	INT	Número de pasadas en el programa de perfilado		
11		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
12		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	Selección perfilado nuevo/continuar	
					1 =	Nuevo
					2 =	Continuar
				DECENAS:	Selección sobredim. perfilado	
					0 =	Desde el contorno bruto hasta el punto más bajo del contorno
					1 =	Desde el contorno bruto hasta el punto más alto del contorno
13		<S_FW>	REAL	Ángulo de despulla del diamante		
14		<S_HW>	REAL	Ángulo de soporte del diamante		

3.25.1.33 CYCLE782: Adaptar a la carga

Nota

Para poder usar CYCLE782 se necesita una licencia para la opción "Adaptación inteligente de la carga".

Sintaxis

```
CYCLE782(<S_MODE>, <S_TESTAXIS>, <S_VALUE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1		<S_MODE>	INT	Modo de mecanizado		
				UNIDADES:	Selección/deselección	
					0 =	Desactivar la adaptación
					1 =	Activar la adaptación
					2 =	Calcular y activar la inercia básica del eje (función de puesta en marcha)
				DECENAS:	Variante de medida (solo para <S_MODE> CENTENAS = 0)	
					0 =	Modo normal
					1 =	Modo exacto (p. ej., con cálculo de rozamiento, etc.)
				CENTENAS	Medición o especificación de un valor de inercia	
					0 =	Medición En este modo se ejecutan desplazamientos para determinar la carga.
					1 =	Especificación de un valor de inercia En este modo no se ejecutan desplazamientos para determinar la carga. En lugar de ello, se transmite un valor de inercia calculado previamente, p. ej., por medio de la optimización automática servo (AST).
				MILLARES	Visualización de resultados de medición	
					0 =	Pantalla de resultados de medición DES
					1 =	Pantalla de resultados de medición CON
				DECENAS DE MILLAR	Duración de la indicación del resultado de medición	
					1 =	Deselección automática de la visualización al cabo de 8 s
					3 =	Confirmación con marcha CN
				CENTENAS DE MILLAR	Selección de carga [1]	
					0 =	Cálculo de la carga en total: eje vacío + pieza
					1 =	Cálculo de la carga por separado: Pieza
2		<S_TESTAXIS>	AXIS	Eje de canal: • para el que deben activarse adaptaciones. • para el que debe calcularse la inercia (solo para <S_MODE> CENTENAS = 0)		
3		<S_VALUE>	REAL	Valor de inercia (solo para <S_MODE> CENTENAS = 1)		

[1] El valor se toma automáticamente del dato de máquina 52212 $MCS_FUNCTION_MASK_TECH, bit 18.

Nota

La interfaz de usuario no soporta la función de puesta en marcha "Calcular y activar la inercia básica del eje" (<S_MODE> UNIDADES = 2). Para ello, debe programarse la llamada de ciclo en sí. El resultado de la medición se escribe en el dato de máquina 53350 $MAS_ILC_BASE_VALUE.

La interfaz de usuario lee el valor del parámetro "Selección de carga" (<S_MODE> CENTENAS DE MILLAR) a partir del dato de máquina 52212 $MCS_FUNCTION_MASK_TECH, bit 18, y lo escribe en la llamada de ciclo.

La carga puede calcularse o predefinirse tanto para ejes lineales como para ejes giratorios. Para ejes lineales se muestra siempre una masa, y para ejes giratorios, siempre una inercia en la pantalla de resultados de medición, o se predefinen en la interfaz de usuario.

Independientemente del tipo de eje, en el dato de máquina 53350 $MAS_ILC_BASE_VALUE y en la adaptación se escribe siempre una inercia.

Ejemplos

Ejemplo 1:

Se desea calcular la carga del eje MX1 al principio de un programa CN y actualizar las adaptaciones en función de ella. Se mostrará el resultado y se confirmará la visualización con marcha CN.

Código de programa

```
...
CYCLE782(31011,MX1)
...
```

Ejemplo 2:

Durante el curso del programa, el valor de inercia calculado, p. ej., por medio de AST, debe enviarse a la variable _MY_VALUE, debe llamarse el ciclo de carga con ese valor y, a partir de él, deben activarse adaptaciones para el eje MX2. No se mostrarán resultados.

Código de programa

```
DEF REAL _MY_VALUE
_MY_VALUE=...
...
CYCLE782(101,MX2,_MY_VALUE)
...
```

Ejemplo 3:

Se desean desactivar las adaptaciones para el eje MX2.

Código de programa

```
...
CYCLE782(0,MX2)
```

```
Código de programa
...
```

3.25.1.34 CYCLE800, orientar plano/orientar herramienta/alinear herramienta

Sintaxis

```
CYCLE800(<_FR>, <_TC>, <_ST>, <_MODE>, <_X0>, <_Y0>, <_Z0>, <_A>,
<_B>, <_C>, <_X1>, <_Y1>, <_Z1>, <_DIR>, <_FR_I>, <_DMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1		<_FR>	INT	Modo de retirada:	0 =	Sin retirada
					1 =	Retirada del eje Z de máquina
					2 =	Retirada del eje Z de máquina y luego XY
					3 =	reservado
					4 =	Retirada en la dirección de la herramienta máxima
					5 =	Retirada en la dirección de la herramienta incremental
2		<_TC>	STRING[32]	Nombre del juego de datos de orientación	""	"" (sin nombre) si sólo existe 1 juego de datos de orientación
					"0"	Deselección del juego de datos de orientación (borrado de los frames de orientación)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
3		<_ST>	INT	Estado de las transformaciones		
				UNIDADES:		
					0 =	Nuevo, el plano de orientación se borra y se vuelve a calcular con los parámetros actuales
					1 =	Aditivo, el plano de orientación se coloca de forma aditiva sobre el plano de orientación activo
				DECENAS:	seguimiento de la punta de herramienta sí/no (sólo activo si en la PeM está configurada la función ORIENTACIÓN)	
					0 =	Sin seguimiento de la punta de la herramienta
					1 =	Con seguimiento de la punta de la herramienta (TRAORI)
				CENTENAS:	posicionar/alinear herramienta (la función se muestra en la máscara de entrada Orientac. herram.)	
					0 =	No posicionar herramienta
					1 =	Posicionar herramienta (preferiblemente fresa para radios)
					2 =	Alinear herramienta de **torneado** (si está configurada Cinemática eje B para tecnol. torneado en PeM Orientación)
					3 =	Alinear herramienta de **fresado** (si está configurada Cinemática eje B para tecnol. torneado en PeM Orientación)
				MILLARES:	parámetro interno Orientación en JOG	
				DECENAS DE MILLAR:	Ver parámetro Sentido <_DIR>	
					0 =	Orientación "sí"
					1 =	Orientación "no", sentido "negativo" [3]
					2 =	Orientación "no", sentido "positivo" [3]
				CENTENAS DE MILLAR:	Ver parámetro Sentido <_DIR>	
					0 =	Compatibilidad
					1 =	Selección de sentido "negativo" optimizada (solo para interfaz de usuario) [4]
					2 =	Selección de sentido "positivo" optimizada (solo para interfaz de usuario) [4]

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
4		<_MODE> [5]	INT	Modo de orientación: evaluación de los ángulos de giro y del orden de giro (codificada al bit)		
				Bit: 7 6	0 0:	Ángulo de giro por eje -> ver parámetros <_A>, <_B>, <_C>
					0 1:	Ángulo espacial -> ver parámetros <_A>, <_B> [1]
					1 0:	Ángulo de proyección -> ver parámetros <_A>, <_B>, <_C> [1]
					1 1:	Modo de orientación de ejes giratorios directo -> ver parámetros <_A>, <_B> [1]
				Bit: 5 4 3 2 1 0 (en ángulos espaciales no es relevante)	x x x x 0 1	1.er giro _A alrededor de X
					x x x x 1 0	1.er giro _A alrededor de Y
					x x x x 1 1	1.er giro _A alrededor de Z
					x x 0 1 x x	2.º giro _B alrededor de X
					x x 1 0 x x	2.º giro _B alrededor de Y
					x x 1 1 x x	2.º giro _B alrededor de Z
					0 1 x x x x	3.er giro _C alrededor de X
					1 0 x x x x	3.er giro _C alrededor de Y
					1 1 x x x x	3.er giro _C alrededor de Z
5	X0	<_X0>	REAL	Punto de referencia X antes del giro		
6	Y0	<_Y0>	REAL	Punto de referencia Y antes del giro		
7	Z0	<_Z0>	REAL	Punto de referencia Z antes del giro		
8	X(A)	<_A>	REAL	1.er giro según el ajuste en el parámetro <_MODE>		
9	Y(B)	<_B>	REAL	2.º giro según el ajuste en el parámetro <_MODE>		
10	Z(C)	<_C>	REAL	3.er giro según el ajuste en el parámetro <_MODE>		
11	X1	<_X1>	REAL	Punto de referencia X después del giro		
12	Y1	<_Y1>	REAL	Punto de referencia Y después del giro		
13	Z1	<_Z1>	REAL	Punto de referencia Z después del giro		
14	- o bien +	<_DIR>	INT	Activar el desplazamiento de los ejes giratorios (predeterminado = -1)		
					-1 =	Posicionar a un valor menor del eje giratorio 1 ó 2 [2]
					+1 =	Posicionar a un valor mayor del eje giratorio 1 ó 2 [2]
					0 =	Orientación no (sólo calcular frame de orientación) [1] [3]
15	FR	<_FR_I>	REAL	Retirar valor (inc) en la dirección de la herramienta incremental		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
16		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	Representación del valor Beta con Alinear herramienta	
					0 =	Valor
					1 =	Flecha

Nota

Si los siguientes parámetros de transferencia se han programado indirectamente (como parámetros), la máscara de entrada no se decompila: <_FR>, <_ST>, <_TC>, <_MODE>, <_DIR>

[1] La selección es posible si en PeM está configurada la función ORIENTACIÓN.

[2] La selección es posible si en PeM está ajustada la referencia de sentido ORIENTACIÓN al eje giratorio 1 o 2.

Ningún campo de selección en la referencia de sentido "no"

[3] La selección de Orientación "no" puede estar oculta SD 55221 bit 0

Orientación "no" sentido "negativo" corresponde a <_DIR> = 0 y _ST DECENAS DE MILLAR = 1

Orientación "no" sentido "positivo" corresponde a <_DIR> = 0 y _ST DECENAS DE MILLAR = 2

[4] La selección de sentido del eje giratorio 1 ó 2 se realiza también cuando el eje giratorio con la referencia de sentido se encuentra en la posición polar (valor de posición igual a cero).

[5] Ejemplo de codificación: giro eje a eje, secuencia de giro ZYX

Binario: 00011011; decimal: 27

Los identificadores de eje XYZ corresponden a los ejes geométricos del canal CN. Los giros alrededor de los ejes XYZ se pueden ejecutar individualmente. Por ejemplo: la secuencia de giro alrededor de ZXZ no está permitida en una llamada del CYCLE800

3.25.1.35 CYCLE801, patrón de posiciones rejilla o marco

Sintaxis

```
CYCLE801(<_SPCA>, <_SPCO>, <_STA>, <_DIS1>, <_DIS2>, <_NUM1>,
<_NUM2>, <_VARI>, <_UMODE>, <_ANG1>, <_ANG2>, <_HIDE>, <_NSP>,
<_DMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetros interno	Tipo de datos	Descripción		
1	X0	<_SPCA>	REAL	Punto de referencia para patrón de posiciones (rejilla/marco) en el 1.er eje (abs)		
2	Y0	<_SPCO>	REAL	Punto de referencia para patrón de posiciones (rejilla/marco) en el 2.er eje (abs)		
3	α0	<_STA>	REAL	Ángulo de giro básico (ángulo respecto al 1.er eje)	< 0 =	Giro en sentido horario
					> 0 =	Giro en sentido antihorario
4	L1	<_DIS1>	REAL	Distancia entre columnas (distancia de posición del 1.er eje, se introduce sin signo)		
5	L2	<_DIS2>	REAL	Distancia entre filas (distancia de posición del 2.º eje, se introduce sin signo)		
6	N1	<_NUM1>	INT	Cantidad de columnas		
7	N2	<_NUM2>	INT	Cantidad de filas		
8		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	patrón de posiciones	
					0 =	Rejilla
					1 =	Marco
				DECENAS:	reservado	
				CENTENAS:	reservado	
9		<_UMODE>	INT	reservado		
10	αX	<_ANG1>	REAL	Ángulo de cizalladura respecto al 1.er eje (posición inclinada de las filas con respecto al 1.er eje)		
					< 0 =	Medición en sentido horario (de 0 a -90 grados)
					> 0 =	Medición en sentido antihorario (de 0 a 90 grados)
11	αY	<_ANG2>	REAL	Ángulo de cizalladura respecto al 2.º eje (posición inclinada de las columnas con respecto al 2.º eje)		
					< 0 =	Medición en sentido horario (de 0 a -90 grados)
					> 0 =	Medición en sentido antihorario (de 0 a 90 grados)
12		<_HIDE>	STRING [200]	Posiciones ocultas • Máximo 198 caracteres • Indicación del número de posición consecutivo, p. ej., "1,3" (las posiciones 1 y 3 no se ejecutan)		

N.°	Máscara de parámetros	Parámetros interno	Tipo de datos	Descripción		
13		<_NSP>	INT	reservado		
14		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)

3.25.1.36 CYCLE802, posiciones libres

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1		<_XA>	INT	Alternativa para todas las posiciones X (valor decimal de 9 dígitos) Número de dígitos: 876543210 (el dígito corresponde a la posición de taladrado Xn)		
				Valor del dígito:	1 =	Absoluto (1.ª posición programada siempre absoluta)
					2 =	Incremental
2		<_YA>	INT	Alternativa para todas las posiciones Y (valor decimal de 9 dígitos) Número de dígitos: 876543210 (el dígito corresponde a la posición de taladrado Yn)		
				Valor del dígito:	1 =	Absoluto (1.ª posición programada siempre absoluta)
					2 =	Incremental
3	X0	<_X0>	REAL	1. Coord. X		
4	Y0	<_Y0>	REAL	1. Coord. Y		
5	X1	<_X1>	REAL	2. Coord. X		
6	Y1	<_Y1>	REAL	2. Coord. Y		
7	X2	<_X2>	REAL	3. Coord. X		
8	Y2	<_Y2>	REAL	3. Coord. Y		
9	X3	<_X3>	REAL	4. Coord. X		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
10	Y3	<_Y3>	REAL	4. Coord. Y
11	X4	<_X4>	REAL	5. Coord. X
12	Y4	<_Y4>	REAL	5. Coord. Y
13	X5	<_X5>	REAL	6. Coord. X
14	Y5	<_Y5>	REAL	6. Coord. Y
15	X6	<_X6>	REAL	7. Coord. X
16	Y6	<_Y6>	REAL	7. Coord. Y
17	X7	<_X7>	REAL	8. Coord. X
18	Y7	<_Y7>	REAL	8. Coord. Y
19	X8	<_X8>	REAL	9. Coord. X
20	Y8	<_Y8>	REAL	9. Coord. Y

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
21		<_VARI>	INT	Mecanizado		
				CENTENAS:	(solo para llamada desde Jobshop) (por el momento, se evalúan solo 0 y 2)	
					0 =	sin amarre de cabezal
					1 =	amarre de cabezal solo para penetración perpendicular con G00 o G01
					2 =	amarre de cabezal durante todo el mecanizado
				MILLARES:	reservado	
				DECENAS DE MILLAR:	patrón de posiciones con/sin eje giratorio – combinación de ejes (con <_VARI> CENTENAS DE MILLAR)	
					0 =	XY (solo XY sin eje giratorio, compatibilidad)
					1 =	X,Y o Z y eje giratorio: XA, YB, ZC (1 eje giratorio con eje geométrico en torno al cual gira el eje giratorio)
					2 =	XY y eje giratorio: XYA, XYB, XYC (1 eje giratorio con 1.er y 2.º eje geométrico, sin TRACYL)
				CENTENAS DE MILLAR:	Eje giratorio	
					0 =	Sin eje giratorio (solo XY, compatibilidad)
					1 =	Eje A (eje giratorio alrededor de X)
					2 =	Eje B (eje giratorio alrededor de Y)
					3 =	Eje C (eje giratorio alrededor de Z)
				DECENAS DE MILLÓN + UNIDADES DE MILLÓN:	Modelo de posición con eje giratorio – Offset (para varios ejes giratorios alrededor del mismo eje; si el índice es demasiado grande, 1.er eje)	
					00 =	1. Eje A, B o C o para compatibilidad
					01 =	2. Eje A, B o C
					...	
					19 =	20. Eje A, B o C

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
22		<_UMODE>	INT	Selección del cabezal que se amarrará: (solo para llamada desde Jobshop) (llamada ciclo de usuario CUST_TECHCYC)		
					3 =	Amarrar/soltar cabezal principal
					23 =	Amarrar/soltar contracabezal
23		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
24		<S_ABA>	INT	Alternativa para todas las posiciones AB (valor decimal de 9 dígitos) Número de dígitos: 876543210 (el dígito corresponde a la posición ABn)		
				Valor del dígito:	1 =	Absoluto (1.ª posición programada siempre absoluta)
					2 =	Incremental
25	A0	<S_AB0>	REAL	1. posición de eje giratorio para modelo de posición con eje giratorio (en combinación con <_VARI>)		
26	A1	<S_AB1>	REAL	2. posición de eje giratorio para modelo de posición con eje giratorio		
27	A2	<S_AB2>	REAL	3. posición de eje giratorio para modelo de posición con eje giratorio		
28	A3	<S_AB3>	REAL	4. posición de eje giratorio para modelo de posición con eje giratorio		
29	A4	<S_AB4>	REAL	5. posición de eje giratorio para modelo de posición con eje giratorio		
30	A5	<S_AB5>	REAL	6. posición de eje giratorio para modelo de posición con eje giratorio		
31	A6	<S_AB6>	REAL	7. posición de eje giratorio para modelo de posición con eje giratorio		
32	A7	<S_AB7>	REAL	8. posición de eje giratorio para modelo de posición con eje giratorio		
33	A8	<S_AB8>	REAL	9. posición de eje giratorio para modelo de posición con eje giratorio		

Nota

Las posiciones no necesarias de los parámetros X1/Y1/A1 a X8/Y8/A8 pueden omitirse. Sin embargo, los valores alternativos para <_XA>, <_YA> y <S_ABA> deben indicarse por completo para las 9 posiciones.

Para el modelo de posición XA, YB o ZC (un eje geométrico y eje giratorio), el eje del plano de mecanizado que no se desplaza más allá del modelo de posición (para G17 y XA, es Y) debe posicionarse antes de la llamada del ciclo.

3.25.1.37 CYCLE830, taladrado profundo 2

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	RP	<RTP>	REAL	Plano de retirada (abs)
2.	Z0	<RFP>	REAL	Punto de referencia (abs)
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, sin signo)
4	Z1	<_DP>	REAL	Profundidad de taladrado final abs/inc (ver <_AMODE> UNIDADES)
5	D	<FDEP>	REAL	1.ª profundidad de taladrado para Romper virutas/Sacar virutas, absoluta o incremental relativa al punto de referencia con punteado/sin punteado o relativa a la profundidad del taladro piloto (ver <_AMODE> DECENAS DE MILLAR)
6	DF	<_DAM>	REAL	Valor absoluto/porcentaje para cada penetración adicional, valor de degresión o porcentaje de degresión (ver <_AMODE> CENTENAS DE MILLAR)
7	DTB	<DTB>	REAL	Tiempo de espera en cada profundidad de taladrado (ver <_AMODE> DECENAS)
8	DTS	<DTS>	REAL	Tiempo de espera para Sacar virutas en punto inicial (ver <_AMODE> CENTENAS)
9	FD1	<FRF>	REAL	Porcentaje de avance en la primera penetración (ver <_AMODE> DECENAS DE MILLÓN)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
10		<VARI>	INT	Mecanizado		
				UNIDADES:	Romper/sacar virutas	
					0 =	En un corte
					1 =	Romper virutas
					2 =	Sacar virutas
					3 =	Romper virutas y sacar virutas
				DECENAS:	Retirada para sacar virutas	
					0 =	A la profundidad de taladro piloto
					1 =	A la distancia de seguridad
				CENTENAS:	inicio de corte blando	
					0 =	no
					1 =	sí
				MILLARES:	Taladro pasante	
					0 =	no
					1 =	sí
				DECENAS DE MILLAR:	punteado/taladro piloto	
					0 =	sin punteado
					1 =	con punteado
					2 =	con taladro piloto
				CENTENAS DE MILLAR:	retirada	
					0 =	A la profundidad de taladro piloto
					1 =	hasta el plano de retirada
11	V1	<_MDEP>	REAL	Penetración mínima incremental (solo en porcentaje para degresión)		
12	V2	<_VRT>	REAL	Valor absoluto de retirada incremental tras cada mecanizado (solo con Romper virutas)		
					0 =	Valor estándar 1 mm
					> 0 =	Valor de retirada variable
13	DT	<_DTD>	REAL	Tiempo de espera en profundidad de taladrado final (ver <_AMODE> MILLARES)		
14	V3	<_DIS1>	REAL	Distancia de parada previa incremental solo con Sacar virutas (ver <_AMODE> UNIDADES DE MILLÓN)		
15	FP	<S_FP>	REAL	Avance para aproximación a taladro piloto como valor o en % (en combinación con <S_AMODE2> CENTENAS)		
16		<S_SDAC2>	INT	Sentido de giro del cabezal para la aproximación		
					3 =	M3
					4 =	M4
					5 =	M5 (predeterminado)
17	SP	<S_SV2>	REAL	Aproximación con	velocidad de giro constante del cabezal (ver <S_AMODE2> DECENAS DE MILLÓN)	
	V4				velocidad de corte constante	
				Velocidad de giro del cabezal en % de la velocidad de giro de taladrado		
18	F	<S_FB>	REAL	Avance de taladrado (ver <S_AMODE2> UNIDADES)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
19		<_SDAC>	REAL	Sentido de giro del cabezal durante el taladrado		
					3 =	M3
					4 =	M4
20	S	<_SV1>	REAL	Taladrado con	velocidad de giro constante del cabezal (ver <S_AMODE2> UNIDADES DE MILLÓN)	
	V5				velocidad de corte constante	
21	SPOS	<S_SPOS>	REAL	Posición del cabezal, solo para aproximación con M5		
22	ZA	<S_ZA>	REAL	Profundidad de punteado incremental relativa a punto de referencia o absoluta (ver <S_AMODE3> UNIDADES)		
23	FA	<S_FA>	REAL	Avance de punteado como valor o en % (en combinación con <S_AMODE2> DECENAS)		
24	ZP	<S_ZP>	REAL	Taladro piloto incremental relativo a punto de referencia, absoluto o factor del diámetro de taladrado (ver <S_AMODE3> DECENAS)		
25	FS	<S_FS>	REAL	Avance de inicio de corte como valor o en % (en combinación con <S_AMODE2> MILLARES)		
26	ZS1	<S_ZS1>	REAL	Profundidad de cada inicio de corte con avance constante (inc)		
27	ZS2	<S_ZS2>	REAL	Profundidad de cada inicio de corte para aumento de avance (inc)		
28	N	<S_N>	INT	Número de pasadas de rotura de virutas antes de cada evacuación de virutas		
29	ZD	<S_ZD>	REAL	Profundidad de taladrado residual incremental referida a profundidad de taladrado final o absoluta (en combinación con <S_AMODE3> CENTENAS)		
30	FD	<S_FD>	REAL	Avance de taladrado residual como valor o en % (en combinación con <S_AMODE2> DECENAS DE MILLAR)		
31	FR	<S_FR>	REAL	Avance de retirada (en combinación con <S_AMODE2> CENTENAS DE MILLAR)		
32		<S_SDAC3>	INT	Sentido de giro del cabezal durante la retirada		
					3 =	M3
					4 =	M4
					5 =	M5
33	SR	<S_SV3>	REAL	Retirada con	velocidad de giro constante del cabezal (ver <S_AMODE2> CENTENAS DE MILLÓN)	
	V6				velocidad de corte constante	
				Velocidad de giro del cabezal en % de la velocidad de giro de taladrado		
34	Refrigerante CON	<S_CON>	STRING[10]	Refrigerante CON, comando M o llamada de subprograma		
35	Refrigerante DES	<S_COFF>	STRING[10]	Refrigerante DES, comando M o llamada de subprograma		
36		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	Profundidad de taladrado relativa a punta/vástago	
					0 =	Punta
					1 =	Vástago

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
37		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:	reservado	
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: Simple
38		<_AMODE>	INT	Modo alternativo 1		
				UNIDADES:	profundidad de taladrado = profundidad de taladrado final Z1 abs/inc	
					0 =	Incremental
					1 =	Absoluto
				DECENAS:	Tiempo de espera en cada profundidad de taladrado DTB en segundos/vueltas	
					0 =	En segundos
					1 =	En vueltas
				CENTENAS:	Tiempo de espera para sacar virutas DTS, en segundos/vueltas	
					0 =	En segundos
					1 =	En vueltas
				MILLARES:	Tiempo de espera en la profundidad de taladrado final DT en segundos/vueltas	
					0 =	En segundos
					1 =	En vueltas
				DECENAS DE MILLAR:	1.ª profundidad de taladrado D abs/inc	
					0 =	Incremental
					1 =	Absoluto
				CENTENAS DE MILLAR:	valor/porcentaje DF para cada penetración adicional (degresión)	
					0 =	Valor absoluto
					1 =	Porcentaje (0,001 hasta 100%)
				UNIDADES DE MILLÓN:	distancia de parada previa V3 automática/manual	
					0 =	Automática (se calcula en el ciclo)
					1 =	Manual (valor programado)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
39		<S_AMODE2>	INT	Modo alternativo 2		
				UNIDADES:	UNIDADES: Avance de taladrado F	
					0 =	F/min
					1 =	F/vuelta
				DECENAS:	Evaluación avance de punteado FA	
					0 =	En % del avance de taladrado
					1 =	F/min
					2 =	F/vuelta
				CENTENAS:	Evaluación avance para aproximación en taladro piloto FP	
					0 =	En % del avance de taladrado
					1 =	F/min
					2 =	F/vuelta
				MILLARES:	Evaluación avance inicio de corte FS	
					0 =	En % del avance de taladrado
					1 =	F/min
					2 =	F/vuelta
				DECENAS DE MILLAR:	Evaluación avance taladro pasante FD	
					0 =	En % del avance de taladrado
					1 =	F/min
					2 =	F/vuelta
				CENTENAS DE MILLAR:	Avance retirada FR	
					0 =	F/min
					1 =	Rápido
				UNIDADES DE MILLÓN:	Taladrado - velocidad de giro del cabezal/velocidad de corte (S/V5)	
					0 =	Velocidad de giro constante del cabezal
					1 =	Velocidad de corte constante
				DECENAS DE MILLÓN:	Aproximación con velocidad de giro del cabezal/velocidad de corte (SP/V4)	
					0 =	Velocidad de giro constante del cabezal
					1 =	Velocidad de corte constante
					2 =	Velocidad de giro del cabezal en % de la velocidad de giro de taladrado
				CENTENAS DE MILLÓN:	Retirada - velocidad de giro del cabezal/ velocidad de corte (SR/V6)	
					0 =	Velocidad de giro constante del cabezal
					1 =	Velocidad de corte constante

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
					2 =	Velocidad de giro del cabezal en % de la velocidad de giro de taladrado
40		<S_AMODE3>	INT	Modo alternativo 3		
				UNIDADES:	profundidad de punteado ZA abs/inc	
					0 =	Incremental
					1 =	Absoluto
				DECENAS:	Profundidad del taladro piloto ZP	
					0 =	Incremental
					1 =	Absoluto
					2 =	Factor del diámetro de taladrado
				CENTENAS:	profundidad de taladrado residual ZD inc/abs	
					0 =	Incremental
					1 =	Absoluto
41	ZPV	<S_ZPV>	REAL	Distancia de parada previa incremental relativa a la profundidad del taladro piloto		

3.25.1.38 CYCLE832, High Speed Settings

Sintaxis

```
CYCLE832(<S_TOL>, <S_TOLM>, <S_OTOL>)
```

Nota

El CYCLE832 no descarga al fabricante de la máquina de las tareas de optimización necesarias durante la puesta en marcha de ésta. Esto comprende a la optimización de los ejes que participan en el mecanizado y los ajustes de la NCU (control anticipativo, limitación de tirones, etc.).

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	tolerancia	<S_TOL>	REAL	Tolerancia de contorno La tolerancia de contorno hace referencia a la tolerancia de los ejes geométricos.		
2		<S_TOLM>	INT	Tipo de mecanizado (tecnología)		
				UNIDADES:		
					0 =	Cancelación
					1 =	Acabado (Finish)
					2 =	Preacabado (Semifinish)
					3 =	Desbaste (Rough)
					4 =	Acabado fino (Precision)
				DECENAS:		
					0 =	Compatibilidad[1] o sin tolerancia de orientación
					1 =	Tolerancia de orientación en el parámetro <S_OTOL>
				CENTENAS ... CENTENAS DE MILLAR	Ocupadas por compatibilidad	
				UNIDADES DE MILLÓN:		
					0 =	Compatibilidad. Se utiliza automáticamente la mejor función de matricería y moldes disponible: • Opción Top Surface no activa: ⇒ Advanced Surface • Opción Top Surface activa: ⇒ Top Surface con alisado
					1 =	Top Surface sin alisado
					2 =	Top Surface con alisado
3	Tolerancia ORI	<S_OTOL>	REAL	Tolerancia de orientación o identificador de versión CYCLE832 Parámetro de tolerancia para la orientación de la herramienta. Se necesita en la ejecución de un programa de mecanizado de alta velocidad en máquinas con transformada de orientación dinámica (p. ej., mecanizado de 5 ejes). El parámetro <S_OTOL> **debe** programarse. Esto es extensible también a aplicaciones en máquinas de 3 ejes con programas sin orientación de la herramienta (<S_OTOL> = 1).		

[1] Tolerancia de orientación a partir de los datos de operador de ciclos SD55451...SD55454 (tolerancia de orientación para modo dinámico...) o SD55445...SD55449 (tolerancia de contorno para modo dinámico...) multiplicada por el factor de SD55441...SD55444.
Más información: Manual de puesta en marcha SINUMERIK Operate

Entrada de texto explícito

Para mejorar la legibilidad de la llamada de ciclos, el parámetro <S_TOLM> (tipo de mecanizado) puede introducirse también como texto explícito. Los textos explícitos no dependen del idioma. Se admiten las entradas siguientes:

_OFF	para	0	Cancelación
_FINISH	para	1	Acabado
_SEMIFIN	para	2	Preacabado
_ROUGH	para	3	Desbaste
_PRECISION	para	4	Acabado fino
_ORI_FINISH	para	11	Acabado con introducción de una tolerancia de orientación
_ORI_SEMIFIN	para	12	Preacabado con introducción de una tolerancia de orientación
_ORI_ROUGH	para	13	Desbaste con introducción de una tolerancia de orientación
_ORI_PRECISION	para	14	Acabado fino con introducción de una tolerancia de orientación
_TOP_SURFACE_SMOOTH_OFF	para	1000000	Top Surface sin alisado
_TOP_SURFACE_SMOOTH_ON	para	2000000	Top Surface con alisado

Para la entrada de texto explícito en Top Surface, los textos explícitos se combinan como en el siguiente ejemplo:

```
CYCLE832(0.1, _TOP_SURFACE_SMOOTH_OFF+_ORI_FINISH, 1)
```

Nota

Los textos explícitos se basan en los nombres de funciones del grupo G 59 (modo dinámico para interpolación de trayectoria). Estos textos explícitos permiten separar claramente la aplicación de máquinas de 3 ejes y máquinas con transformada de orientación de varios ejes (TRAORI).

Cancelación de CYCLE832

Al cancelar CYCLE832 debe transferirse el parámetro <S_TOL> con cero.

Ejemplo: `CYCLE832(0,0,1)`

La sintaxis `CYCLE832()` también se permite para cancelar CYCLE832.

Ejemplos

Ejemplo 1: CYCLE832 en máquina de 3 ejes sin transformada de orientación

a) Llamada de ciclo con entrada de texto explícito

Código de programa	Comentarios
G710	; Sistema de acotado métrico.
CYCLE832(0.004,_FINISH,1)	; Llamada CYCLE832 con: Tolerancia de contorno = 0,004 mm, tipo de mecanizado: acabado
...	; Ejecución de un programa de mecanizado de alta velocidad

b) Llamada de ciclo sin entrada de texto explícito

Código de programa	Comentarios
G710	; ver arriba
CYCLE832(0.004,1,1)	; ver arriba
...	; ver arriba

Ejemplo 2: CYCLE832 en máquina de 5 ejes con transformada de orientación

a) Llamada de ciclo y cancelación con entrada de texto explícito

Código de programa	Comentarios
G710	; Sistema de acotado métrico.
TRAORI	; Activar transformada de orientación.
CYCLE832(0.3,_ORI_ROUGH,0.8)	; Llamada CYCLE832 con: Tolerancia de contorno = 0,3 mm, tipo de mecanizado: Desbaste con entrada de una tolerancia de orientación, tolerancia de orientación = 0,8 grados
...	; Ejecución de un programa de mecanizado de alta velocidad
CYCLE832(0,_OFF,1)	; Tolerancia de contorno = 0, tipo de mecanizado: Cancelación CYCLE832, tolerancia de orientación = 0 grados

b) Llamada de ciclo y cancelación sin entrada de texto explícito

Código de programa	Comentarios
G710	; ver arriba
TRAORI	; ver arriba
CYCLE832(0.3,13,0.8)	; ver arriba
...	; ver arriba
CYCLE832(0,0,1)	; ver arriba

3.25.1.39 CYCLE840, roscado con mandril de compensación

Sintaxis

```
CYCLE840(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDR>, <SDAC>,
<ENC>, <MPIT>, <PIT>, <_AXN>, <_PITA>, <_TECHNO>, <_PITM>, <_PTAB>,
<_PTABA>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción			
1	RP	<RTP>	REAL	Plano de retirada (abs)			
2	Z0	<RFP>	REAL	Punto de referencia (abs)			
3	SC	<SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)			
4	Z1	<DP>	REAL	Profundidad de taladrado (abs), ver <_AMODE>			
5	Z1	<DPR>	REAL	Profundidad de taladrado (inc), ver <_AMODE>			
6	DT	<DTB>	REAL	Tiempo de espera en segundos en la profundidad de taladrado/en la distancia de seguridad tras la retirada, ver <ENC>			
7		<SDR>	INT	Sentido de giro para la retirada			
8	SDE	<SDAC>	INT	Sentido de giro al final del ciclo			
9		<ENC>	INT	Roscado con macho con encóder de cabezal (G33)/Roscado con macho sin encóder de cabezal (G63)			
					0 =	con encóder de cabezal	- Paso de <MPIT>/<PIT> - sin DT
					20 =	con encóder de cabezal	- Paso de <MPIT>/<PIT> - con DT tras retirada a distancia de seguridad
					11 =	sin encóder de cabezal	- Paso de <MPIT>/<PIT> - con DT a profundidad de taladrado
					1 =	sin encóder de cabezal	- Paso a partir del avance programado - con DT a profundidad de taladrado (avance = velocidad de giro · paso)
10		<MPIT>	REAL	Tamaño de rosca sólo para "ISO métrica" (el paso se calcula internamente durante el tiempo de ejecución) Rango de valores: de 3 a 48 (para M3 a M48), alternativamente a <PIT>			
11		<PIT>	REAL	Paso de rosca como valor; unidad de medida, ver <_PITA> Rango de valores: > 0, alternativamente a MPIT			

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción			
12		<_AXN>	INT	Eje de taladrado	0 =	3. eje geométrico	
					1 =	1. eje geométrico	
					2 =	2. eje geométrico	
					≥ 3 =	3. eje geométrico	
13		<_PITA>	INT	Unidad de medida del paso de rosca (evaluación <PIT> y <MPIT>)			
					0 =	Paso en mm	- Evaluación <MPIT>/<PIT>
					1 =	Paso en mm	- Evaluación <PIT>
					2 =	Paso en TPI	- Evaluación <PIT> (filetes de rosca por pulgada)
					3 =	Paso en pulgadas	- Evaluación <PIT>
					4 =	MÓDULO	- Evaluación <PIT>
14		<_TECHNO>	INT	Tecnología[1]			
				UNIDADES:	Comportamiento de parada precisa		
					0 =	Comportamiento de parada precisa como antes de la llamada activa de ciclo	
					1 =	Parada precisa G601	
					2 =	Parada precisa G602	
					3 =	Parada precisa G603	
				DECENAS:	Control anticipativo		
					0 =	Con/sin control anticipativo como antes de la llamada activa de ciclo	
					1 =	Con control anticipativo FFWON	
					2 =	Sin control anticipativo FFWOF	
15		<_PITM>	STRING[15]	Cadena de caracteres como marca para la introducción del paso de rosca[2]			
16		<_PTAB>	STRING[5]	Cadena de caracteres para la tabla de roscas ("", "ISO", "BSW", "BSP", "UNC")[2]			
17		<_PTABA>	STRING[20]	Cadena de caracteres para la selección en la tabla de roscas (p. ej. "M 10", "M 12"...)[2]			
18		<_GMODE>	INT	reservado			

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
19		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:	modo de compatibilidad (solo para la máscara de entrada que se decompilará) si MD 52216 bit 0 = 1[1]	
					0 =	se muestran los parámetros tecnológicos (compatibilidad): los parámetros TECHNO están activos
					1 =	no se muestran los parámetros tecnológicos: está activa la tecnología "como programado antes de la llamada de ciclo"
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple
20		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad de taladrado Z1 (abs/inc)	
					0 =	Compatibilidad, a partir de programación <DP>/<DPR>
					1 =	Incremental
					2 =	Absoluto

[1] Los campos de tecnología pueden estar ocultos en función del dato de operador SD52216 MCS_FUNCTION_MASK_DRILL

[2] Los parámetros 15, 16 y 17 se utilizan solamente al seleccionar la rosca en las tablas de roscas de la máscara de entrada. No se puede acceder a las tablas de roscas mediante la definición de ciclo en el tiempo de ejecución del ciclo.

3.25.1.40 CYCLE899, ranura abierta

Sintaxis

```
CYCLE899(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_LENG>, <_WID>, <_PA>,
<_PO>, <_STA>, <_MID>, <_MIDA>, <_FAL>, <_FALD>, <_FFP1>, <_CDIR>,
<_VARI>, <_GMODE>, <_DMODE>, <_AMODE>, <_UMODE>, <_FS>, <_ZFS>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	RP	<_RTP>	REAL	Plano de retirada (abs)		
2	Z0	<_RFP>	REAL	Punto de referencia del eje de herramienta (abs)		
3	SC	<_SDIS>	REAL	Distancia de seguridad (aditiva al punto de referencia, se introduce sin signo)		
4	Z1	<_DP>	REAL	Profundidad de ranura (abs/inc), ver <_AMODE>		
5	L	<_LENG>	REAL	Longitud de la ranura (inc)		
6	W	<_WID>	REAL	Anchura de la ranura (inc)		
7	X0	<_PA>	REAL	Punto de referencia/posición inicial 1.er eje (abs)		
8	Y0	<_PO>	REAL	Punto de referencia/posición inicial 2.º eje (abs)		
9	α0	<_STA>	REAL	Ángulo de giro respecto al 1.er eje		
10	DZ	<_MID>	REAL	Profundidad de penetración máxima (inc), sólo para fresado trocoidal		
11	DXY	<_MIDA>	REAL	Penetración máxima del plano, ver <_AMODE>		
12	UXY	<_FAL>	REAL	Creces para acabado en el plano		
13	UZ	<_FALD>	REAL	Creces para acabado profundidad		
14	F	<_FFP1>	REAL	Avance		
15		<_CDIR>	INT	Sentido de fresado		
				UNIDADES:		
					0 =	En concordancia
					1 =	En oposición
					4 =	Alternadamente
16		<_VARI>	INT	Mecanizado		
				UNIDADES:		
					1 =	Desbaste
					2 =	Acabado
					3 =	Acabado fondo
					4 =	Acabado borde
					5 =	Preacabado
					6 =	Achaflanado
				DECENAS:	reservado	
				CENTENAS:	reservado	
				MILLARES:		
					1 =	Fresado trocoidal
					2 =	Fresado en plongée (plunging)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
17		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	selección del mecanizado/sólo cálculo del punto inicial	
					1 =	Mecanizado normal
				MILLARES:	acotado a través de centro/borde	
					0 =	Acotado a través de centro
					1 =	Acotado a través de borde "izquierdo" (sentido "-" del 1.er eje)
					2 =	Acotado a través de borde "derecho" (sentido "+" del 1.er eje)
18		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	---	reservado
				CENTENAS:	---	reservado
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple
19		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	profundidad de ranura Z1	
					0 =	Absoluto
					1 =	Incremental
				DECENAS:	Unidad para la penetración del plano (<_MIDA>) DXY	
					0 =	mm
					1 =	% del diámetro de la herramienta
				CENTENAS:	profundidad de penetración en el achaflanado ZFS	
					0 =	Absoluto
					1 =	Incremental
20		<_UMODE>	INT	reservado		
21	FS	<_FS>	REAL	Ancho del chaflán (inc)		
22	ZFS	<_ZFS>	REAL	Profundidad de penetración (punta de la herramienta) en el achaflanado (abs/inc), ver <_AMODE>		

3.25.1.41 CYCLE930, ranurado

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1	X0	<_SPD>	REAL	Punto de referencia en el eje de refrentado (siempre diámetro)		
2	Z0	<_SPL>	REAL	Punto de referencia en el eje longitudinal		
3	B1	<_WIDG>	REAL	Ancho de ranurado abajo		
4	B2	<_WIDG2>	REAL	Ancho de ranurado arriba (sólo para la interfaz)		
5	T1	<_DIAG>	REAL	Profundidad de ranura en el punto de referencia, con abs y torneado longitudinal = diámetro; si no, inc		
6	T2	<_DIAG2>	REAL	Profundidad de ranura con respecto al punto de referencia (sólo para la interfaz), con abs y torneado longitudinal = diámetro; si no, inc		
7	α0	<_STA>	REAL	Ángulo del corte oblicuo (-180 ≤ <_STA> ≤ 180)		
8	α1	<_ANG1>	REAL	Ángulo de flanco 1 (0 ≤ <_ANG1> < 90) en el lado de la ranura determinado por el punto de referencia		
9	α2	<_ANG2>	REAL	Ángulo de flanco 2 (0 ≤ <_ANG2> < 90) con respecto al punto de referencia		
10	R1/FS1	<_RCO1>	REAL	Radio de redondeo o ancho del chaflán 1, exterior en el punto de referencia		
11	R2/FS2	<_RCI1>	REAL	Radio de redondeo o ancho del chaflán 2, interior en el punto de referencia		
12	R3/FS3	<_RCI2>	REAL	Radio de redondeo o ancho del chaflán 3, interior frente al punto de referencia		
13	R4/FS4	<_RCO2>	REAL	Radio de redondeo o ancho del chaflán 4, exterior frente al punto de referencia		
14	U	<_FAL>	REAL	Creces para acabado en X y Z, ver <_VARI> (DECENAS DE MILLAR) (se introduce sin signo)		
15	D	<_IDEP1>	REAL	Máxima penetración en profundidad (se introduce sin signo)		
					0 =	1. 0 = 1.er corte directamente en toda la profundidad
					> 0 =	1.er corte <_IDEP1>, 2.º corte 2 · <_IDEP1>, etc.
16	SC	<_SDIS>	REAL	Distancia de seguridad (se introduce sin signo)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
17		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	reservado	
				DECENAS:	mecanizado tecnológico	
					1 =	Desbaste
					2 =	Acabado
					3 =	Desbaste y acabado
				CENTENAS:	Posición longitudinal/transversal exterior/interior +Z/+Z o bien +X/-X	
					1 =	Longitudinal/exterior +Z
					2 =	Transversal/interior -X
					3 =	Longitudinal/interior +Z
					4 =	Transversal/interior +X
					5 =	Longitudinal/exterior -Z
					6 =	Transversal/exterior -X
					7 =	Longitudinal/interior -Z
					8 =	Transversal/exterior +X
				MILLARES:	Posición punto de referencia	
					0 =	Punto de referencia arriba
					1 =	Punto de referencia abajo
				DECENAS DE MILLAR:	definición de cómo actúan las creces para acabado	
					0 =	Creces para acabado U paralelas al contorno
					1 =	Creces para acabado UX y UZ separadas
18		<_DN>	INT	Número D para el 2.º filo de la herramienta		
					> 0 =	Número D para corrección de herramienta del 2.º filo de la cuchilla de ranurar
					0 =	Sin el 2.º filo programado
19	N	<_NUM>	INT	Número de ranuras (0 = 1 ranura)		
20	DP	<_DBH>	REAL	Distancia de las ranuras (sólo necesario si <_NUM> > 1)		
21	F	<_FF1>	REAL	Avance		
22		<_NR>	INT	La identificación para la forma de la ranura equivale a los pulsadores de menú verticales para seleccionar la forma		
					0 =	Flancos de 90° sin chaflanes/ redondeos
					1 =	Flancos oblicuos con chaflanes/ redondeos (sin α0)
					2 =	Como 1, pero en cono (con α0)
23	UX	<_FALX>	REAL	Creces para acabado en eje X, ver <_VARI> (DECENAS DE MILLAR) (se introduce sin signo)		
24	UZ	<_FALZ>	REAL	Creces para acabado en eje Z, ver <_VARI> (DECENAS DE MILLAR) (se introduce sin signo)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
25		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
26		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	acotado de profundidad (sólo para la interfaz)	
					0 =	En el punto de referencia
					1 =	Frente al punto de referencia
				DECENAS:	profundidad	
					0 =	Absoluta
					1 =	Incremental
				CENTENAS:	acotado de anchura (sólo para la interfaz)	
					0 =	En el diámetro exterior (arriba)
					1 =	En el diámetro interior (abajo)
				MILLARES:	Radio/chaflán 1 (<_RCO1>)	
					0 =	Radio
					1 =	Chaflán
				DECENAS DE MILLAR:	Radio/chaflán 2 (<_RCI1>)	
					0 =	Radio
					1 =	Chaflán
				CENTENAS DE MILLAR:	Radio/chaflán 3 (<_RCI2>)	
					0 =	Radio
					1 =	Chaflán
				UNIDADES DE MILLÓN:	Radio/chaflán 4 (<_RCO2>)	
					0 =	Radio
					1 =	Chaflán

3.25.1.42 CYCLE940, garganta forma E y F/garganta rosca

Con el ciclo CYCLE940 se pueden programar diferentes gargantas. Algunas de ellas se diferencian considerablemente en la parametrización.

Las columnas adicionales de la tabla muestran qué parámetros se necesitan con qué formas de garganta. Equivalen a los pulsadores de menú de selección verticales de la máscara de ciclos:

- E: garganta Forma E
- F: garganta Forma F

- A-D: garganta rosca DIN (formas A-D)
- T: garganta rosca (libre definición de la forma)

Sintaxis

```
CYCLE940(<_SPD>, <_SPL>, <_FORM>, <_LAGE>, <_SDIS>, <_FFP>, <_VARI>,
<_EPD>, <_EPL>, <_R1>, <_R2>, <_STA>, <_VRT>, <_MID>, <_FAL>,
<_FALX>, <_FALZ>, <_PITI>, <_PTAB>, <_PTABA>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Progr. con forma				Descripción		
				E	F	A-D	T			
1	X0	<_SPD>	REAL	x	x	x	x	Punto de referencia en el eje de refrentado (siempre diámetro)		
2	Z0	<_SPL>	REAL	x	x	x	x	Punto de referencia del eje longitudinal (abs)		
3	FORM	<_FORM>	CHAR	x	x	x	x	Forma de la garganta (letras mayúsculas, p. ej. "T") Selección de la tabla de la que deben tomarse los valores de destalonado		
									A =	Exterior, referencia DIN76, A = normal
									B =	Exterior, referencia DIN76, B = corta
									C =	Interior, referencia DIN76, C = normal
									D =	Interior, referencia DIN76, D = corta
									E =	Referencia DIN509
									F =	Referencia DIN509
									T =	Forma libre
4	LAGE	<_LAGE>	INT	x	x	x	x	Posición de la garganta (paralela a Z)	0 =	Exterior +Z: ____|
									1 =	Exterior -Z: |____/
									2 =	Interior +Z: /-----|
									3 =	Interior -Z: |-----\
5	SC	<_SDIS>		x	x	x	x	Distancia de seguridad (inc)		
6	F	<_FFP>		x	x	x	x	Avance de mecanizado (mm/vuelta)		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Progr. con forma				Descripción			
7		<_VARI>	INT	-	-	x	x	Tipo de mecanizado			
								UNIDADES:	Mecanizado		
									1 =	Desbaste	
									2 =	Acabado	
									3 =	Desbaste + acabado	
								DECENAS:	estrategia de mecanizado		
									0 =	Paralelo al contorno	
									1 =	Longitudinal	
				Las gargantas de la forma E y F siempre se mecanizan de una pasada como el acabado.							
8	X1	<_EPD>		x	x	-	-	Creces X (abs/inc), ver <_AMODE>			
				-	-	-	x	Profundidad de destalonado (abs/inc), ver <_AMODE>			
9	Z1	<_EPL>		-	x	-	-	Creces Z			
				-	-	-	x	Ancho de destalonado (abs/inc), ver <_AMODE>			
10	R1	<_R1>		-	-	-	x	Radio de redondeo en el bisel			
11	R2	<_R2>		-	-	-	x	Radio de redondeo en la esquina			
12	α	<_STA>		-	-	x	x	Ángulo de penetración			
13	VX	<_VRT>		x	x	-	-	Avance transversal X (abs/inc), ver <_AMODE>			
				-	-	x	x	Avance transversal X en acabado (abs/inc), ver <_AMODE>			
14	D	<_MID>		-	-	x	x	Penetración en profundidad			
15	U	<_FAL>		-	-	x	x	Creces para acabado paralelas al contorno, ver <_AMODE>			
16	UX	<_FALX>		-	-	x	x	Creces para acabado X			
17	UZ	<_FALZ>		-	-	x	x	Creces para acabado Z			
18	P	<_PITI>	INT	-	-	x	-	La selección del paso, forma A-D, corresponde a 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	-	-	Selección de radio/profundidad, forma 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]					Cadena de caracteres para la tabla de roscas ("", "ISO", "BSW", "BSP", "UNC") (sólo para la interfaz)			
20		<_PTABA>	STRING [20]					Cadena de caracteres para la selección en la tabla de roscas (p. ej. "M 10", "M 12"...) (sólo para la interfaz)			

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Progr. con forma	Descripción		
21		<_DMODE>	INT		Modo de visualización		
				x x x x	UNIDADES:	Plano de mecanizado G17/G18/G19	
						0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
						1 =	G17 (solo activo en el ciclo)
						2 =	G18 (solo activo en el ciclo)
						3 =	G19 (solo activo en el ciclo)
22		<_AMODE>	INT		Modo alternativo		
				x x - x	UNIDADES:	Parámetro <_EPD> Creces X o Profundidad de destalonado	
						0 =	Absoluto (siempre diámetro)
						1 =	Incremental
				x x - x	DECENAS:	Parámetro <_EPL> Creces Z o Ancho de destalonado	
						0 =	Absoluto
						1 =	Incremental
				x x x x	CENTENAS:	Parámetro <_VRT> Avance transversal X	
						0 =	Absoluto (siempre diámetro)
						1 =	Incremental
				- - x x	MILLARES:	Creces para acabado	
						0 =	Creces para acabado paralelas al contorno (<_FAL>)
						1 =	Creces para acabado separadas (<_FALX>/<_FALZ>)

3.25.1.43 CYCLE951, desbaste

Sintaxis

```
CYCLE951(<_SPD>, <_SPL>, <_EPD>, <_EPL>, <_ZPD>, <_ZPL>, <_LAGE>,
<_MID>, <_FALX>, <_FALZ>, <_VARI>, <_RF1>, <_RF2>, <_RF3>, <_SDIS>,
<_FF1>, <_NR>, <_DMODE>, <_AMODE>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	X0	<_SPD>	REAL	Punto de referencia (abs, siempre diámetro)
2	Z0	<_SPL>	REAL	Punto de referencia (abs)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
3	X1	<_EPD>	REAL	Punto final		
4	Z1	<_EPL>	REAL	Punto final		
5	XM α1 α2	<_ZPD>	REAL	Punto intermedio, ver <_DMODE> (DECENAS)		
6	ZM α1 α2	<_ZPL>	REAL	Punto intermedio, ver <_DMODE> (DECENAS)		
7	Posición	<_LAGE>	INT	Posición de la esquina de desbaste	0 =	Exterior/trasera
					1 =	Exterior/delantera
					2 =	Interior/trasera
					3 =	Interior/delantera
8	D	<_MID>	REAL	Penetración en profundidad máxima durante la entrada		
9	UX	<_FALX>	REAL	Creces para acabado en X		
10	UZ	<_FALZ>	REAL	Creces para acabado en Z		
11		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	dirección de desbaste (longitudinal o transversal) en el sistema de coordenadas	
					1 =	Longitudinal
					2 =	Transversal
				DECENAS:		
					1 =	Desbaste hasta creces para acabado
					2 =	Acabado
				CENTENAS:	Reservado	
				MILLARES:	Reservado	
				DECENAS DE MILLAR:	Reservado	
12	R1/FS1	<_RF1>	REAL	Radio de redondeo o ancho del chaflán 1, ver <_AMODE> (DECENAS DE MILLAR)		
13	R2/FS2	<_RF2>	REAL	Radio de redondeo o ancho del chaflán 2, ver <_AMODE> (CENTENAS DE MILLAR)		
14	R3/FS3	<_RF3>	REAL	Radio de redondeo o ancho del chaflán 3, ver <_AMODE> (UNIDADES DE MILLÓN)		
15	SC	<_SDIS>	REAL	Distancia de seguridad		
16	F	<_FF1>	REAL	Avance para desbaste/acabado		
17		<_NR>	INT	Identificación del tipo de desbaste (corresponde al pulsador de menú vertical para seleccionar la forma):		
					0 =	Desbaste 1, esquina de 90 grados sin chaflanes/redondeos
					1 =	Desbaste 2, esquina de 90 grados con chaflanes/redondeos
					2 =	Desbaste 3, cualquier esquina con chaflanes/redondeos

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
18		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	Forma de entrada <_ZPD>/<_ZPL>	
					0 =	Xm/Zm
					1 =	Xm/α1
					2 =	Xm/α2
					3 =	α1/Zm
					4 =	α2/Zm
					5 =	α1/α2
21		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	punto intermedio en X	
					0 =	Absoluto, valor del eje de refrentado en el diámetro
					1 =	Incremental, valor del eje de refrentado en el radio
				DECENAS:	punto intermedio en Z	
					0 =	Absoluto
					1 =	Incremental
				CENTENAS:	punto final en X	
					0 =	Absoluto, valor del eje de refrentado en el diámetro
					1 =	Incremental, valor del eje de refrentado en el radio
				MILLARES:	punto final en Z	
					0 =	Absoluto
					1 =	Incremental
				DECENAS DE MILLAR:	radio/chaflán 1	
					0 =	Radio
					1 =	Chaflán
				CENTENAS DE MILLAR:	radio/chaflán 2	
					0 =	Radio
					1 =	Chaflán
				UNIDADES DE MILLÓN:	radio/chaflán 3	
					0 =	Radio
					1 =	Chaflán

3.25.1.44 CYCLE952, desbastar/desbastar resto/ranurar/ranurar resto/ranurar derecha-izquierda/ ranurar derecha-izquierda resto

Sintaxis

```
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>)
```

Parámetros

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
1	PRG	<_PRG>	STRING[100]	Nombre del programa de desbaste
2	CON	<_CON>	STRING[100]	Nombre del programa desde el que se lee el contorno de la pieza en bruto actualizado (con mecanizado residual)
3	CONR	<_CONR>	STRING[100]	Nombre del programa en el que se escribe el contorno de la pieza en bruto actualizado (ver <_AMODE> DECENAS DE MILLAR)

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
4		<_VARI>	INT	Tipo de mecanizado		
				UNIDADES:	tipo de desbaste	
					1 =	Longitudinal
					2 =	Transversal
					3 =	Paralelo al contorno
				DECENAS:	Mecanizado tecnológico (ver <_GMODE> CENTENAS)	
					1 =	Desbaste
					2 =	Acabado
					3 =	reservado
					4 =	Desbaste bicanal
					5 =	Acabado bicanal
				CENTENAS:	Sentido de mecanizado	
					1 =	Sentido de mecanizado X -
					2 =	Sentido de mecanizado X +
					3 =	Sentido de mecanizado Z -
					4 =	Sentido de mecanizado Z +
				MILLARES:	dirección de penetración	
					1 =	Exterior X -
					2 =	Interior X +
					3 =	Lado frontal Z -
					4 =	Lado posterior Z +
				DECENAS DE MILLAR:	definición de cómo actúan las creces para acabado	
					0 =	Creces para acabado UX y UZ separadas
					1 =	Creces para acabado U paralelas al contorno
				CENTENAS DE MILLAR:	repasado	
					0 =	Compatibilidad, repasado automático
					1 =	Con seguimiento en el contorno
					2 =	Sin repasado
					3 =	Repasado automático
				UNIDADES DE MILLÓN:	destalonados	
					0 =	El dígito no se evalúa con ranurado, - resto y ranurado dcha./ izqda. - resto
					1 =	Mecanizar destalonados
					2 =	No mecanizar destalonados
				DECENAS DE MILLÓN:	detrás/delante de centro de giro	
					0 =	Mecanizado delante del centro de giro
					1 =	reservado

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
5	F	<_F>	REAL	Avance para desbaste/acabado		
	FZ			Avance abscisa ranurado derecha/izquierda		
6	FR	<_FR>	REAL	Avance para penetrar en destalonados, desbaste		
	FX			Avance ordenada ranurado derecha/izquierda		
7	RP	<_RP>	REAL	Plano de retirada con mecanizado interior (abs., siempre diámetro)		
8	D	<_D>	REAL	Penetración en desbaste (ver <_AMODE> UNIDADES)		
9	DX	<_DX>	REAL	Penetración X (ver <_AMODE> UNIDADES)		
10	DZ	<_DZ>	REAL	Penetración Z (ver <_AMODE> UNIDADES)		
11	UX	<_UX>	REAL	Creces para acabado X (ver <_VARI> DECENAS DE MILLAR)		
12	UZ	<_UZ>	REAL	Creces para acabado Z (ver <_VARI> DECENAS DE MILLAR)		
13	U	<_U>	REAL	Creces para acabado paralelas al contorno (ver <_VARI> DECENAS DE MILLAR)		
14	U1	<_U1>	REAL	Creces para acabado adicionales en el acabado (ver <_AMODE> MILLARES)		
15	BL	<_BL>	INT	Definición de pieza en bruto	1 =	Cilindro con creces
					2 =	Creces respecto a contorno de pieza acabada
					3 =	El contorno de la pieza en bruto está indicado
16	XD	<_XD>	REAL	Definición de pieza en bruto X (ver <_AMODE> CENTENAS DE MILLAR)		
17	ZD	<_ZD>	REAL	Definición de pieza en bruto Z (ver <_AMODE> UNIDADES DE MILLÓN)		
18	XA	<_XA>	REAL	Límite 1 X (abs., siempre diámetro)		
19	ZA	<_ZA>	REAL	Límite 1 Z (abs.)		
20	XB	<_XB>	REAL	Límite 2 X (ver <_AMODE> DECENAS DE MILLÓN)		
21	ZB	<_ZB>	REAL	Límite 2 Z (ver <_AMODE> CENTENAS DE MILLÓN)		
22	XDA	<_XDA>	REAL	Límite de penetración 1 de la posición de penetración 1 en el lado frontal (abs., siempre diámetro)		
23	XDB	<_XDB>	REAL	Límite de penetración 2 de la posición de penetración 1 en el lado frontal (abs., siempre diámetro)		
24	N	<_N>	INT	Número de entallas (ranuras)		
25	DP	<_DP>	REAL	Separación entre ranuras	Ranura longitudinal: paralela al eje Z	
					Ranura transversal: paralela al eje X	
26	DI	<_DI>	REAL	Distancia para la interrupción del avance	0 =	Sin interrupción
					> 0 =	Con interrupción
27	SC	<_SC>	REAL	Distancia de seguridad para esquivar obstáculos, incremental		
28	D2	<_DN>	INT	Número D para 2.º filo si no está programado ⇒ D+1		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
29		<_GMODE>	INT	Modo geométrico (evaluación de los valores geométricos programados)		
				UNIDADES:	reservado	
				DECENAS:	reservado	
				CENTENAS:	selección del mecanizado/sólo cálculo del punto inicial	
					0 =	Mecanizado normal (no se necesita el modo de compatibilidad)
					1 =	Mecanizado normal
					2 =	Calcular la posición inicial, sin mecanizado (sólo para la llamada desde ShopMill/ShopTurn)
				MILLARES:	limitación	
					0 =	no
					1 =	sí
				DECENAS DE MILLAR:	introducir límite 1 X	
					0 =	no
					1 =	sí
				CENTENAS DE MILLAR:	introducir límite 2 X	
					0 =	no
					1 =	sí
				UNIDADES DE MILLÓN:	introducir límite 1 Z	
					0 =	no
					1 =	sí
				DECENAS DE MILLÓN:	introducir límite 2 Z	
					0 =	no
					1 =	sí

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
30		<_DMODE>	INT	Modo de visualización		
				UNIDADES:	Plano de mecanizado G17/G18/G19	
					0 =	Compatibilidad, permanece activo el plano que estaba activo antes de la llamada de ciclo
					1 =	G17 (solo activo en el ciclo)
					2 =	G18 (solo activo en el ciclo)
					3 =	G19 (solo activo en el ciclo)
				DECENAS:	modo tecnológico	
					1 =	Mecanizado de contorno
					2 =	Ranurado de contorno
					3 =	Ranurado derecha/izquierda
				CENTENAS:	mecanizar material sobrante	
					0 =	no
					1 =	sí
				MILLARES:	---	reservado
				DECENAS DE MILLAR:	Escalado tecnológico dentro de máscaras de ciclo (Página 1192)	
					0 =	Entrada: completa
					1 =	Entrada: simple
				CENTENAS DE MILLAR:	Nombre de programa automático	
					0 =	no
					1 =	sí

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
31		<_AMODE>	INT	Modo alternativo		
				UNIDADES:	selección de la penetración	
					0 =	Penetración DX y DZ con el tipo de desbaste paralelo al contorno
					1 =	Penetración D
				DECENAS:	estrategia de penetración	
					0 =	Profundidad de corte variable (90 ... 100%)
					1 =	Profundidad de corte constante
				CENTENAS:	Distribución del corte	
					0 =	Uniforme
					1 =	Alinear en bordes
				MILLARES:	selección de las creces de contorno U1, acabado doble	
					0 =	no
					1 =	sí
				DECENAS DE MILLAR:	actualizar la selección de la pieza en bruto	
					0 =	no
					1 =	sí
				CENTENAS DE MILLAR:	selección de las creces en la pieza en bruto XD	
					0 =	Absoluto, valor del eje de refrentado en el diámetro
					1 =	Incremental, valor del eje de refrentado en el radio
				UNIDADES DE MILLÓN:	selección de las creces en la pieza en bruto ZD	
					0 =	Absoluto
					1 =	Incremental
				DECENAS DE MILLÓN:	selección del límite 2 XB	
					0 =	Absoluto, valor del eje de refrentado en el diámetro
					1 =	Incremental, valor del eje de refrentado en el radio
				CENTENAS DE MILLÓN:	selección del límite 2 ZB	
					0 =	Absoluto
					1 =	Incremental
				MILLAR DE MILLÓN:		
					0 =	Canal maestro
					1 =	Canal esclavo
32		<_PK>	INT	Número del canal asociado cuando hay más de 2 canales en la máquina.		

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción
33	DCH	<_DCH>	REAL	Decalaje de canal
34	FS	<_FS>	REAL	Avance de acabado en mecanizado completo

3.25.1.45 CYCLE4071 - Rectificado longitudinal con penetración en el punto de inversión

Sintaxis

```
CYCLE4071(<S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>, <S_H>, <S_A1>,
<S_A2>)
```

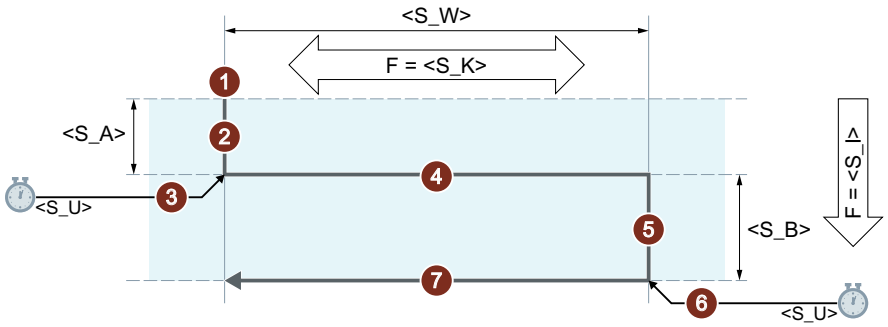
Parámetros

N.°	Parámetro	Tipo de datos	Descripción
1	<S_A>	REAL	Profundidad de penetración al principio
2	<S_B>	REAL	Profundidad de penetración al final
3	<S_W>	REAL	Anchura de rectificado
4	<S_U>	REAL	Tiempo de afinado
5	<S_I>	REAL	Avance para penetración
6	<S_K>	REAL	Avance para penetración transversal
7	<S_H>	INT	Número de repeticiones
8	<S_A1>	AXIS	Eje de penetración (opcional) o 1.er eje geométrico
9	<S_A2>	AXIS	Eje de vaivén (opcional) o 2.º eje geométrico

Función

El ciclo sirve para ejecutar penetraciones recurrentes. La profundidad de penetración puede ser distinta al principio y al final. Entre las penetraciones tiene lugar un movimiento tangencial.

Ejecución



① Inicio del ciclo en la posición actual del eje de vaivén.

② Desplazamiento del eje de penetración a la profundidad de penetración al principio <S_A> con avance para penetración <S_I>.

③ Afinar con el tiempo de afinado <S_U>.

④ Desplazamiento del eje de vaivén con la anchura de rectificado <S_W> como carrera de desplazamiento y el avance para penetración transversal <S_K>.

⑤ Desplazamiento del eje de penetración a la profundidad de penetración al final <S_B> con avance para penetración <S_I>.

⑥ Afinar con el tiempo de afinado <S_U>.

⑦ Desplazamiento del eje de vaivén con la anchura de rectificado <S_W> como carrera de desplazamiento al punto inicial y el avance para penetración transversal <S_K>.

Identifica los pasos secuenciales repetitivos.
La ejecución se repite hasta alcanzar el número programado de repeticiones <S_H>.

Nota

La ejecución no se puede interrumpir con una secuencia individual.

Ejemplo

Ejecutar dos movimientos de vaivén con los parámetros de ciclo siguientes:

- Profundidad de penetración al principio: 0,02 mm
- Profundidad de penetración al final: 0,01 mm
- Carrera: 100 mm
- Tiempo de afinado: 1 s
- Avance de penetración: 1 mm/min
- Avance transversal: 1000 mm/min
- Repeticiones: 2
- Eje de vaivén y eje de penetración: Ejes geométricos estándar

Código de programa

```
N10 T1 D1
```

Código de programa

```
N20 CYCLE4071(0.02,0.01,100,1,1,1000,2)
N30 M30
```

3.25.1.46 CYCLE4072 - Rectificado longitudinal con penetración en el punto de inversión y señal de cancelación

Sintaxis

```
CYCLE4072(<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>,
<S_H>, <S_A1>, <S_A2>)
```

Parámetros

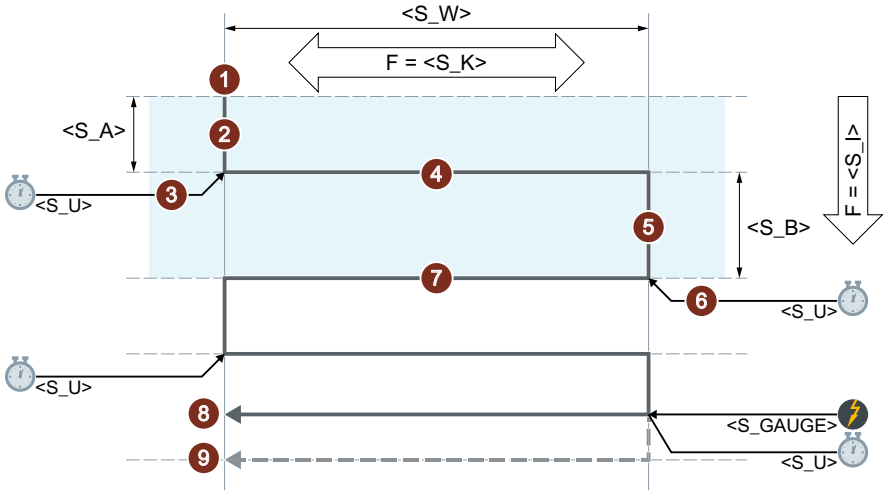
N.°	Parámetro	Tipo de datos	Descripción
1	<S_GAUGE>	STRING	Condiciones de cancelación para la penetración: 1. Número de una entrada rápida 2. Expresión lógica
2	<S_A>	REAL	Profundidad de penetración al principio
3	<S_B>	REAL	Profundidad de penetración al final
4	<S_W>	REAL	Anchura de rectificado
5	<S_U>	REAL	Tiempo de afinado
6	<S_I>	REAL	Avance para penetración
7	<S_K>	REAL	Avance para penetración transversal
8	<S_H>	INT	Número de repeticiones
9	<S_A1>	AXIS	Eje de penetración (opcional) o 1.er eje geométrico
10	<S_A2>	AXIS	Eje de vaivén (opcional) o 2.º eje geométrico

Función

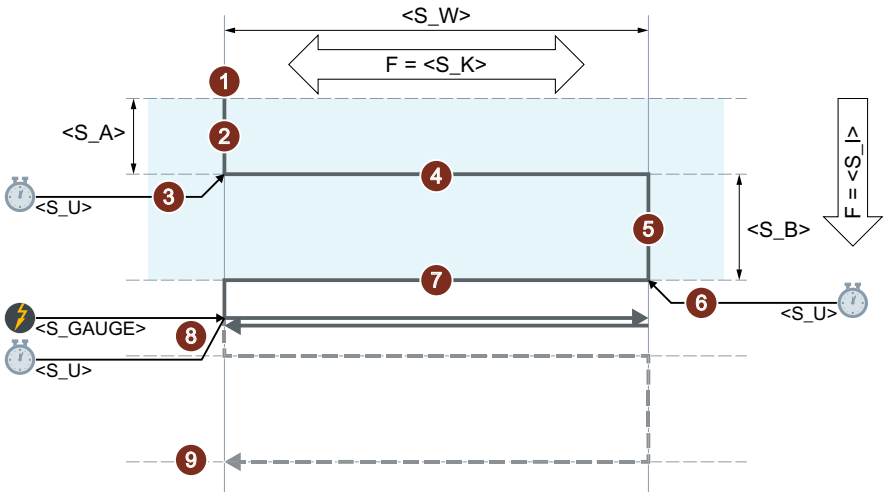
El ciclo sirve para ejecutar penetraciones recurrentes teniendo en cuenta una señal de cancelación externa. La profundidad de penetración puede ser distinta al principio y al final. Entre las penetraciones tiene lugar un movimiento tangencial. La penetración en profundidad se cancela cuando se cumple la condición de cancelación. Tras cancelarse la penetración en profundidad se ejecuta siempre una pasada completa.

Ejecución

Cancelación de la penetración al final



Cancelación de la penetración al principio



① Inicio del ciclo en la posición actual del eje de vaivén.

② Desplazamiento del eje de penetración a la profundidad de penetración al principio <S_A> con avance para penetración <S_I>.

③ Afinar con el tiempo de afinado <S_U>.

④ Desplazamiento del eje de vaivén con la anchura de rectificado <S_W> como carrera de desplazamiento y el avance para penetración transversal <S_K>.

⑤ Desplazamiento del eje de penetración a la profundidad de penetración al final <S_B> con avance para penetración <S_I>.

⑥ Afinar con el tiempo de afinado <S_U>.

⑦ Desplazamiento del eje de vaivén con la anchura de rectificado **<**S_W**>** como carrera de desplazamiento al punto inicial y el avance para penetración transversal <S_K>.

⑧ Señal de cancelación: El mecanizado finaliza al alcanzarse el siguiente punto inicial.

⑨ Sin señal de cancelación: La ejecución se repite hasta alcanzar el número programado de repeticiones <S_H>.

Identifica los pasos secuenciales repetitivos.

Nota

La ejecución no se puede interrumpir con una secuencia individual.

Recursos

El ciclo utiliza como recursos una acción síncrona modal y una variable de acción síncrona. La acción síncrona se determina dinámicamente a partir del rango libre de la banda de la acción síncrona (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). Como variable de acción síncrona se utiliza SYG_IS[1].

Ejemplos

Ejemplo 1: Oscilación con dos carreras

Parámetros de ciclo:

- Profundidad de penetración al principio: 0,02 mm
- Profundidad de penetración al final: 0,01 mm
- Carrera: 100 mm
- Tiempo de afinado: 1 s
- Avance de penetración: 1 mm/min
- Avance transversal: 1000 mm/min
- Repeticiones: 2
- Eje de vaivén y eje de penetración: Ejes geométricos estándar

Señal de cancelación: entrada rápida 1 ($A_IN[1])

Código de programa

```
N10 T1 D1
N20 CYCLE4072("1",0.02,0.01,100,1,1,1000,2)
N30 M30
```

Ejemplo 2: Oscilación con dos carreras

Parámetros de ciclo:

- Profundidad de penetración al principio: 0,02 mm
- Profundidad de penetración al final: 0,01 mm
- Carrera: 100 mm
- Tiempo de afinado: 1 s
- Avance de penetración: 1 mm/min
- Avance transversal: 1000 mm/min
- Repeticiones: 2
- Eje de vaivén y eje de penetración: Ejes geométricos estándar

Señal de cancelación: Variable $A_DBR[20] < 0,01

Código de programa

```
N10 T1 D1
N20 CYCLE4072("($A_DBR[20]<0.01)",0.02,0.01,100,1,1,1000,2)
N30 M30
```

3.25.1.47 CYCLE4073 - Rectificado longitudinal con penetración continua

Sintaxis

```
CYCLE4073(<S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>, <S_A1>, <S_A2>)
```

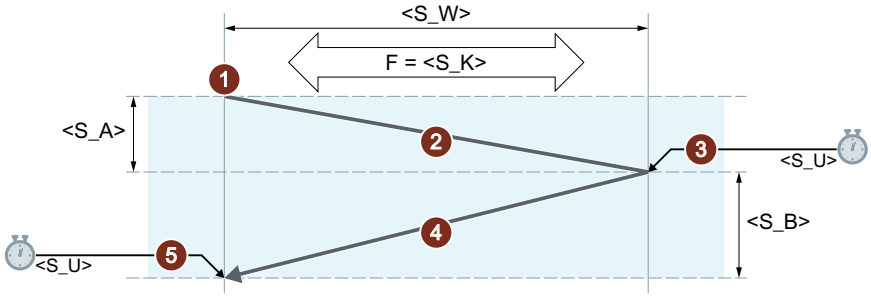
Parámetros

N.°	Parámetro	Tipo de datos	Descripción
1	<S_A>	REAL	Profundidad de penetración al principio
2	<S_B>	REAL	Profundidad de penetración al final
3	<S_W>	REAL	Anchura de rectificado
4	<S_U>	REAL	Tiempo de afinado
5	<S_K>	REAL	Avance para penetración transversal
6	<S_H>	INT	Número de repeticiones
7	<S_A1>	AXIS	Eje de penetración (opcional) o 1.er eje geométrico
8	<S_A2>	AXIS	Eje de vaivén (opcional) o 2.º eje geométrico

Función

El ciclo sirve para ejecutar penetraciones recurrentes. La penetración desde el principio hasta el final puede ser distinta a la penetración desde el final hasta el principio.

Ejecución



① Inicio del ciclo en la posición actual del eje de vaivén con profundidad de penetración 0.

② Desplazamiento del eje de vaivén con la anchura de rectificado <S_W> como carrera de desplazamiento y avance para penetración transversal <S_K> con aumento continuo de la profundidad de penetración hasta la penetración al principio <S_A>.

③ Afinar con el tiempo de afinado <S_U>.

④ Desplazamiento del eje de vaivén con la anchura de rectificado <S_W> como carrera de desplazamiento al punto inicial y avance para penetración transversal <S_K> con aumento continuo de la profundidad de penetración hasta la profundidad de penetración al final <S_B>.

⑤ Afinar con el tiempo de afinado <S_U>.

Identifica los pasos secuenciales repetitivos.

La ejecución se repite hasta alcanzar el número programado de repeticiones <S_H>.

Nota

La ejecución no se puede interrumpir con una secuencia individual.

Ejemplo

Oscilación con dos carreras

Parámetros de ciclo:

- Profundidad de penetración al principio: 0,02 mm
- Profundidad de penetración al final: 0,01 mm
- Carrera: 100 mm
- Tiempo de afinado: 1 s
- Avance transversal: 1000 mm/min
- Repeticiones: 2
- Eje de vaivén y eje de penetración: Ejes geométricos estándar

Código de programa

```
N10 T1 D1
N20 CYCLE4073(0.02,0.01,100,1,1000,2)
N30 M30
```

3.25.1.48 CYCLE4074 - Rectificado longitudinal con penetración continua y señal de cancelación

Sintaxis

```
CYCLE4074(<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>,
<S_A1>, <S_A2>)
```

Parámetros

N.°	Parámetro	Tipo de datos	Descripción
1	<S_GAUGE>	STRING	Condiciones de cancelación para la penetración: 1. Número de una entrada rápida 2. Expresión lógica
2	<S_A>	REAL	Profundidad de penetración al principio
3	<S_B>	REAL	Profundidad de penetración al final
4	<S_W>	REAL	Anchura de rectificado
5	<S_U>	REAL	Tiempo de afinado
6	<S_K>	REAL	Avance para penetración transversal
7	<S_H>	INT	Número de repeticiones

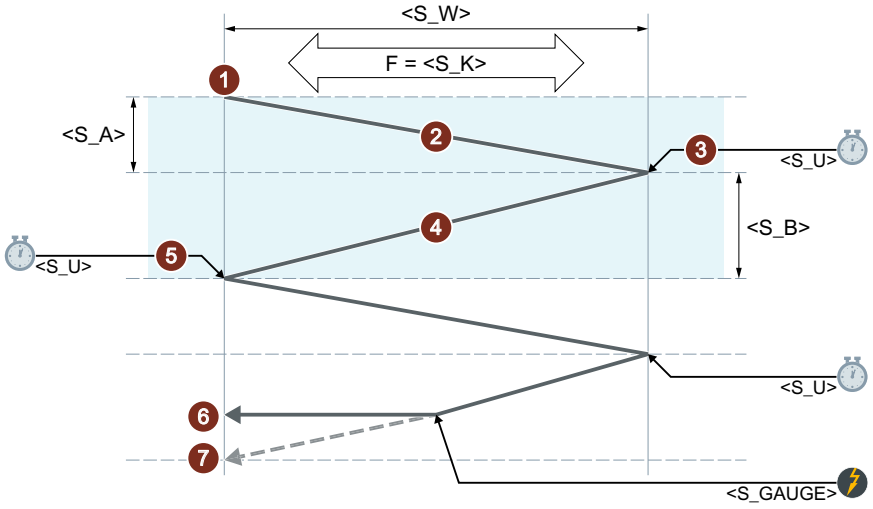
N.°	Parámetro	Tipo de datos	Descripción
8	<S_A1>	AXIS	Eje de penetración (opcional) o 1.er eje geométrico
9	<S_A2>	AXIS	Eje de vaivén (opcional) o 2.º eje geométrico

Función

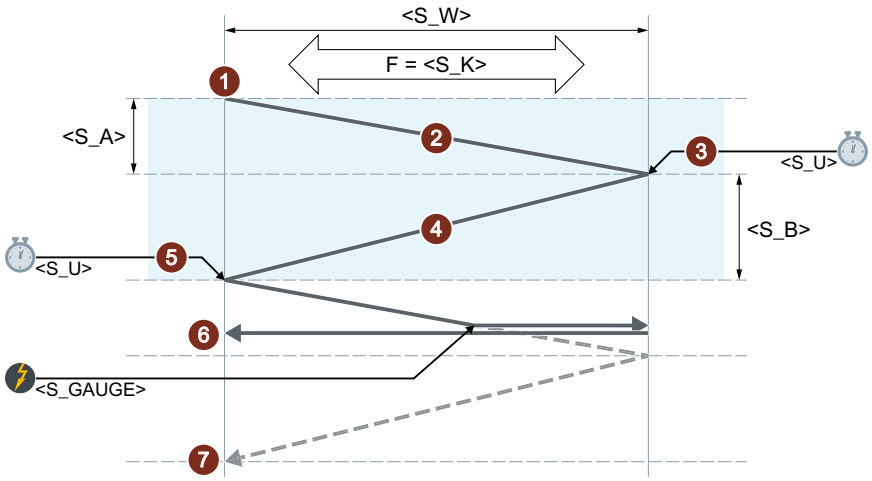
El ciclo sirve para ejecutar penetraciones recurrentes teniendo en cuenta, p. ej., una señal de cancelación externa. La profundidad de penetración puede ser distinta al principio y al final. La penetración en profundidad se cancela cuando se cumple la condición de cancelación. Tras cancelarse la penetración en profundidad se ejecuta siempre una pasada completa.

Ejecución

Cancelación de la penetración desde el final hasta el principio



Cancelación de la penetración desde el principio hasta el final



① Inicio del ciclo en la posición actual del eje de vaivén con profundidad de penetración 0.

② Desplazamiento del eje de vaivén con la anchura de rectificado <S_W> como carrera de desplazamiento y avance para penetración transversal <S_K> con aumento continuo de la profundidad de penetración hasta la penetración al principio <S_A>.

③ Afinar con el tiempo de afinado <S_U>.

④ Desplazamiento del eje de vaivén con la anchura de rectificado <S_W> como carrera de desplazamiento al punto inicial y avance para penetración transversal <S_K> con aumento continuo de la profundidad de penetración hasta la penetración al final <S_B>.

⑤ Afinar con el tiempo de afinado <S_U>.

⑥ Señal de cancelación: Se cancela la penetración en profundidad. El mecanizado finaliza al alcanzarse el siguiente punto inicial.

⑦ Sin señal de cancelación: La ejecución se repite hasta alcanzar el número programado de repeticiones <S_H>.

Identifica los pasos secuenciales repetitivos.

Nota

La ejecución no se puede interrumpir con una secuencia individual.

Recursos

El ciclo utiliza como recursos una acción síncrona modal y una variable de acción síncrona. La acción síncrona se determina dinámicamente a partir del rango libre de la banda de la acción síncrona (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). Como variable de acción síncrona se utiliza SYG_IS[1].

Ejemplos

Ejemplo 1: Oscilación con dos carreras

Parámetros de ciclo:

- Profundidad de penetración al principio: 0,02 mm
- Profundidad de penetración al final: 0,01 mm
- Carrera: 100 mm
- Tiempo de afinado: 1 s
- Avance transversal: 1000 mm/min
- Repeticiones: 2
- Eje de vaivén y eje de penetración: Ejes geométricos estándar

Señal de cancelación: entrada rápida 1 ($A_IN[1])

Código de programa

```
N10 T1 D1
N20 CYCLE4074("1",0.02,0.01,100,1,1000,2)
N30 M30
```

Ejemplo 2: Oscilación con dos carreras

Parámetros de ciclo:

- Profundidad de penetración al principio: 0,02 mm
- Profundidad de penetración al final: 0,01 mm
- Carrera: 100 mm
- Tiempo de afinado: 1 s
- Avance transversal: 1000 mm/min
- Repeticiones: 2
- Eje de vaivén y eje de penetración: Ejes geométricos estándar

Señal de cancelación: Variable $A_DBR[20] < 0,01

Código de programa

```
N10 T1 D1
N20 CYCLE4074("($A_DBR[20]<0.01)",0.02,0.01,100,1,1000,2)
N30 M30
```

3.25.1.49 CYCLE4075 - Rectificado plano con penetración en el punto de inversión

Sintaxis

```
CYCLE4075(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>,
<S_A2>)
```

Parámetros

N.°	Parámetro	Tipo de datos	Descripción
1	<S_I>	REAL	Profundidad de penetración al principio
2	<S_J>	REAL	Profundidad de penetración al final
3	<S_K>	REAL	Profundidad total de penetración
4	<S_A>	REAL	Anchura de rectificado
5	<S_R>	REAL	Avance para penetración
6	<S_F>	REAL	Avance para penetración transversal
7	<S_P>	REAL	Tiempo de afinado
8	<S_A1>	AXIS	Eje de penetración (opcional)
9	<S_A2>	AXIS	Eje de vaivén (opcional)

Función

El ciclo sirve para mecanizar por pasos de penetración con una profundidad total de penetración. Las profundidades de penetración pueden ser distintas al principio y al final. Entre las penetraciones tiene lugar un movimiento tangencial.

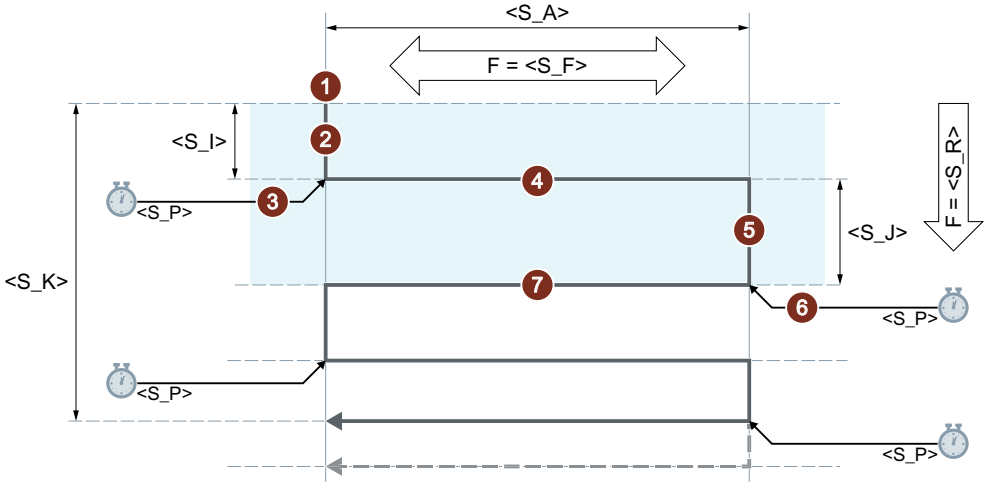
La información de recorrido P1 a P4 puede ser negativa o positiva.

La indicación del eje de penetración o del eje de vaivén es opcional. Si no se indica uno de los dos parámetros, o ninguno de ellos, el ciclo utiliza los dos primeros ejes geométricos del canal.

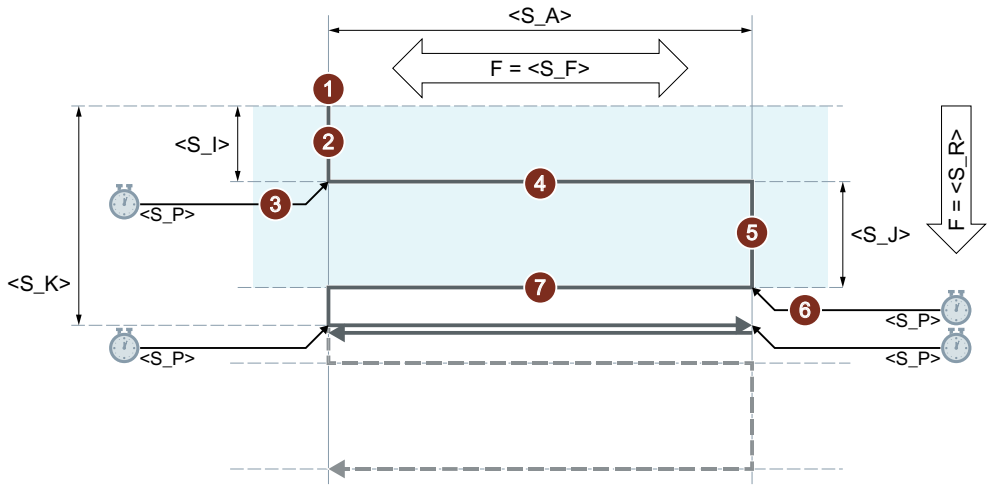
Si la suma de las profundidades de penetración al principio y al final es igual a 0, o la profundidad total de penetración es igual a 0, se realiza una sola carrera de afinado.

Ejecución

Profundidad de penetración total alcanzada al penetrar en el segundo punto de inversión



Profundidad de penetración total alcanzada al penetrar en el primer punto de inversión



① Inicio del ciclo en la posición actual del eje de vaivén.

② Desplazamiento del eje de penetración a la profundidad de penetración al principio <S_I> con avance para penetración <S_R>.

③ Afinar con el tiempo de afinado <S_P>.

④ Desplazamiento del eje de vaivén con la anchura de rectificado <S_A> como carrera de desplazamiento y el avance para penetración transversal <S_F>.

⑤ Desplazamiento del eje de penetración a la profundidad de penetración al final <S_J> con avance para penetración <S_R>.

⑥ Afinar con el tiempo de afinado <S_P>.

⑦ Desplazamiento del eje de vaivén con la anchura de rectificado <S_A> como carrera de desplazamiento al punto inicial y el avance para penetración transversal <S_F>.

Identifica los pasos secuenciales repetitivos.

La secuencia se repite hasta alcanzar la profundidad total de penetración <S_K>. La última pasada se divide entonces de forma inhomogénea.

Nota

La ejecución no se puede interrumpir con una secuencia individual.

Ejemplo

Oscilación con:

- Profundidad de penetración de 0,02 mm al principio
- Profundidad de penetración de 0,01 mm al final
- Profundidad total de penetración 1 mm
- Carrera 100 mm
- Avance de penetración 1 mm/min
- Avance transversal 1000 mm/min

- Tiempo de afinado 1 segundo
- Ejes geométricos estándar

Código de programa

```
N10 T1 D1
N20 CYCLE4075(0.02,0.01,1,100,1,1000,1)
N30 M30
```

3.25.1.50 CYCLE4077 - Rectificado plano con penetración en el punto de inversión y señal de cancelación

Sintaxis

```
CYCLE4077(<S_GAUGE>, <S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>,
<S_P>, <S_A1>, <S_A2>)
```

Parámetros

N.°	Parámetro	Tipo de datos	Descripción
1	<S_GAUGE>	STRING	Condición de cancelación de la penetración: • Número de una entrada rápida • Expresión lógica
2	<S_I>	REAL	Profundidad de penetración al principio
3	<S_J>	REAL	Profundidad de penetración al final
4	<S_K>	REAL	Profundidad total de penetración
5	<S_A>	REAL	Anchura de rectificado
6	<S_R>	REAL	Avance para penetración
7	<S_F>	REAL	Avance para penetración transversal
8	<S_P>	REAL	Tiempo de afinado
9	<S_A1>	AXIS	Eje de penetración (opcional)
10	<S_A2>	AXIS	Eje de vaivén (opcional)

Función

El ciclo sirve para mecanizar por pasos de penetración con una profundidad total de penetración. Las profundidades de penetración pueden ser distintas al principio y al final. Entre las penetraciones tiene lugar un movimiento tangencial. La penetración en profundidad se cancela cuando la señal de cancelación de la entrada rápida es igual a 1 o se cumple la condición de cancelación. Tras la cancelación se ejecuta una carrera completa.

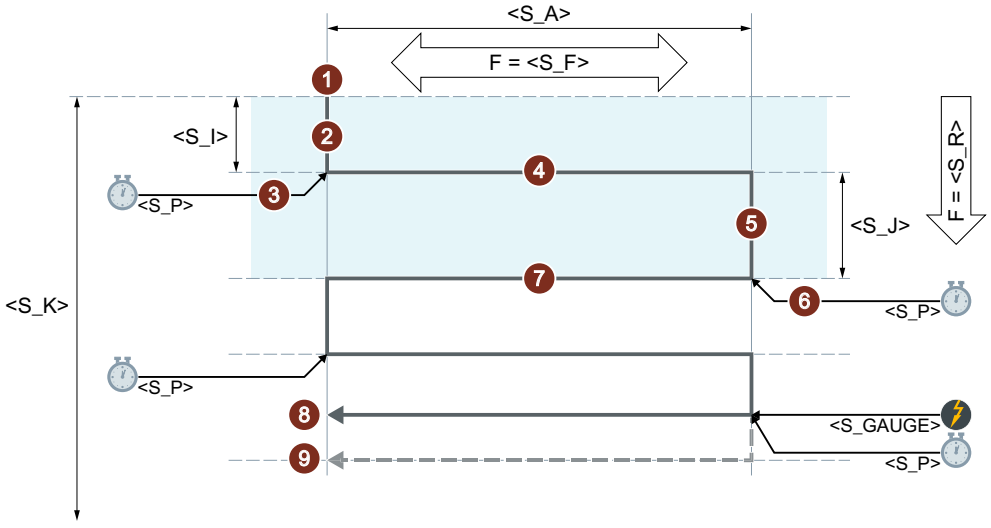
La información de recorrido P2 a P5 puede ser negativa o positiva.

La indicación del eje de penetración o del eje de vaivén es opcional. Si no se indica uno de los dos parámetros, o ninguno de ellos, el ciclo utiliza los dos primeros ejes geométricos del canal.

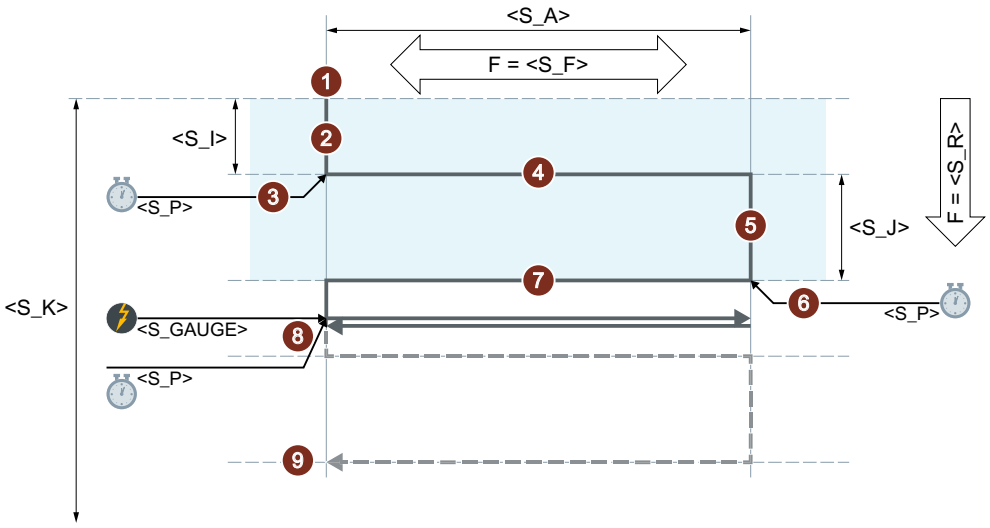
Si la suma de las profundidades de penetración al principio y al final es igual a 0, o la profundidad total de penetración es igual a 0, se realiza una sola carrera de afinado.

Ejecución

Cancelación de la penetración al final



Cancelación de la penetración al principio



① Inicio del ciclo en la posición actual del eje de vaivén.

② Desplazamiento del eje de penetración a la profundidad de penetración al principio <S_I> con avance para penetración <S_R>.

③ Afinar con el tiempo de afinado <S_P>.

④ Desplazamiento del eje de vaivén con la anchura de rectificado <S_A> como carrera de desplazamiento y el avance para penetración transversal <S_F>.

⑤ Desplazamiento del eje de penetración a la profundidad de penetración al final <S_J> con avance para penetración <S_R>.

⑥ Afinar con el tiempo de afinado <S_P>.

⑦ Desplazamiento del eje de vaivén con la anchura de rectificado <S_A> como carrera de desplazamiento al punto inicial y el avance para penetración transversal <S_F>.

⑧ Señal de cancelación: El mecanizado finaliza al alcanzarse el siguiente punto inicial.

⑨ Sin señal de cancelación: La secuencia se repite hasta alcanzar la profundidad total de penetración <S_K>. La última pasada se divide entonces de forma inhomogénea.

Identifica los pasos secuenciales repetitivos.

Nota

La ejecución no se puede interrumpir con una secuencia individual.

Recursos

El ciclo utiliza como recursos una acción síncrona modal y una variable de acción síncrona. La acción síncrona se determina dinámicamente a partir del rango libre de la banda de la acción síncrona (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR – 1199 ...). Como variable de acción síncrona se utiliza SYG_IS[1].

Ejemplos

Ejemplo 1

Oscilación con:

- Profundidad de penetración de 0,02 mm al principio
- Profundidad de penetración de 0,01 mm al final
- Profundidad total de penetración 1 mm
- Carrera 100 mm
- Avance de penetración 1 mm/min
- Avance transversal 1000 mm/min
- Tiempo de afinado 1 segundo
- Ejes geométricos estándar

Señal de cancelación: entrada rápida 1 ($A_IN[1])

Código de programa

```
N10 T1 D1
N20 CYCLE4077("1",0.02,0.01,1,100,1,1000,1)
N30 M30
```

Ejemplo 2

Oscilación con:

- Profundidad de penetración de 0,02 mm al principio
- Profundidad de penetración de 0,01 mm al final
- Profundidad total de penetración 1 mm
- Carrera 100 mm
- Avance de penetración 1 mm/min
- Avance transversal 1000 mm/min
- Tiempo de afinado 1 segundo
- Ejes geométricos estándar

Señal de cancelación: variable Dual Port RAM 20 inferior a 0,01 ($A_DBR[20] < 0,01)

Código de programa

```
N10 T1 D1
N20 CYCLE4077("($A_DBR[20]<0.01)",0.02,0.01,1,100,1,1000,1)
N30 M30
```

3.25.1.51 CYCLE4078 - Rectificado plano con penetración continua

Sintaxis

```
CYCLE4078(<S_I>, <S_J>, <S_K>, <S_A>, <S_F>, <S_P>, <S_A1>, <S_A2>)
```

Parámetros

N.°	Parámetro	Tipo de datos	Descripción
1	<S_I>	REAL	Profundidad de penetración desde el principio hasta el final
2	<S_J>	REAL	Profundidad de penetración desde el final hasta el principio
3	<S_K>	REAL	Profundidad total de penetración
4	<S_A>	REAL	Anchura de rectificado
5	<S_F>	REAL	Avance
6	<S_P>	REAL	Tiempo de afinado
7	<S_A1>	AXIS	Eje de penetración (opcional)
8	<S_A2>	AXIS	Eje de vaivén (opcional)

Función

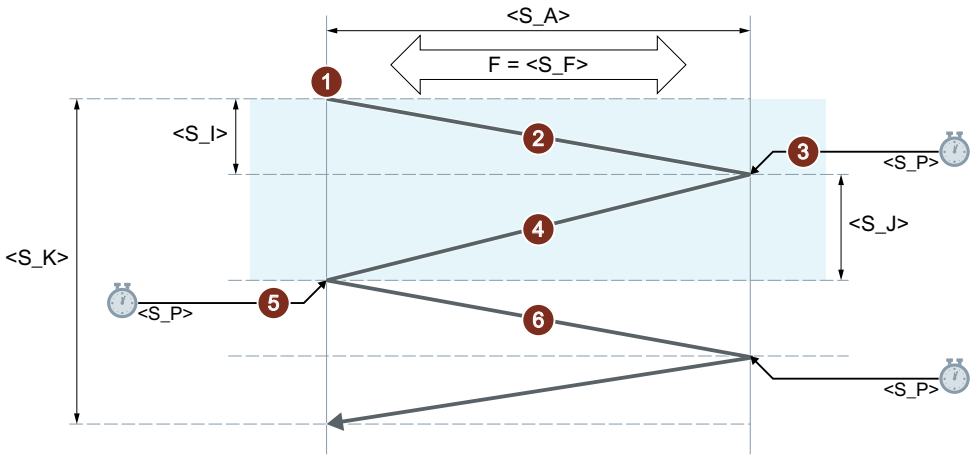
El ciclo sirve para mecanizar por penetración continua con una profundidad total de penetración. Las profundidades de penetración del principio al final y del final al principio pueden ser distintas.

La información de recorrido P1 a P4 puede ser negativa o positiva.

La indicación del eje de penetración o del eje de vaivén es opcional. Si no se indica uno de los dos parámetros, o ninguno de ellos, el ciclo utiliza los dos primeros ejes geométricos del canal.

Si la suma de las profundidades de penetración P1 y P2 es igual a 0 o bien la profundidad total de penetración es igual a 0, se realiza una sola carrera de afinado.

Ejecución



① Inicio del ciclo en la posición actual del eje de vaivén con profundidad de penetración 0.

② Desplazamiento del eje de vaivén con la anchura de rectificado <S_A> como carrera de desplazamiento y avance <S_F> con aumento continuo de la profundidad de penetración hasta la profundidad de penetración al principio **<S_I>**.

③ Afinar con el tiempo de afinado <S_P>.

④ Desplazamiento del eje de vaivén con la anchura de rectificado <S_A> como carrera de desplazamiento al punto inicial y avance <S_F> con aumento continuo de la profundidad de penetración hasta la profundidad de penetración al final <S_J>.

⑤ Afinar con el tiempo de afinado <S_P>.

⑥ Desplazamiento del eje de vaivén con la anchura de rectificado <S_A> como carrera de desplazamiento al punto inicial y avance <S_F>.

Identifica los pasos secuenciales repetitivos.

La secuencia se repite hasta alcanzar la profundidad total de penetración <S_K>. La última pasada se divide entonces de forma inhomogénea.

Nota

La ejecución no se puede interrumpir con una secuencia individual.

Ejemplo

Oscilación con:

- Profundidad de penetración de 20 mm al principio
- Profundidad de penetración de 10 mm al final
- Profundidad total de penetración 100 mm
- Carrera 100 mm
- Avance 1000 mm/min
- Tiempo de afinado 1 segundo
- Ejes geométricos estándar

Código de programa

```
N10 T1 D1
N20 CYCLE4078(20,10,100,100,1000,1)
N30 M30
```

3.25.1.52 CYCLE4079 - Rectificado plano con penetración intermitente

Sintaxis

```
CYCLE4079(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>,
<S_A2>)
```

Parámetros

N.°	Parámetro	Tipo de datos	Descripción
1	<S_I>	REAL	Profundidad de penetración al principio
2	<S_J>	REAL	Profundidad de penetración al final
3	<S_K>	REAL	Profundidad total de penetración
4	<S_A>	REAL	Anchura de rectificado
5	<S_R>	REAL	Avance para penetración
6	<S_F>	REAL	Avance para penetración transversal
7	<S_P>	REAL	Tiempo de afinado
8	<S_A1>	AXIS	Eje de penetración (opcional)
9	<S_A2>	AXIS	Eje de vaivén (opcional)

Función

El ciclo sirve para mecanizar por pasos de penetración con una profundidad total de penetración. Las profundidades de penetración pueden ser distintas al principio y al final. Entre las penetraciones tiene lugar un movimiento tangencial.

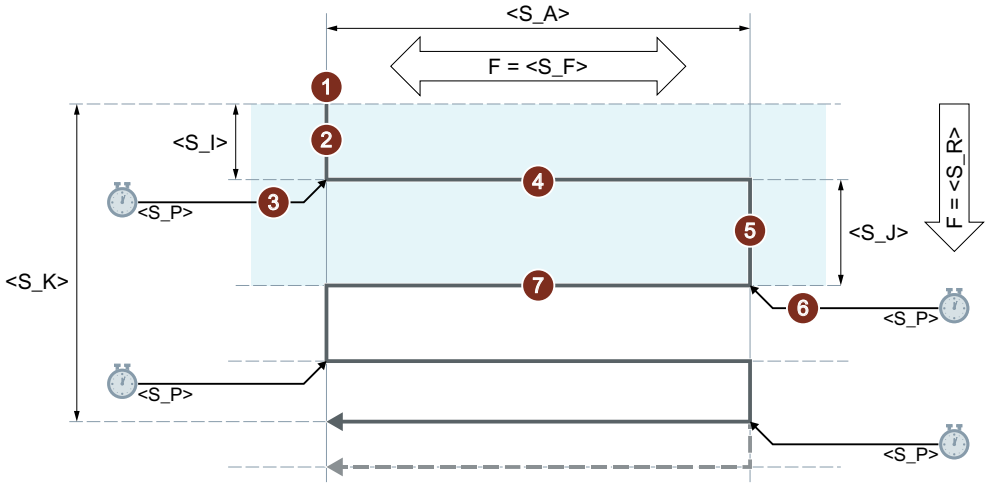
La información de recorrido P1 a P4 puede ser negativa o positiva.

La indicación del eje de penetración o del eje de vaivén es opcional. Si no se indica uno de los dos parámetros, o ninguno de ellos, el ciclo utiliza los dos primeros ejes geométricos del canal.

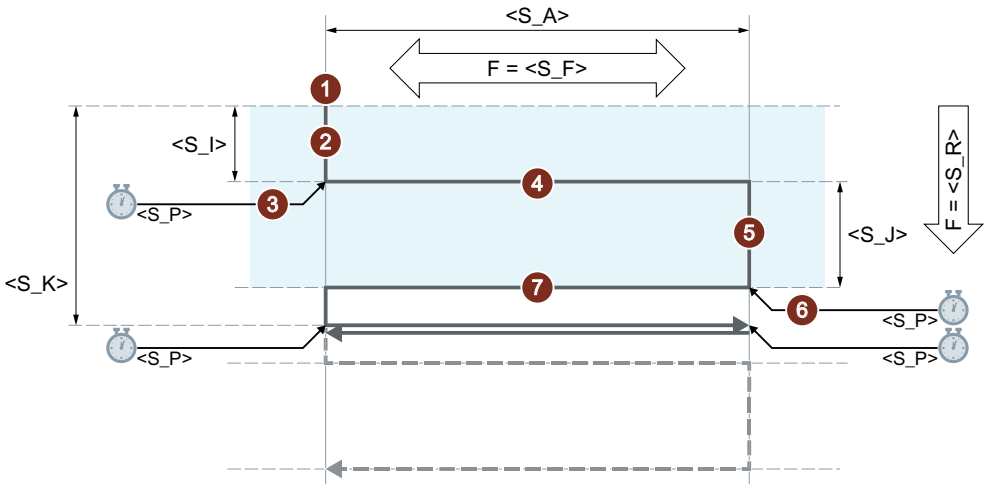
Si la suma de las profundidades de penetración al principio y al final es igual a 0, o la profundidad total de penetración es igual a 0, se realiza una sola carrera de afinado.

Ejecución

Profundidad de penetración total alcanzada al penetrar en el segundo punto de inversión



Profundidad de penetración total alcanzada al penetrar en el primer punto de inversión



① Inicio del ciclo en la posición actual del eje de vaivén.

② Desplazamiento del eje de penetración a la profundidad de penetración al principio <S_I> con avance para penetración <S_R>.

③ Afinar con el tiempo de afinado <S_P>.

④ Desplazamiento del eje de vaivén con la anchura de rectificado <S_A> como carrera de desplazamiento y el avance para penetración transversal <S_F>.

⑤ Desplazamiento del eje de penetración a la profundidad de penetración al final <S_J> con avance para penetración <S_R>.

⑥ Afinar con el tiempo de afinado <S_P>.

⑦ Desplazamiento del eje de vaivén con la anchura de rectificado <S_A> como carrera de desplazamiento al punto inicial y el avance para penetración transversal <S_F>.

Identifica los pasos secuenciales repetitivos.

La secuencia se repite hasta alcanzar la profundidad total de penetración <S_K>. La última pasada se divide entonces de forma inhomogénea.

Nota

La ejecución no se puede interrumpir con una secuencia individual.

Ejemplo

Oscilación con:

- Profundidad de penetración de 0,02 mm al principio
- Profundidad de penetración de 0,01 mm al final
- Profundidad total de penetración 1 mm
- Carrera 100 mm
- Avance de penetración 1 mm/min
- Avance transversal 1000 mm/min

- Tiempo de afinado 1 segundo
- Ejes geométricos estándar

```
Código de programa
N10 T1 D1
N20 CYCLE4079(0.02,0.01,1,100,1,1000,1)
N30 M30
```

3.25.1.53 GROUP_BEGIN - Inicio bloque de programa

Sintaxis

```
GROUP_BEGIN(<_LEVEL>, <_NAME>, <_SP>, <_MODE>, <S_ICON>)
```

Parámetro

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción		
1		<_LEVEL>	INT	Plano		
				0 =	Plano principal	
				1 =	1.er subplano	
2		<_NAME>	STRING[128]	Nombre de bloque		
3		<_SP>	INT	Cabezal		
				0 =	Sin cabezal	
				1 =	Cabezal principal	
				2 =	Contracabezal	
4		<_MODE>	INT	Modo		
				Bit 0	= 1	GROUP_ADDEND existe
				Bit 1	= 1	ShopTurn: retirada automática (desplazar al punto de cambio de la herramienta)
				Bit 12	Reservado	
				Bit 13	Reservado	
5		<S_ICON>	STRING[32]	Nombre del icono (solo para interfaz de usuario)		

3.25.1.54 GROUP_END - Fin bloque de programa

Sintaxis

```
GROUP_END(<_LEVEL>, <_SP>)
```

Parámetro

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción	
1		<_LEVEL>	INT	Plano	
				0 =	Plano principal
				1 =	1.er subplano
2		<_SP>	INT	Cabezal	
				0 =	Sin cabezal
				1 =	Cabezal principal
				2 =	Contracabezal

3.25.1.55 GROUP_ADDEND - Fin adición aprox.

Sintaxis

```
GROUP_ADDEND(<_LEVEL>, <_SP>)
```

Parámetro

N.°	Máscara de parámetros	Parámetro interno	Tipo de datos	Descripción	
1		<_LEVEL>	INT	Plano	
				0 =	Plano principal
				1 =	1.er subplano
2		<_SP>	INT	Cabezal	
				0 =	Sin cabezal
				1 =	Cabezal principal
				2 =	Contracabezal

3.25.1.56 Condiciones marginales

Escalado tecnológico dentro de máscaras de ciclo

Si el escalado tecnológico está activo, en diferentes máscaras de ciclo puede seleccionarse la entrada simplificada, con la que tan solo se muestran los parámetros de ciclo más importantes.

La entrada simplificada puede seleccionarse, p. ej., en las siguientes máscaras de ciclo:

Tecnología	Máscara de ciclo
Taladrado	Taladrado profundo
	Roscado
Fresado	Caja rectangular
	Fresado del contorno: caja

Tecnología	Máscara de ciclo
Torneado	Tallado de roscas: longitudinal
	Torneado de contorno: desbaste
	Torneado de contorno: ranurado
	Torneado de contorno: ranurado derecha/izquierda

En la interfaz de usuario de las máscaras de ciclo correspondientes se puede elegir entre "Entrada: **simple**" y "Entrada: **completa**".

Parámetros de ciclo no mostrados

Los parámetros de ciclo que no se muestran en la entrada simplificada se preasignan con valores fijos adecuados desde el punto de vista tecnológico, pero que no pueden modificarse. O bien se asignan valores parametrizables a los parámetros de ciclo a través de datos de operador de ciclos específicos de canal. Ver a continuación el apartado "Puesta en marcha" > "Datos de operador de ciclos específicos de canal"

Conmutación "Entrada: completa" > "Entrada: simple"

Si una máscara de ciclo se rellena con el ajuste "Entrada completa" y a continuación se cambia a "Entrada simple", al generar la llamada de ciclo para los parámetros que ya no se muestran se utilizan los valores predeterminados o los valores de datos de operador.

Puesta en marcha

Datos de máquina de configuración específicos de canal

Con el dato de máquina se activa la posibilidad de escalado tecnológico dentro de máscaras de ciclo:

MD52210 $MCS_FUNCTION_MASK_DISP, bit 9 = 1 (mostrar selección "Entrada simple")

Datos de operador de ciclos específicos de canal

Si está activada la entrada simplificada dentro de las máscaras de ciclo, los valores de determinados parámetros de ciclo pueden preasignarse a través de los siguientes datos de operador:

Número	Indicador	Descripción
SD55300	$SCS_EASY_SAFETY_CLEARANCE	Distancia de seguridad
SD55301	$SCS_EASY_DWELL_TIME	Tiempo de parada
SD55305	$SCS_EASY_DRILL_DEEP_FD1	Taladrado profundo: porcentaje: 1.er avance
SD55306	$SCS_EASY_DRILL_DEEP_DF	Taladrado profundo: porcentaje: penetración
SD55307	$SCS_EASY_DRILL_DEEP_V1	Taladrado profundo: penetración mínima en profundidad
SD55308	$SCS_EASY_DRILL_DEEP_V2	Taladrado profundo: valor de retirada
SD55309	$SCS_EASY_THREAD_RETURN_DIST	Tallado de roscas: distancia de retroceso

3.25.2 Ciclos de medida

Los ciclos de medida son subprogramas especiales proporcionados por Siemens para resolver determinadas tareas de medición. Como es habitual en los ciclos en general, los ciclos de medida pueden adaptarse al problema concreto mediante parametrización.

Los ciclos de medida están disponibles para mediciones en los siguientes campos y tecnologías:

- Mediciones de herramientas torneado/fresado
- Mediciones de piezas torneado/fresado

Información adicional

Manual de programación Ciclos de medida

3.25.3 Vista general de parámetros de ciclos de medida

3.25.3.1 Parámetros de ciclo de medida 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)
```

Tabla 3-8 Parámetros de llamada CYCLE973 [1)]

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
1		S_MVAR	Variante de medida (predeterminado = 0012103)	
			Valores:	UNIDADES: calibración en una superficie, un borde o una ranura 0 = longitud en superficie/borde (en WKS) con valor teórico conocido 1 = radio en superficie (en WKS) con valor teórico conocido 2 = longitud en ranura (en MKS), ver S_CALNUM 3 = radio en ranura (en MKS), ver S_CALNUM
				DECENAS: reservado 0 = 0
				CENTENAS: reservado 0 = 0
				MILLARES: selección de eje de medición y sentido de medición al calibrar [2)] 0 = sin indicación (sin selección de eje de medición y sentido de medición al calibrar en superficie del fondo de la ranura) [4)] 1 = indicar selección de eje de medición y sentido de medición, ver S_MA, S_MD (un sentido de medición en un eje de medición) 2 = indicar selección de eje de medición, ver S_MA (dos sentidos de medición en un eje de medición)
				DECENAS DE MILLAR: determinación de desviación de posición (inclinación del palpador) [2), 3)] 0 = determinar desviación de posición 1 = no determinar desviación de posición
				CENTENAS DE MILLAR: reservado 0 = 0
				UNIDADES DE MILLÓN: adaptar longitud de herramienta [7)] 0 = No adaptar longitud de herramienta (solo puntos de conmutación) 1 = Adaptar longitud de herramienta
2	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número del palpador) (predeterminado = 1)	
3		S_CALNUM	Número de la ranura de calibrado al calibrar en una ranura (predeterminado = 1) [5)]	
4		S_SETV	Valor teórico al calibrar en una superficie	
5	X0	S_MA	Eje de medición (número del eje) [6)] (predeterminado = 1)	
			Valores:	1 = 1.er eje del plano (con G18 Z) 2 = 2.º eje del plano (con G18 X) 3 = 3.er eje del plano (con G18 Y) [6)]
6	+-	S_MD	Sentido de medición (predeterminado = 1)	
			Valores:	0 = sentido de medición positivo 1 = sentido de medición negativo
7	DFA	S_FA	Trayecto de medición	
8	TSA	S_TSA	Zona de confianza	
9	VMS	S_VMS	Velocidad de medición variable al calibrar [2)]	
10	Mediciones	S_NMSP	Número de mediciones en la misma posición [2)] (predeterminado = 1)	

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
11		S_MCBIT	reservado	
12		_DMODE	Modo display	
			Valores:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
13		_AMODE	Modo alternativo	

1) Todos los valores predeterminados = 0 o están identificados con predeterminado = x

2) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Solo relevante al calibrar en dos sentidos de eje

4) Solo el eje de medición y el sentido de medición se determinan automáticamente a partir de la posición de filo (SL) del palpador. SL = 8 → -X , SL = 7 → -Z

5) El número de la ranura de calibrado (n) remite a los siguientes datos generales de operador (todas las posiciones en MKS):
Con posición de filo SL=7:
SD54615 $SNS_MEA_CAL_EDGE_BASE_AX1[n] Posición del fondo de la ranura en el 1.er eje del plano (con G18 Z)
SD54621 $SNS_MEA_CAL_EDGE_PLUS_DIR_AX2[n] Posición de la pared de la ranura en el sentido positivo del 2.º eje del plano (con G18 X)
SD54622 $SNS_MEA_CAL_EDGE_MINUS_DIR_AX2[n] Posición de la pared de la ranura en el sentido negativo del 2.º eje del plano
Con posición de filo SL=8:
SD54619 $SNS_MEA_CAL_EDGE_BASE_AX2[n] Posición del fondo de la ranura en el 2.º eje del plano
SD54620 $SNS_MEA_CAL_EDGE_UPPER_AX2[n] Posición del borde superior de la ranura en el 2.º eje del plano (solo para el preposicionamiento del palpador)
SD54617 $SNS_MEA_CAL_EDGE_PLUS_DIR_AX1[n] Posición de la pared de la ranura en el sentido positivo del 1.er eje del plano
SD54618 $SNS_MEA_CAL_EDGE_MINUS_DIR_AX1[n] Posición de la pared de la ranura en el sentido negativo del 1.er eje del plano
Nota:
Los valores de posición para la pared de la ranura +- pueden haberse determinado de forma aproximada.
El ancho de la ranura, que se obtiene a partir de la diferencia de los valores de posición de la pared de la ranura, debe haberse determinado con exactitud (reloj comparador).
Al calibrar en la ranura, se parte de la base de que la longitud de herramienta del palpador del eje calibrado = 0.
Los valores de posición para el fondo de la ranura también deben determinarse de forma exacta en la máquina (sin cotas de plano).

6) Eje de medición S_MA=3 al calibrar en una superficie y en un torno con el 3.er eje real del plano (con G18 Y).

7) Adaptar longitud de herramienta con calibración de longitud en ranura o con longitudes en superficie.
El palpador de pieza en tornos puede describirse con 2 longitudes (X, Z).
Palpador torneado tipo 580
Posición de filo 7: con la calibración de longitud se corrige opcionalmente la longitud Z.
Palpador torneado tipo 580
Posición de filo 8: con la calibración de longitud se corrige opcionalmente la longitud X
Con las variantes de medida de radio en ranura o radio en superficie, no se efectúa ninguna adaptación de la longitud de herramienta.
Solo se guardan siempre los puntos de conmutación correspondientes.

3.25.3.2 Parámetros de ciclo de medida 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)
```

Tabla 3-9 Parámetros de llamada CYCLE974 [1)]

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
1		S_MVAR	Variante de medida	
			Valores:	UNIDADES: 0 = medición de superficie frontal 1 = medición interior 2 = medición exterior
				DECENAS: reservado
				CENTENAS: objetivo de corrección 0 = solo medición (sin corrección de DO ni corrección de herramienta) 1 = medición, determinación y corrección de DO (ver S_KNUM)[3)] 2 = medición y corrección de herramienta (ver S_KNUM1)
				MILLARES: reservado
				DECENAS DE MILLAR: medición con o sin giro del cabezal principal (husillo de cabezal) 0 = medición sin giro 1 = medición con giro
2	Selección	S_KNUM	Corrección en decalaje de origen (DO), DO base o referencia base[2)]	
			Valores:	UNIDADES:
				DECENAS: 0 = sin corrección 1 a máx. 99 números del decalaje de origen o 1 a máx. 16 números del decalaje base
				CENTENAS: reservado
				MILLARES: Corrección en DO, DO base o referencia base 0 = corrección DO ajustable 1 = corrección DO base específico de canal 2 = corrección referencia base 3 = corrección DO base global 9 = corrección DO activo o, con G500, último DO base específico de canal activo
				DECENAS DE MILLAR: Corrección gruesa o fina en DO, DO base o referencia base 0 = corrección fina [6)] 1 = corrección gruesa

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
3	Selección	S_KNUM1	Corrección en corrección de herramienta [2), 4)]	
			Valores:	UNIDADES:
				DECENAS:
				CENTENAS: 0 = sin corrección 1 a máx. 999 números D (número de filo) para corrección de herramienta; para corrección aditiva y de preparación, ver también S_DLNUM
				MILLARES: 0 o bien números D unívocos
				DECENAS DE MILLAR: 0 o bien números D unívocos de 1 a máx. 32.000 si hay configurados números D unívocos en DM
				CENTENAS DE MILLAR: corrección de herramienta [2)] 0 = sin indicación (corrección en geometría de herramienta) 1 = corrección de longitud L1 2 = corrección de longitud L2 3 = corrección de longitud L3 4 = corrección radio
				UNIDADES DE MILLÓN: corrección de herramienta [2)] 0 = sin indicación (corrección del desgaste de la longitud de herramienta) 1 = corrección de herramienta, corrección aditiva (SK) [5)] El valor de corrección de herramienta se suma a la SK existente 2 = corrección de herramienta, corrección de preparación (EK) [5)] EK (nueva) = EK (antigua) + valor de corrección SK (antigua), SK (nueva) = 0 3 = corrección de herramienta, corrección de preparación (EK) [5)] El valor de corrección de herramienta se suma a la EK existente 4 = corrección de herramienta, geometría
				DECENAS DE MILLÓN: corrección de herramienta [2)] 0 = sin indicación (corrección en geometría de herramienta normal, no invertida) 1 = corrección invertida
				CENTENAS DE MILLÓN: corrección de herramienta 0 = corrección de herramienta sin herramienta Duplo 1 = corrección de herramienta en la herramienta Duplo (_DP)
4	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número del palpador) (predeterminado = 1)	
5	X0	S_SETV	Valor teórico	
6	X	S_MA	Eje de medición (número del eje) (predeterminado = 1)	
			Valores:	1 = 1.er eje del plano (con G18 Z) 2 = 2.º eje del plano (con G18 X) 3 = 3.er eje del plano (con G18 Y)[5)]
7	DFA	S_FA	Trayecto de medición	
8	TSA	S_TSA	Zona de confianza	
9	α	S_STA1	Ángulo inicial al medir con giro	
10	Mediciones	S_NMSP	Número de mediciones en la misma posición [2)] (predeterminado = 1)	
11	T	S_TNAME	Nombre de herramienta [2)]	
12	DL	S_DLNUM	Número DL, corrección aditiva de preparación [5)]	
13	ST	_DP	Número de la herramienta de sustitución a corregir (número Duplo)	

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
14	TZL	S_TZL	Corrección cero [2), 4)]	
15	DIF	S_TDIF	Control de diferencia dimensional [2), 4)]	
16	TUL	S_TUL	Límite superior de tolerancia (incremental respecto al valor teórico) [4)]	
17	TLL	S_TLL	Límite inferior de tolerancia (incremental respecto al valor teórico) [4)]	
18	TMV	S_TMV	Zona de corrección con cálculo de valor medio [2)]	
19	FW	S_K	Factor de ponderación para cálculo de valor medio [2)]	
20	EVN	S_EVNUM	Número para la memoria de valores medios empíricos [2), 7)]	
21		S_MCBIT	reservado	
22		_DMODE	Modo display	
			Valores:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
23		_AMODE	Modo alternativo	
			Valores:	UNIDADES: tolerancia dimensional sí/no 0 = no 1 = sí

1) Todos los valores predeterminados = 0 o están identificados con predeterminado = x

2) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Corrección en DO solo posible con medición sin giro

4) En la corrección de herramienta, considerar el dato de máquina específico de canal MD 20360 TOOL_PARAMETER_DEF_MASK bit 0 y bit 1

5) Solo si la función "Corrección aditiva de preparación" está configurada en el dato general de máquina MD 18108 $MN_MM_NUM_SUMCORR . Además, en el dato general de máquina MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK debe estar ajustado bit 8 = 1.

6) Si no está configurado el DO "fino" en los DM, se corrige según el DO "grueso".

7) Cálculo de valor medio empírico solo posible en la corrección de herramienta
Rango de valores de la memoria de valores medios empíricos:
1 a 20 números (n) de la memoria de valores empíricos, ver dato de operador específico de canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
10000 a 200000 números (n) de la memoria de valores medios, ver dato de operador específico de canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

3.25.3.3 Parámetros de ciclo de medida 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)
```

Tabla 3-10 Parámetros de llamada CYCLE994 [1)]

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
1		S_MVAR	Variante de medida	
			Valo-res:	UNIDADES: medición interior o exterior (predeterminado = 1) 1 = medición interior 2 = medición exterior
				DECENAS: reservado
				CENTENAS: objetivo de corrección 0 = solo medición (sin corrección de DO ni corrección de herramienta) 1 = medición, determinación y corrección de DO (ver S_KNUM)[3)] 2 = medición y corrección de herramienta (ver S_KNUM1)
				MILLARES: zona para rodear 0 = sin zona para rodear 1 = Eje para rodear 1.er eje del plano (con G18 Z). Eje de medición, ver "S_MA". 2 = Eje para rodear 2.º eje del plano (con G18 X). Eje de medición, ver "S_MA". 3 = Eje para rodear 3.er eje real del plano (con G18 Y). Eje de medición, ver "S_MA". [8)]
2	Selección	S_KNUM	Corrección decalaje de origen (DO), DO base o referencia base[2)]	
			Valo-res:	UNIDADES:
				DECENAS: 0 = sin corrección 1 a máx. 99 números del decalaje de origen o 1 a máx. 16 números del decalaje base
				CENTENAS: reservado
				MILLARES: Corrección DO, base o referencia base 0 = corrección DO ajustable 1 = corrección DO base específico de canal 2 = corrección referencia base 3 = corrección DO base global 9 = corrección DO activo o, con G500, en último DO base específico de canal activo
				DECENAS DE MILLAR: Corrección gruesa o fina en DO, DO base o referencia base 0 = corrección fina [6)] 1 = corrección gruesa

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
3	Selección	S_KNUM1	Corrección en corrección de herramienta [2), 4)]	
			Valo-res:	UNIDADES:
				DECENAS:
				CENTENAS: 0 = sin corrección 1 a máx. 999 números D (número de filo) con corrección de herramienta; para corrección aditiva y de preparación, ver también S_DLNUM
				MILLARES: 0 o bien números D unívocos
				DECENAS DE MILLAR: 0 o bien números D unívocos de 1 a máx. 32.000 si hay configurados números D unívocos en DM
				CENTENAS DE MILLAR: corrección de herramienta [2)] 0 = sin indicación (corrección de geometría de herramienta) 1 = corrección de longitud L1 2 = corrección de longitud L2 3 = corrección de longitud L3 4 = corrección de radio
				UNIDADES DE MILLÓN: corrección de herramienta [2)] 0 = sin indicación (corrección del desgaste de la longitud de herramienta) 1 = corrección de herramienta, corrección aditiva (SK) [5)] El valor de corrección de herramienta se suma a la SK existente 2 = corrección de herramienta, corrección de preparación (EK) [5)] EK (nueva) = EK (antigua) + valor de corrección SK (antigua), SK (nueva) = 0 3 = corrección de herramienta, corrección de preparación (EK) [5)] El valor de corrección de herramienta se suma a la EK existente 4 = corrección de herramienta, geometría
				DECENAS DE MILLÓN: corrección de herramienta [2)] 0 = sin indicación (corrección en geometría de herramienta normal, no invertida) 1 = corrección invertida
				CENTENAS DE MILLÓN: corrección de herramienta 0 = corrección de herramienta sin herramienta Duplo 1 = corrección de herramienta en la herramienta Duplo (_DP)
4	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número del palpador) (predeterminado = 1)	
5	X0	S_SETV	Valor teórico	
6	X	S_MA	Número del eje de medición (predeterminado = 1)	
			Valo-res:	1 = 1.er eje del plano (con G18 Z) 2 = 2.º eje del plano (con G18 X) 3 = 3.er eje del plano (con G18 Y) [8)]
7	X1	S_SZA	Extensión de rodeo en el eje de medición	
8	Y1	S_SZO	Extensión de rodeo en el eje para rodear	
9	DFA	S_FA	Trayecto de medición	
10	TSA	S_TSA	Zona de confianza	
11	Mediciones	S_NMSP	Número de mediciones en la misma posición [2)] (predeterminado = 1)	
12	T	S_TNAME	Nombre de herramienta [2)]	
13	DL	S_DLNUM	Número DL, corrección aditiva de preparación [5)]	
14	ST	_DP	Número de la herramienta de sustitución a corregir (número Duplo)	

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
15	TZL	S_TZL	Corrección cero [2), 4)]	
16	DIF	S_TDIF	Control de diferencia dimensional [2), 4)]	
17	TUL	S_TUL	Límite superior de tolerancia (incremental respecto al valor teórico) [4)]	
18	TLL	S_TLL	Límite inferior de tolerancia (incremental respecto al valor teórico) [4)]	
19	TMV	S_TMV	Zona de corrección con cálculo de valor medio [2)]	
20	FW	S_K	Factor de ponderación para cálculo de valor medio [2)]	
21	EVN	S_EVNUM	Número para la memoria de valores empíricos [2), 7)]	
22		S_MCBIT	reservado	
23		_DMODE	Modo display	
			Valores:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
24		_AMODE	Modo alternativo	
			Valores:	UNIDADES: tolerancia dimensional sí/no 0 = no 1 = sí

1) Todos los valores predeterminados = 0 o están identificados con predeterminado = x

2) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Corrección en DO solo posible con medición sin giro

4) En la corrección de herramienta, considerar el dato de máquina de canal MD 20360 TOOL_PARAMETER_DEF_MASK

5) Solo si la función "Corrección aditiva de preparación" está configurada en el dato general de máquina MD 18108 $MN_MM_NUM_SUMCORR . Además, en el dato general de máquina MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK debe estar ajustado bit 8 = 1.

6) Si no está configurado el DO "fino" en los DM, se corrige según el DO "grueso".

7) Cálculo de valor medio empírico solo posible en la corrección de herramienta
Rango de valores de la memoria de valores medios empíricos:
1 a 20 números (n) de la memoria de valores empíricos, ver dato de operador específico de canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
10000 a 200000 números (n) de la memoria de valores medios, ver dato de operador específico de canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

8) Si eje Y está disponible en la máquina

3.25.3.4 Parámetros de ciclo de medida 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)
```

Tabla 3-11 Parámetros de llamada CYCLE976 [1)]

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
1		S_MVAR	Variante de medida (predeterminado = 1000)	
			Valores:	UNIDADES: calibración en superficie, en esfera de calibración o en anillo de calibración [2)] 0 = longitud en superficie con valor teórico conocido 1 = radio en anillo de calibración con diámetro conocido (valor teórico) y centro conocido. 2 = radio en anillo de calibración con diámetro conocido (valor teórico) y centro desconocido 3 = radio y longitud en esfera de calibración 4 = radio en borde con valor teórico conocido. Considerar la selección de eje de medición y sentido de medición. [3)] 5 = radio entre dos bordes con conocimiento de valor teórico y distancia entre bordes. Debe seleccionarse el eje de medición.
				DECENAS: reservado 0 = 0
				CENTENAS: reservado 0 = 0
				MILLARES: selección de eje de medición y sentido de medición al calibrar 0 = sin indicación (la selección de eje de medición y sentido de medición no es necesaria) [8)] 1 = indicar selección de eje de medición y sentido de medición, ver S_MA, S_MD (un sentido de medición en un eje de medición) 2 = indicar selección de eje de medición, ver S_MA (dos sentidos de medición en un eje de medición)
				DECENAS DE MILLAR: determinación de desviación de posición (inclinación del palpador) [2)] 0 = determinar desviación de posición del palpador [6)] 1 = no determinar desviación de posición
				CENTENAS DE MILLAR: calibración en paralelo al eje o en ángulo 0 = calibración en paralelo al eje en WKS activo 1 = calibración en ángulo [7)]
				UNIDADES DE MILLÓN: determinación de longitud de herramienta al calibrar en superficie o en esfera 0 = sin determinación de la longitud de herramienta 1 = determinación de la longitud de herramienta [4)] 2 = calibración del eje de penetración en la esfera, determinación de la longitud de herramienta, entrada de la diferencia de medida de las longitudes de herramienta en los datos de calibración
2	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número del palpador) (predeterminado = 1)	
3		S_SETV	Valor teórico	
4	Z0	S_SETV0	Valor teórico de la referencia de longitud con calibración de esfera	

N.°	Parámetro de máscara	Paráme-tro de ciclo	Significado	
5	X/Y/Z	S_MA	Eje de medición (número del eje) [2), 6)] (predeterminado = 1)	
			Valo-res:	1 = 1.er eje del plano (con G17 X) 2 = 2.º eje del plano (con G17 Y) 3 = 3.er eje del plano (con G17 Z)
6	+ -	S_MD	Sentido de medición [2), 6)]	
			Valo-res:	0 = positivo 1 = negativo
7	DFA	S_FA	Trayecto de medición	
8	TSA	S_TSA	Zona de confianza	
9	VMS	S_VMS	Velocidad de medición variable al calibrar [2)]	
10	α	S_STA1	Ángulo inicial [2), 5)]	
11	Mediciones	S_NMSP	Número de mediciones en la misma posición [2)] (predeterminado = 1)	
12	X0	S_SETV1	Punto de referencia borde con calibración entre 2 bordes [3)]	
13		_DMODE	Modo display	
			Valo-res:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
14		_AMODE	Modo alternativo	

1) Todos los valores predeterminados = 0 o están identificados con predeterminado = x

2) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Al efectuar la calibración "Radio en anillo de calibración", deben conocerse el diámetro y el centro del anillo (4 sentidos de medición).
Al efectuar la calibración "Radio en dos bordes", solo debe conocerse la distancia de los bordes en dirección al eje de medición (2 sentidos de medición).
Al efectuar la calibración "Radio en un borde", debe conocerse el valor teórico de la superficie.

4) Con la variante de medida de solo calibración en una superficie (longitud en superficie), la longitud de herramienta corregida se obtiene de S_MD y S_MA

5) Solo con la variante de medida "Anillo de calibración ... y centro conocido" (S_MVAR=1xxx02)

6) Eje de medición solo con variante de medida S_MVAR=0 o =xx1x01 o =xx2x01 o =20000
Variante de medida: "Calibración en una superficie" → selección de eje de medición y sentido de medición
o en "Anillo de calibración... y centro conocido" → selección de un sentido de eje y selección de eje de medición y sentido de medición
o en "Anillo de calibración... y centro conocido" → selección de dos sentidos de eje y selección de eje de medición
o "Determinación de longitud de palpador" → S_MA=3 → 3.er eje del plano (con G17 Z)

7) Variante de medida de solo calibración en anillo de calibración o en esfera de calibración
Con "Calibración en esfera de calibración" y medición en ángulo, la esfera se rodea en el ecuador.

8) Con la calibración "Radio en anillo de calibración" con centro desconocido, cuatro sentidos de medición en el plano (con G17 +-X +-Y).
Con la calibración "Longitud en superficie" en sentido negativo del eje de herramienta (con G17 -Z).

3.25.3.5 Parámetros de ciclo de medida 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)
```

Tabla 3-12 Parámetros de llamada CYCLE978 [1)]

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
1		S_MVAR	Variante de medida	
			Valo-res:	UNIDADES: elemento de contorno 0 = medir superficie
				DECENAS: reservado
				CENTENAS: objetivo de corrección 0 = solo medición (sin corrección de DO ni corrección de herramienta) 1 = medición, determinación y corrección de DO (ver S_KNUM) 2 = medición y corrección de herramienta (ver S_KNUM1)
				MILLARES: reservado
				DECENAS DE MILLAR: medir con/sin giro del cabezal o alinear el palpador en el sentido de conmutación [9)] 0 = medir sin giro del cabezal, sin alinear el palpador 1 = medir con giro del cabezal 2 = alinear palpador en el sentido de conmutación
2	Selección	S_KNUM	Corrección decalaje de origen (DO), DO base o referencia base[2)]	
			Valo-res:	UNIDADES:
				DECENAS: 0 = sin corrección 1 a máx. 99 números del decalaje de origen o 1 a máx. 16 números del decalaje base
				CENTENAS: reservado
				MILLARES: Corrección DO, base o referencia base 0 = corrección DO ajustable 1 = corrección DO base específico de canal 2 = corrección referencia base 3 = corrección DO base global 9 = corrección DO activo o, con G500, en último DO base específico de canal activo
				DECENAS DE MILLAR: Corrección gruesa o fina en DO, DO base o referencia base 0 = corrección fina [6)] 1 = corrección gruesa

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
3	Selección	S_KNUM1	Corrección en corrección de herramienta [2]	
			Valo-res:	UNIDADES:
				DECENAS:
				CENTENAS: 0 = sin corrección 1 a máx. 999 números D (número de filo) con corrección de herramienta; para corrección aditiva y de preparación, ver también S_DLNUM
				MILLARES: 0 o bien números D unívocos
				DECENAS DE MILLAR: 0 o bien números D unívocos de 1 a máx. 32.000 si hay configurados números D unívocos en los DM
				CENTENAS DE MILLAR: corrección de herramienta [2] 0 = sin indicación (corrección en geometría de herramienta) 1 = corrección de longitud L1 2 = corrección de longitud L2 3 = corrección de longitud L3 4 = corrección radio
				UNIDADES DE MILLÓN: corrección de herramienta [2] 0 = sin indicación (corrección del desgaste del radio de herramienta) 1 = corrección de herramienta, corrección aditiva (SK) [5] El valor de corrección de herramienta se suma a la SK existente 2 = corrección de herramienta, corrección de preparación (EK) [5] EK (nueva) = EK (antigua) + valor de corrección SK (antigua), SK (nueva) = 0 3 = corrección de herramienta, corrección de preparación (EK) [5] El valor de corrección de herramienta se suma a la EK existente 4 = corrección de herramienta, geometría
				DECENAS DE MILLÓN: corrección de herramienta [2] 0 = sin indicación (corrección en geometría de herramienta normal, no invertida) 1 = corrección invertida
				CENTENAS DE MILLÓN: corrección de herramienta 0 = corrección de herramienta sin herramienta Duplo 1 = corrección de herramienta en la herramienta Duplo (_DP)
4	icono+número	S_PRNUM	Número del campo de los parámetros de palpador (no número de palpador) (rango de valores 1 a 40)	
5	X0	S_SETV	Valor teórico	
6	DFA	S_FA	Trayecto de medición	
7	TSA	S_TSA	Zona de confianza	
8	X	S_MA	Número del eje de medición [7] (rango de valores 1 a 3)	
			Valo-res:	1 = 1.er eje del plano (con G17 X) 2 = 2.º eje del plano (con G17 Y) 3 = 3.er eje del plano (con G17 Z), medir en dirección de herramienta
9		S_MD	Sentido de medición del eje de medición	
			Valo-res:	1 = sentido de medición positivo 2 = sentido de medición negativo
10	Mediciones	S_NMSP	Número de mediciones en la misma posición [2] (rango de valores 1 a 9)	
11	TR	S_TNAME	Nombre de herramienta [3]	
12	DL	S_DLNUM	Número DL, corrección aditiva de preparación [5]	

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
13	ST	_DP	Número de la herramienta de sustitución a corregir (número Duplo)	
14	TZL	S_TZL	Corrección cero [2), 3)]	
15	DIF	S_TDIF	Control de diferencia dimensional [2), 3)]	
16	TUL	S_TUL	Límite superior de tolerancia (incremental respecto al valor teórico) [3)]	
17	TLL	S_TLL	Límite inferior de tolerancia (incremental respecto al valor teórico) [3)]	
18	TMV	S_TMV	Zona de corrección con cálculo de valor medio [2)]	
19	FW	S_K	Factor de ponderación para cálculo de valor medio [2)]	
20	EVN	S_EVNUM	Juego de datos para memoria de valores empíricos [2), 8)]	
21		S_MCBIT	reservado	
22		_DMODE	Modo display	
			Valores:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
23		_AMODE	Modo alternativo	
			Valores:	UNIDADES: tolerancia dimensional sí/no 0 = no 1 = sí

1) Todos los valores predeterminados = 0 o están identificados como rango de valores "a" a "b"

2) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Solo con corrección en herramienta; de lo contrario, el parámetro = ""

4) Solo con corrección en herramienta y tolerancia dimensional "Sí"; de lo contrario, el parámetro = 0

5) Solo si la función "Corrección aditiva de preparación" está configurada en el dato general de máquina MD 18108 $MN_MM_NUM_SUMCORR . Además, en el dato general de máquina MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK debe estar ajustado bit 8 = 1.

6) Si no está configurado el DO "fino" en los DM, se corrige según el DO "grueso".

7) Corrección en geometría de herramienta:
- al medir en el plano (S_MA=1 o S_MA=2), corrección de radio de herramienta
- al medir en dirección de herramienta (S_MA=3) corrección de longitud de herramienta L1

8) El cálculo de valor medio empírico es posible en la corrección de herramienta y la corrección en DO.
Rango de valores de la memoria de valores medios empíricos:
1 a 20 números (n) de la memoria de valores empíricos, ver dato de operador específico de canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
10000 a 200000 números (n) de la memoria de valores medios, ver dato de operador específico de canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

9) Al medir con giro del cabezal, el radio/diámetro del palpador debe haberse determinado con exactitud. Esto debe llevarse a cabo con una variante de calibración de CYCLE976: radio en anillo, en borde o en esfera. En caso contrario se altera el resultado de medición.

3.25.3.6 Parámetros de ciclo de medida 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)
```

Tabla 3-13 Parámetros de llamada CYCLE998 [1)]

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
1		S_MVAR	Variante de medida (predeterminado = 5)	
			Valores:	UNIDADES: elemento de contorno 5 = medir borde (un ángulo) 6 = medir plano (dos ángulos)
				DECENAS: reservado
				CENTENAS: objetivo de corrección 0 = solo medición y ninguna corrección de DO 1 = medición, determinación y corrección de DO (ver S_KNUM)
				MILLARES: Zona protegida 0 = sin considerar una zona protegida 1 = considerando una zona protegida
				DECENAS DE MILLAR: medición con giro de cabezal (medición diferencial) 0 = medición sin giro de cabezal 1 = medición con giro de cabezal
				CENTENAS DE MILLAR: medición en ángulo o en paralelo al eje 0 = medición en ángulo 1 = medición en paralelo al eje
2	Selección	S_KNUM	Corrección decalaje de origen (DO), DO base o referencia base[2)]	
			Valores:	UNIDADES:
				DECENAS: 0 = sin corrección 1 a máx. 99 números del decalaje de origen o 1 a máx. 16 números del decalaje base
				CENTENAS: reservado
				MILLARES: Corrección DO, base o referencia base 0 = corrección DO ajustable 1 = corrección DO base específico de canal 2 = corrección referencia base 9 = corrección DO activo o, con G500, en último DO base específico de canal activo
				DECENAS DE MILLAR: Corrección gruesa o fina en DO, DO base o referencia base[3)] 0 = corrección fina 1 = corrección gruesa

N.°	Parámetro de máscara	Paráme-tro de ciclo	Significado	
3	A, B, C	S_RA	Objetivo de corrección: giro de coordenadas o eje giratorio[14)]	
			Valo-res:	0 = objetivo de corrección giro de coordenadas en torno al eje que se obtiene del parámetro S_MA [4)] >0 = objetivo de corrección eje giratorio. Número de eje de canal para el eje giratorio (preferentemente mesa giratoria). La corrección angular se efectúa en la parte de traslación del DO del eje giratorio.
4	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (predeterminado = 1)	
5	DX/DY/DZ	S_SETV	Trayecto (incremental) de la posición inicial al punto de medida P1 del eje de medición (S_MA) [5)]	
6	α	S_STA1	Valor teórico de ángulo con "Alinear borde" o "Alinear plano" en torno al 1.er eje del plano (con G17 X)[9)]	
7	β	S_INCA	Valor teórico de ángulo con "Alinear plano" en torno al 2.º eje del plano (con G17 Y)[9)]	
8	DFA	S_FA	Trayecto de medición	
9	TSA	S_TSA	Zona de confianza Vigilancia de la diferencia de ángulo con el valor teórico de ángulo [grados] [6)]	
10	X/Y/Z	S_MA	Eje de medición, eje de desplazamiento [7)] (predeterminado = 201)	
			Valo-res:	UNIDADES: número del eje de medición 1 = 1.er eje del plano (con G17 X) 2 = 2.º eje del plano (con G17 Y) 3 = 3.er eje del plano (con G17 Z)
				DECENAS: reservado
				CENTENAS: número del eje de desplazamiento 1 = 1.er eje del plano (con G17 X) 2 = 2.º eje del plano (con G17 Y) 3 = 3.er eje del plano (con G17 Z)
11	+-	S_MD	Sentido de medición del eje de medición [8)]	
			Valo-res:	0 = el sentido de medición se determina a partir del valor teórico y la posición real del eje de medición (compatibilidad) 1 = sentido de medición positivo 2 = sentido de medición negativo
12	L2	S_ID	Con variante de medida "Alinear borde": Distancia (incremental) entre los puntos de medida P1 y P2 en el eje de desplazamiento (valor > 0) Con la variante de medida "Alinear plano" se aplican los parámetros indicados a continuación.	
13	L2	S_SETV0	Distancia entre los puntos de medida P1 y P2 en el 1.er eje del plano [10)]	
14		S_SETV1	Distancia entre los puntos de medida P1 y P2 en el 2.º eje del plano [11), 12)]	
15	L3x	S_SETV2	Distancia entre los puntos de medida P1 y P3 en el 1.er eje del plano [11)]	
16	L3y	S_SETV3	Distancia entre los puntos de medida P1 y P3 en el 2.º eje del plano [10)]	
17	Mediciones	S_NMSP	Número de mediciones en la misma posición [2)] (predeterminado = 1)	
18		S_EVNUM	Juego de datos para memoria de valores empíricos [2), 13)]	

N.°	Parámetro de máscara	Paráme-tro de ciclo	Significado	
19		_DMODE	Modo display	
			Valo-res:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
20		_AMODE	Reservado (modo alternativo)	

1) Todos los valores predeterminados = 0 o están identificados con predeterminado = x

2) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) DO "fino" solo con objetivo de corrección eje giratorio y MD 52207 $MCS_AXIS_USAGE_ATTRIB[n] , bit 6 = 1.
Si no está configurado el DO en los DM, se corrige según el DO "grueso".

4) Ejemplo de corrección en giro de coordenadas: S_MA=102 Con eje de medición Y y eje de desplazamiento X resulta un giro de coordenadas en torno a Z (con G17).

5) Valor solo relevante con zona protegida "Sí" (S_MVAR dígito de MILLARES = 1)

6) Si se posiciona el punto de medida P1 en el punto de medida P2 en el eje de desplazamiento, se suman los ángulos a los parámetros S_STA1 y S_TSA .

7) El número del eje de medición debe ser diferente al número del eje de desplazamiento (p. ej., 101 no permitido).

8) Sentido de medición solo "Alinear borde" y "Medición en paralelo al eje" (S_MVAR=10x105)

9) Rango angular S_STA1 ±45 grados con "Alinear borde"
Rango angular S_STA1 0 a +60 grados y S_INCA ±30 grados con "Alinear plano"

10) Con variante de medida "Alinear plano" y "Alinear borde"

11) Con variante de medida "Medir plano" y "Medición en paralelo al eje"

12) No para versión del SW de ciclos de medida 04.04.

13) Cálculo del valor empírico con corrección en DO; rango de valores de la memoria de valores medios empíricos:
1 a 20 números (n) de la memoria de valores empíricos, ver dato de operador específico de canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]

14) El parámetro S_RA solo es relevante para la variante de medida "Alinear borde" (S_MVAR xxxxx5).

3.25.3.7 Parámetros de ciclo de medida 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)
```

Tabla 3-14 Parámetros de llamada CYCLE977 [1)]

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
1		S_MVAR	Variante de medida	
			Valores:	UNIDADES: elemento de contorno (rango de valores 1 a 6) 1 = medir taladro 2 = medir saliente (eje) 3 = medir ranura 4 = medir nervio 5 = medir rectángulo interior 6 = medir rectángulo exterior
				DECENAS: reservado
				CENTENAS: objetivo de corrección 0 = solo medición (sin corrección de DO ni corrección de herramienta) 1 = medición, determinación y corrección de DO (ver S_KNUM) 2 = medición y corrección de herramienta (ver S_KNUM1)
				MILLARES: zona protegida 0 = sin considerar una zona protegida 1 = considerando una zona protegida
				DECENAS DE MILLAR: medir con/sin giro del cabezal (medición diferencial) o alinear el palpador en el sentido de conmutación 0 = medir sin giro del cabezal, no alinear el palpador 1 = medir con giro del cabezal 2 = alinear palpador en el sentido de conmutación
2	Selección	S_KNUM	Corrección decalaje de origen (DO), DO base o referencia base[2)]	
			Valores:	UNIDADES:
				DECENAS: 0 = sin corrección 1 a máx. 99 números del decalaje de origen o 1 a máx. 16 números del decalaje base
				CENTENAS: reservado
				MILLARES: Corrección DO, base o referencia base 0 = corrección DO ajustable 1 = corrección DO base específico de canal 2 = corrección referencia base 3 = corrección DO base global 9 = corrección DO activo o, con G500, en último DO base específico de canal activo
				DECENAS DE MILLAR: Corrección gruesa o fina en DO, DO base o referencia base 0 = corrección fina [6)] 1 = corrección gruesa

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
3	Selección	S_KNUM1	Corrección en corrección de herramienta [2]	
			Valores:	UNIDADES:
				DECENAS:
				CENTENAS: 0 = sin corrección 1 a máx. 999 números D (número de filo) para corrección de herramienta; para corrección aditiva y de preparación, ver también S_DLNUM
				MILLARES: 0 o bien números D unívocos
				DECENAS DE MILLAR: 0 o bien números D unívocos de 1 a máx. 32.000 si hay configurados números D unívocos en los DM
				CENTENAS DE MILLAR: corrección de herramienta [2] 0 = sin indicación (corrección de radio de herramienta) 1 = corrección de longitud L1 2 = corrección de longitud L2 3 = corrección de longitud L3 4 = corrección radio
				UNIDADES DE MILLÓN: corrección de herramienta [2] 0 = sin indicación (corrección del desgaste del radio de herramienta) 1 = corrección de herramienta, corrección aditiva (SK) [5] El valor de corrección de herramienta se suma a la SK existente 2 = corrección de herramienta, corrección de preparación (EK) [5] EK (nueva) = EK (antigua) + valor de corrección SK (antigua), SK (nueva) = 0 3 = corrección de herramienta, corrección de preparación (EK) [5] El valor de corrección de herramienta se suma a la EK existente 4 = corrección de herramienta, geometría
				DECENAS DE MILLÓN: corrección de herramienta [2] 0 = sin indicación (corrección en geometría de herramienta normal, no invertida) 1 = corrección invertida
				CENTENAS DE MILLÓN: corrección de herramienta 0 = corrección de herramienta sin herramienta Duplo 1 = corrección de herramienta en la herramienta Duplo (_DP)
4	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número de palpador) (rango de valores 1 a 40)	
5	X0	S_SETV	Valor teórico	
6	X0	S_SETV0	Valor teórico en rectángulo 1.er eje del plano (X con G17)	
7	Y0	S_SETV1	Valor teórico en rectángulo 2.º eje del plano (Y con G17)	
8	XM	S_XM	Especificación de valor teórico para centro, eje geométrico X	
9	YM	S_YM	Especificación de valor teórico para centro, eje geométrico Y	
10	DFA	S_FA	Trayecto de medición	
11	TSA	S_TSA	Zona de confianza	
12	α 0	S_STA1	Ángulo inicial	

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
13	DZ	S_ID	Magnitud incremental 1. Penetración incremental del 3.er eje del plano (Z con G17) Dirección de penetración mediante signo de S_ID. En la medición de saliente, nervio y rectángulo exterior, con S_ID se define el descenso a la altura de medida. 2. Consideración de una zona protegida En la medición de taladro, ranura y rectángulo interior y con una zona protegida, con S_ID se define la altura de rebasamiento.	
14	X1	S_SZA	Diámetro o longitud (ancho) de la zona protegida [7]	
15	Y1	S_SZO	En la medición de rectángulo: ancho de la zona protegida del 2.º eje del plano	
16	X	S_MA	Número del eje de medición [7] (solo en la medición de ranura o nervio)	
			Valores:	1 = 1.er eje del plano (con G17 X) 2 = 2.º eje del plano (con G17 Y).
17	ST	_DP	Número de la herramienta de sustitución a corregir (número Duplo)	
18	Mediciones	S_NMSP	Número de mediciones en la misma posición [2] (rango de valores 1 a 9)	
19	TR	S_TNAME	Nombre de herramienta [2]	
20	DL	S_DLNUM	Número DL, corrección aditiva de preparación [5]	
21	TZL	S_TZL	Corrección cero [2, 4]	
22	DIF	S_TDIF	Control de diferencia dimensional [2, 4]	
23	TUL	S_TUL	Límite superior de tolerancia (incremental respecto al valor teórico) [4]	
24	TLL	S_TLL	Límite inferior de tolerancia (incremental respecto al valor teórico) [4]	
25	TMV	S_TMV	Zona de corrección con cálculo de valor medio [2]	
26	FW	S_K	Factor de ponderación para cálculo de valor medio [2]	
27		S_EVNUM	Juego de datos para memoria de valores medios empíricos [2, 8]	
28		S_MCBIT	reservado	
29		_DMODE	Modo display	
			Valores:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
30		_AMODE	Modo alternativo	
			Valores:	UNIDADES: tolerancia dimensional sí/no 0 = no 1 = sí

[1] Todos los valores predeterminados = 0 o están identificados como rango de valores "a" a "b"

[2] Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

[3] Solo con corrección en herramienta; de lo contrario, el parámetro = ""

[4] Solo con corrección en herramienta y tolerancia dimensional "Sí"; de lo contrario, el parámetro = 0

[5] Solo si la función "Corrección aditiva de preparación" está configurada en el dato general de máquina MD 18108 $MN_MM_NUM_SUMCORR . Además, en el dato general de máquina MD 18080 $MN_MM_TOOL_MANAGEMENT_MASK debe estar ajustado bit 8 = 1.

[6] Si no está configurado el DO "fino" en los DM, se corrige según el DO "grueso".

[7] Diámetro o ancho de zona protegida dentro de taladro o ranura.
Diámetro o ancho de zona protegida fuera de saliente o nervio.

[8] Cálculo de valor medio empírico posible en la corrección de herramienta
Rango de valores de la memoria de valores medios empíricos:
1 a 20 números (n) de la memoria de valores empíricos, ver dato de operador específico de canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
10000 a 200000 números (n) de la memoria de valores medios, ver dato de operador específico de canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

3.25.3.8 Parámetros de ciclo de medida 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)
```

Tabla 3-15 Parámetros de llamada CYCLE961 [1)]

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
1		S_MVAR	Variante de medida (predeterminado ≥ 6)	
			Valores:	UNIDADES: elemento de contorno 5 = preparación de esquina interior en ángulo recto, especificación de valor teórico ángulo y distancias de A1 a A3 6 = preparación de esquina exterior en ángulo recto, especificación de valor teórico ángulo y distancias de A1 a A3 7 = preparación de esquina interior, especificación ángulo y distancias de A1 a A4 8 = preparación de esquina exterior, especificación ángulo y distancias de A1 a A4
				DECENAS: especificación de valor teórico como distancia o mediante cuatro puntos 0 = especificación de valor teórico como distancia (polar) 1 = especificación de valor teórico mediante cuatro puntos (puntos de medida P1 a P4)
				CENTENAS: objetivo de corrección 0 = solo medición (sin corrección de DO ni corrección de herramienta) 1 = medición, determinación y corrección de DO, ver S_KNUM
				MILLARES: zona protegida 0 = sin considerar una zona protegida (obstáculo) 1 = considerando una zona protegida (obstáculo), ver S_ID
				DECENAS DE MILLAR: posición de esquina en WKS 0 = la posición de la esquina se determina mediante el parámetro S_STA1 (compatibilidad) 1 = posición 1 de la esquina en punto inicial posicionado de la medición[6)] 2 = posición 2 de la esquina, las distancias en el 1.er eje del plano (con G17 X) son negativas (ver S_SETV0, S_SETV1) 3 = posición 3 de la esquina, las distancias en el 1.er y 2.º eje del plano (con G17 XY) son negativas (ver S_SETV0 a S_SETV3) 4 = posición 4 de la esquina, las distancias en el 2.º eje del plano (con G17 Y) son negativas (ver S_SETV2, S_SETV3)

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
2	Selección	S_KNUM	Corrección decalaje de origen (DO), DO base o referencia base[2]	
			Valores:	UNIDADES:
				DECENAS: 0 = sin corrección 1 a máx. 99 números del decalaje de origen o 1 a máx. 16 números del decalaje base
				CENTENAS: reservado
				MILLARES: Corrección DO, base o referencia base 0 = corrección DO ajustable 1 = corrección DO base específico de canal 2 = corrección referencia base 9 = corrección DO activo o, con G500, último DO base específico de canal activo
				DECENAS DE MILLAR: Corrección gruesa o fina en DO, DO base o referencia base 0 = corrección fina [5] 1 = corrección gruesa
3	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número de palpador) (rango de valores 1 a 40)	
4	L1/X1	S_SETV0	Distancia L1 entre el polo y el punto de medida P1 en la dirección del 1.er eje del plano (con G17 X) [3] (si la distancia L1 = 0, se calcula automáticamente L1 = M_SETV1/2) o punto inicial P1x del 1.er eje del plano (con G17 X) [4]	
5	L2/Y1	S_SETV1	Distancia L2 entre el polo y el punto de medida P2 en la dirección del 1.er eje del plano [3] o punto inicial P1y del 2.º eje del plano (con G17 Y) [4]	
6	L3/X2	S_SETV2	Distancia L3 entre el polo y el punto de medida P3 en la dirección del 2.º eje del plano [3] (si la distancia L3 = 0, se calcula automáticamente L3 = M_SETV3/2 si la esquina no forma ángulo recto) o punto inicial P2x del 1.er eje del plano [4]	
7	L4/Y2	S_SETV3	Distancia L4 entre el polo y el punto de medida P3 en la dirección del 2.º eje del plano si la esquina no forma ángulo recto [3] o punto inicial P2y del 2.º eje del plano [4]	
8	XP/X3	S_SETV4	Posición del polo en el 1.er eje del plano [3] o punto inicial P3x del 1.er eje del plano [4]	
9	XP/Y3	S_SETV5	Posición del polo en el 2.º eje del plano [3] o punto inicial P3y del 2.º eje del plano [4]	
10	X4	S_SETV6	Punto inicial P4x del 1.er eje del plano [4]	
11	Y4	S_SETV7	Punto inicial P4y del 2.º eje del plano [4]	
12	X0	S_SETV8	Valor teórico de la esquina medida en el 1.er eje del plano al corregir en DO	
13	Y0	S_SETV9	Valor teórico de la esquina medida en el 2.º eje del plano al corregir en DO	
14	α0	S_STA1	Ángulo inicial en sentido positivo del 1.er eje del plano con respecto al borde de referencia de la pieza en MKS (+-270 grados)	
15	α1	S_INCA	Ángulo entre bordes de referencia de la pieza al medir una esquina que no forma ángulo recto [7]	
16	DZ	S_ID	Penetración a la altura de medida en cada punto de medida con zona protegida activa (ver S_MVAR).	
17	DFA	S_FA	Trayecto de medición	

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
18	TSA	S_TSA	Zona de confianza Vigilancia de la diferencia de ángulo con el valor teórico de ángulo [grados]	
19	Mediciones	S_NMSP	Número de mediciones en la misma posición [2] (rango de valores 1 a 9) [2]	
20		S_MCBIT	reservado	
21		_DMODE	Modo display	
			Valores:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
22		_AMODE	Modo alternativo	

1) Todos los valores predeterminados = 0 o están identificados como rango de valores "a" a "b"

2) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Especificación de puntos de medida en coordenadas polares considerando el ángulo inicial S_STA1 en el punto de medida 3 ó 4 del ángulo de incremento S_INCA.

4) Especificación de puntos de medida en sistema de coordenadas rectangular (especificación mediante 4 puntos)

5) Si no está configurado el DO "fino" en los DM, se corrige según el DO "grueso".

6) Rango de valores de ángulo S_INCA: -180 a +180 grados

7) Ángulo inicial S_STA1, rango de valores: esquina rectangular: +-90 grados, cualquier esquina: +-45 grados

3.25.3.9 Parámetros de ciclo de medida 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)
```

Tabla 3-16 Parámetros de llamada CYCLE979 [0)]

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
1		S_MVAR	Variante de medida	
			Valores:	UNIDADES: elemento de contorno 1 = medir taladro 2 = medir saliente (eje)
				DECENAS: reservado
				CENTENAS: objetivo de corrección 0 = solo medición (sin corrección de DO ni corrección de herramienta) 1 = medición, determinación y corrección de DO (ver S_KNUM) 2 = medición y corrección de herramienta (ver S_KNUM1)
				MILLARES: número de puntos de medida 0 = 3 puntos de medida 1 = 4 puntos de medida
				DECENAS DE MILLAR: medir con/sin giro del cabezal (medición diferencial) o alinear el palpador en el sentido de conmutación 0 = medir sin giro del cabezal, sin alinear el palpador 1 = medir con giro del cabezal 2 = alinear palpador en el sentido de conmutación
2	Selección	S_KNUM	Corrección decalaje de origen (DO), DO base o referencia base[2)]	
			Valores:	UNIDADES:
				DECENAS: 0 = sin corrección 1 a máx. 99 números del decalaje de origen o 1 a máx. 16 números del decalaje base
				CENTENAS: reservado
				MILLARES: Corrección DO, base o referencia base 0 = corrección DO ajustable 1 = corrección DO base específico de canal 2 = corrección referencia base 3 = corrección DO base global 9 = corrección DO activo o, con G500, en último DO base específico de canal activo
				DECENAS DE MILLAR: Corrección gruesa o fina en DO, DO base o referencia base 0 = corrección fina [6)] 1 = corrección gruesa

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
3	Selección	S_KNUM1	Corrección en corrección de herramienta [2]	
			Valores:	UNIDADES:
				DECENAS:
				CENTENAS: 0 = sin corrección 1 a máx. 999 números D (número de filo) para corrección de herramienta; para corrección aditiva y de preparación, ver también S_DLNUM
				MILLARES: 0 o bien números D unívocos
				DECENAS DE MILLAR: 0 o bien números D unívocos de 1 a máx. 32.000 si hay configurados números D unívocos en los DM
				CENTENAS DE MILLAR: corrección de herramienta [2] 0 = sin indicación (corrección en radio de herramienta) 1 = corrección de longitud L1 2 = corrección de longitud L2 3 = corrección de longitud L3 4 = corrección radio
				UNIDADES DE MILLÓN: corrección de herramienta [2] 0 = sin indicación (corrección del desgaste del radio de herramienta) 1 = corrección de herramienta, corrección aditiva (SK) [5] El valor de corrección de herramienta se suma a la SK existente 2 = corrección de herramienta, corrección de preparación (EK) [5] EK (nueva) = EK (antigua) + valor de corrección SK (antigua), SK (nueva) = 0 3 = corrección de herramienta, corrección de preparación (EK) [5] El valor de corrección de herramienta se suma a la EK existente 4 = corrección de herramienta, geometría
				DECENAS DE MILLÓN: corrección de herramienta [2] 0 = sin indicación (corrección en geometría de herramienta normal, no invertida) 1 = corrección invertida
				CENTENAS DE MILLÓN: corrección de herramienta 0 = corrección de herramienta sin herramienta Duplo 1 = corrección de herramienta en la herramienta Duplo (_DP)
4	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número de palpador) (rango de valores 1 a 40)	
5	X0	S_SETV	Valor teórico	
6	DFA	S_FA	Trayecto de medición	
7	TSA	S_TSA	Zona de confianza	
8	X0	S_CPA	Centro del 1.er eje del plano (con G17 X)	
9	Y0	S_CPO	Centro del 2.º eje del plano (con G17 Y)	
10	XM	S_XM	Especificación de valor teórico para centro, eje geométrico X	
11	YM	S_YM	Especificación de valor teórico para centro, eje geométrico Y	
12	alfa 0	S_STA1	Ángulo inicial [7]	
13	alfa 1	S_INCA	Ángulo de incremento [8]	
14	Mediciones	S_NMSP	Número de mediciones en la misma posición [1] (rango de valores 1 a 9)	
15	ST	_DP	Número de la herramienta de sustitución a corregir (número Duplo)	
16	T	S_TNAME	Nombre de herramienta [2]	

N.°	Parámetros de máscara	Parámetros de ciclo	Significado	
17	DL	S_DLNUM	Número DL, corrección aditiva de preparación [1), 4)]	
18	TZL	S_TZL	Corrección cero [1), 2)]	
19	DIF	S_TDIF	Control de diferencia dimensional [1), 2)]	
20	TUL	S_TUL	Límite superior de tolerancia (incremental respecto al valor teórico) [2)]	
21	TLL	S_TLL	Límite inferior de tolerancia (incremental respecto al valor teórico) [2)]	
22	TMV	S_TMV	Zona de corrección con cálculo de valor medio [1)]	
23	FW	S_K	Factor de ponderación para cálculo de valor medio [1)]	
24		S_EVNUM	Juego de datos para memoria de valores empíricos [1), 6)]	
25		S_MCBIT	reservado	
26		_DMODE	Modo display	
			Valores:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
27		_AMODE	Modo alternativo	
			Valores:	UNIDADES: tolerancia dimensional sí/no 0 = no 1 = sí

0) Todos los valores predeterminados = 0 o están identificados como rango de valores "a" a "b".

1) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

2) Solo con corrección en herramienta; de lo contrario, el parámetro = ""

3) Solo con corrección en herramienta y tolerancia dimensional "Sí"; de lo contrario, el parámetro = 0

4) Solo si la función "Corrección aditiva de preparación" está configurada en el dato general de máquina MD 18108 $MN_MM_NUM_SUMCORR .

5) Si no está configurado el DO "fino" en los DM, se corrige según el DO "grueso".

6) Cálculo de valor medio empírico solo posible en la corrección de herramienta
Rango de valores de la memoria de valores medios empíricos:
1 a 20 números (n) de la memoria de valores empíricos, ver dato de operador específico de canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1]
10000 a 200000 números (n) de la memoria de valores medios, ver dato de operador específico de canal SD55625 $SCS_MEA_AVERAGE_VALUE[n-1]

7) Rango de valores de ángulo inicial: -360 a +360 grados.

8) Rango de valores de ángulo de incremento: >0 a ≤90 grados con 4 puntos de medida o >0 a ≤120 grados con 3 puntos de medida.

3.25.3.10 Parámetros de ciclo de medida 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)
```

Tabla 3-17 Parámetros de llamada CYCLE997 [1), 2)]

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
1		S_MVAR	Variante de medida (predeterminado = 9)	
			Valores:	UNIDADES: elemento de contorno 9 = medir esfera
				DECENAS: Repetición de la medida 0 = sin repetición de la medida 1 = con repetición de la medida
				CENTENAS: objetivo de corrección 0 = solo medición (ninguna corrección de DO) 1 = medición, determinación y corrección de DO (ver S_KNUM)
				MILLARES: Estrategia de medición 0 = medición en paralelo al eje, sin ángulo inicial, alineación del palpador conforme a SD55740, bit 1 1 = medición girando alrededor, con ángulo inicial, alineación del palpador conforme a SD55740, bit 1 2 = medición girando alrededor, con ángulo inicial, alinear el palpador en el sentido de conmutación 3 = medición en paralelo al eje, con ángulo inicial, alineación del palpador conforme a SD55740, bit 1 4 = medición en paralelo al eje, con ángulo inicial, alinear el palpador en el sentido de conmutación
				DECENAS DE MILLAR: número de esferas que van a medirse 0 = medir una esfera 1 = medir tres esferas
				CENTENAS DE MILLAR: número de puntos de medida, solo con medición en ángulo (considerar estrategia de medición: dígito de MILLARES > 0) 0 = tres puntos de medida al medir en ángulo (giro alrededor de la esfera) 1 = cuatro puntos de medida al medir en ángulo (giro alrededor de la esfera)
				UNIDADES DE MILLÓN: determinación del diámetro teórico de la esfera 0 = sin determinación del diámetro teórico de la esfera 1 = determinación del diámetro teórico de la esfera

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
2.	Selección	S_KNUM	Corrección en decalaje de origen (DO), DO base o referencia base[3]	
			Valores:	UNIDADES:
				DECENAS: 0 = sin corrección 1 a máx. 99 números del decalaje de origen o 1 a máx. 16 números del decalaje base
				CENTENAS: reservado
				MILLARES: Corrección en DO, DO base o referencia base 0 = corrección en DO ajustable 1 = corrección en DO base específico de canal 2 = corrección en referencia base 3 = corrección en DO base global[7] 9 = corrección en DO activo o, con G500, en último DO base específico de canal activo
				DECENAS DE MILLAR: corrección gruesa o fina en DO, DO base o referencia base 0 = corrección fina [6] 1 = corrección gruesa
3	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número de palpador) (rango de valores 1 a 40)	
4		S_SETV	Diámetro de las esferas [4]	
5	DFA	S_FA	Trayecto de medición	
6	TSA	S_TSA	Zona de confianza	
7	alfa 0	S_STA1	Ángulo inicial al medir en ángulo	
8	alfa 1	S_INCA	Ángulo de incremento al medir en ángulo	
9	X1	S_SETV0	Posición teórica de la 1.ª esfera del 1.er eje del plano (con G17 X) al medir 3 esferas	
10	Y1	S_SETV1	Posición teórica de la 1.ª esfera del 2.º eje del plano (con G17 Y) al medir 3 esferas	
11	Z1	S_SETV2	Posición teórica de la 1.ª esfera del 3.er eje del plano (con G17 Z) al medir 3 esferas	
12	X2	S_SETV3	Posición teórica de la 2.ª esfera del 1.er eje del plano al medir 3 esferas	
13	Y2	S_SETV4	Posición teórica de la 2.ª esfera del 2.º eje del plano al medir 3 esferas	
14	Z2	S_SETV5	Posición teórica de la 2.ª esfera del 3.er eje del plano al medir 3 esferas	
15	X3	S_SETV6	Posición teórica de la 3.ª esfera del 1.er eje del plano al medir 3 esferas	
16	Y3	S_SETV7	Posición teórica de la 3.ª esfera del 2.º eje del plano al medir 3 esferas	
17	Z3	S_SETV8	Posición teórica de la 3.ª esfera del 3.er eje del plano al medir 3 esferas	
18	TVL	S_TNVL	Valor límite de distorsión del triángulo (suma de las desviaciones) al medir 3 esferas [5]	
19	Mediciones	S_NMSP	Número de mediciones en la misma posición [2] (rango de valores 1 a 9)	
20		S_MCBIT	reservado	
21		_DMODE	Modo display	
			Valores:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
22		_AMODE	Modo alternativo	

1) Todos los valores predeterminados = 0 o están identificados como rango de valores "a" a "b"

2) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE

3) Posicionado intermedio con giro alrededor del ecuador de la esfera

4) Medición de 3 esferas: a todas las esferas se aplica el mismo diámetro teórico (_SETV).

5) Valor predeterminado para `S_TNVL=1.2`
Corrección en DO: En el DO solo se corrige si la distorsión determinada es inferior al valor límite `S_TNVL` .

6) Si no está configurado el DO "fino" en los DM, se corrige según el DO "grueso".

7) Con la variante de medida "Medir tres esferas" no es posible la corrección en un frame básico global (`S_KNUM` = 3001 a 3016), puesto que el frame no posee componente de rotación.

3.25.3.11 Parámetros de ciclo de medida 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)
```

Tabla 3-18 Parámetros de llamada CYCLE995 1)

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
1		S_MVAR	Variante de medida (predeterminado = 5)	
			Valo-res:	UNIDADES: Elemento de contorno 5 = geometría del cabezal (paralelismo con el eje de la herramienta)
				DECENAS: Repetición de la medida 1 = con repetición de la medida
				CENTENAS: sin objetivo de corrección 0 = solo medida
				MILLARES: Estrategia de medición 2 = medición bajo el ángulo, alinear el palpador en el sentido de conmutación
				DECENAS DE MILLAR: número de esferas que van a medirse 0 = medir una esfera
				CENTENAS DE MILLAR: número de puntos de medida 1 = 4 puntos de medida en la medición en ángulo (girando alrededor de la esfera)
				UNIDADES DE MILLÓN: determinación del diámetro teórico de la esfera 0 = sin determinación del diámetro teórico de la esfera 1 = determinación del diámetro teórico de la esfera
2.	Selección	S_KNUM	Objetivo de corrección 0 = 0	
3	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número de palpador) (rango de valores 1 a 40)	
4	DM	S_SETV	Diámetro de la esfera de calibración 4)	
5	DFA	S_FA	Trayecto de medición	
6	TSA	S_TSA	Zona de confianza 5)	

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
7	alfa 0	S_STA1	Ángulo inicial al medir en ángulo [3]	
8		S_INCA	Ángulo de incremento al medir en ángulo [2]	
9	DZ	S_DZ	Distancia 1.ª medición P1 a 2.ª medición P2 en el mango del palpador	
10		S_SETV0	Posición teórica de la esfera del 1.er eje del plano (con G17 X)[2]	
11		S_SETV1	Posición teórica de la esfera del 2.º eje del plano (con G17 Y)[2]	
12		S_SETV2	Posición teórica de la esfera del 3.er eje del plano (con G17 Z)[2]	
13	TUL	S_TUL	Límite superior de tolerancia de la desviación angular	
14	TZL	S_TZL	Zona de corrección cero [1, 4]	
15	Número	S_NMSP	Número de mediciones en la misma posición [2] (rango de valores 1 a 9)	
16		S_MCBIT	reservado [2]	
17		_DMODE	Modo display	
			Valo-res:	UNIDADES: Plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
18		_AMODE	Modo alternativo	
			Valo-res:	UNIDADES: Tolerancia dimensional sí/no 0 = no 1 = sí

todos los valores predeterminados = 0 o están identificados como rango de valores "a" a "b"

1) Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE.

2) Los parámetros no se usan actualmente y tampoco se muestran en la máscara de entrada.
El parámetro ángulo de incremento S_INCA tiene un valor fijo de 90 grados.

3) Rango de valores de ángulo inicial: -360 a +360 grados.

4) Con tolerancia dimensional sí:
Si los ángulos medidos son menores que el valor de la zona de corrección cero TZL, los parámetros de resultado para los ángulos (_OVR[2], _OVR[3]) y las diferencias (_OVR[7], _OVR[8]) se establecen a cero.
La visualización de TZL se produce a través del SD54760 $SNS_MEA_FUNCTION_MASK_PIECE general, bit 25 = 1.
(Habilitar selección de corrección cero al medir angularidad del cabezal)

5) El parámetro TSA se refiere a la 1.ª medición de la esfera de calibración.

3.25.3.12 Parámetros de ciclo de medida 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)
```

Tabla 3-19 Parámetros de llamada CYCLE996 [1)]

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
1		S_MVAR	Variante de medida (predeterminado = 1)	
			Valores:	UNIDADES: orden de medida 0 = calcular cinemática (selección con: Visualización de resultado, protocolo, modificación de juego de datos de orientación, con confirmación de usuario dado el caso), ver _AMODE 1 = 1.ª medición 2 = 2.ª medición 3 = 3.ª medición
				DECENAS: reservado 0 = 0
				CENTENAS: Variante de medida en 1.ª medición a 3.ª medición 0 = medición de la esfera de calibración en paralelo al eje 1 = medición de la esfera de calibración en ángulo, sin desplazamiento del cabezal [3)] 2 = medición de la esfera de calibración y desplazamiento del cabezal en el sentido de conmutación del palpador [3)] 3 = medición en paralelo al eje, con ángulo inicial [8)] 4 = medición en paralelo al eje, con ángulo inicial, desplazamiento del cabezal en el sentido de conmutación del palpador [8)]
				MILLARES: objetivo de corrección con cálculo de cinemática [4)] 0 = solo medida. Los juegos de datos de orientación se calculan pero no sufren modificaciones 1 = calcular juego de datos de orientación. Los juegos de datos de orientación se modifican tras la confirmación del usuario, dado el caso [4)]
				DECENAS DE MILLAR: eje de medición (eje giratorio 1 o 2) o cadena vectorial abierta o cerrada con cálculo de cinemática 0 = cadena vectorial cerrada (solo con cálculo de cinemática) 1 = eje giratorio 1 (solo con 1.ª a 3.ª medición) 2 = eje giratorio 2 (solo con 1.ª a 3.ª medición) [5)] 3 = cadena vectorial abierta (solo con cálculo de cinemática)
				CENTENAS DE MILLAR: normalización eje giratorio 1 con cálculo de cinemática 0 = sin normalización eje giratorio 1 1 = normalización en sentido 1.er eje del plano (con G17 X) 2 = normalización en sentido 2.º eje del plano (con G17 Y) 3 = normalización en sentido 3.er eje del plano (con G17 Z)
				UNIDADES DE MILLÓN: normalización eje giratorio 2 con cálculo de cinemática [5)] 0 = sin normalización eje giratorio 2 1 = normalización en sentido 1.er eje del plano (con G17 X) 2 = normalización en sentido 2.º eje del plano (con G17 Y) 3 = normalización en sentido 3.er eje del plano (con G17 Z)
				DECENAS DE MILLÓN: fichero para protocolos 0 = sin fichero para protocolos 1 = fichero para protocolos con los vectores calculados (Toolcarrier) y la 1.ª transformada de 5 ejes dinámica (TRAORI(1)), si está configurado en los DM.
2		S_TC	Número del juego de datos de orientación (ToolCarrier)	

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
3	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número del palpador) (predeterminado = 1)	
4		S_SETV	Diámetro de la esfera de calibración	
5	alfa 0	S_STA1	Ángulo inicial al medir en ángulo	
6	alfa 0	S_SETV0	Valor de posición eje giratorio 1 (con eje giratorio manual o semiautomático)	
7	alfa 1	S_SETV1	Valor de posición eje giratorio 2 (con eje giratorio manual o semiautomático) [6]	
8	XN	S_SETV2	Valor de posición para normalización eje giratorio 1	
9	XN	S_SETV3	Valor de posición para normalización eje giratorio 2 [6]	
10	delta	S_SETV4	Valor de tolerancia de los vectores de offset I1 a I4	
11	delta	S_SETV5	Valor de tolerancia de los vectores de eje giratorio V1 y V2	
12	TVL	S_TNVL	Valor límite del segmento angular del eje giratorio (rango de valores 1 a 60 grados), (predeterminado = 20) [7]	
13	DFA	S_FA	Trayecto de medición	
14	TSA	S_TSA	Zona de confianza	
15	Mediciones	S_NMSP	Número de mediciones en la misma posición [2] (predeterminado = 1)	
16		S_MCBIT	reservado	
17		_DMODE	Modo display	
			Valores:	UNIDADES: Plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
18		_AMODE	Modo alternativo	
			Valores:	UNIDADES: control de tolerancia sí/no 0 = no 1 = sí: evaluación de los valores de tolerancia de los vectores S_SETV4, S_SETV5
				DECENAS: confirmación del usuario al introducir los vectores calculados en el juego de datos de orientación [4] 0 = sí: el usuario debe confirmar la modificación 1 = no: los vectores calculados se introducen de inmediato (solo tiene efecto si los dígitos de CENTENAS y MILLARES = 0)
				CENTENAS: visualización de resultado de medición [5] 0 = no 1 = sí
				MILLARES: visualización de resultado de medición editable 0 = no 1 = sí, y editable (solo tiene efecto si el dígito de CENTENAS = 1)

[1] Todos los valores predeterminados = 0 o están identificados con predeterminado = x

[2] Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE.

[3] Esta variante permite, p. ej., medir con posiciones de 90 grados de la cinemática en la esfera de calibración sin que se choque con el vástago de fijación de la esfera de calibración. Puede especificarse un ángulo inicial S_STA1 (0 a 360 grados). El ángulo de incremento es igual a 90 grados al efectuar el giro alrededor de la esfera.
El dato de operador específico de canal SD55634 $SCS_MEA_FEED_PLANE_VALUE se utiliza como avance en la trayectoria circular.

[4] Antes de la introducción se efectúa una consulta al usuario con M0. Los vectores se introducen solamente con Marcha CN. Si se cancela con RESET el programa de medida, no se introduce ningún vector calculado. Los vectores solo se introducen si no se supera la tolerancia de los vectores de offset en el cálculo.

[5] Visualización de resultado de medición solo con variante de medida "Calcular cinemática". Si el resultado de la medición también debe mostrarse tras la 1.ª a 3.ª medición, hay que ajustar el dato de operador específico de canal SD55613 $SCS_MEA_RESULT_DISPLAY.

[6] Eje giratorio 2 solo con cinemáticas con dos ejes giratorios

[7] Valor límite segmento angular del eje giratorio. Rango de valores de S_TNVL entre 20 y 60 grados. Con valores de S_TNVL < 20 grados, puede haber inexactitudes debidas a imprecisiones de medición del orden de micras del palpador. Si se sobrepasa el valor límite, se emite el aviso de error 61430 con la indicación del valor mínimo para el límite.

[8] Desplazamiento del cabezal en el sentido de conmutación del palpador si SD54760, bit 17 = 1

3.25.3.13 Parámetros de ciclo de medida 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)
```

Tabla 3-20 Parámetros de llamada CYCLE9960 [1]

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
1	Selección	S_MVAR	Variante de medida (predeterminado = 1)	
			Valores:	UNIDADES: Variante de medida 0 = medir y calcular cinemática (selección con: visualización de resultados, protocolo, modificación del juego de datos de orientación (ver millares S_MVAR)) 1 = medir cabezal de referencia 2 = adaptar cabezal a cabezal de referencia 3 = medir y calcular nodos de interpolación (E996)
				DECENAS: reservado
				CENTENAS: Variante de medida en la medición de esfera 2 = medición de la esfera de calibración y desplazamiento del cabezal en el sentido de conmutación del palpador 4 = medición en paralelo al eje, con ángulo inicial, desplazamiento del cabezal en el sentido de conmutación del palpador [2]
				MILLARES: Objetivo de corrección con cálculo de cinemática [3] 0 = medir y calcular. Los juegos de datos se calculan y no sufren modificaciones 1 = los juegos de datos calculados se modifican, en caso dado, tras la confirmación del usuario 2 = la cinemática ya medida se calcula y, en caso dado, se modifica tras la confirmación del usuario
				DECENAS DE MILLAR: eje de medición (eje giratorio 1 o 2) 1 = medir y calcular todos los ejes giratorios existentes 4 = medir y calcular solo el eje giratorio 1 5 = medir y calcular solo el eje giratorio 2

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
2		S_TNAME	Nombre de la transformada (juego de datos de orientación o transformada sobre la base de cadenas cinemáticas)	
3	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número del palpador) (predeterminado = 1)	
4		S_SETV	Diámetro de la esfera de calibración	
5	alfa 1	S_SETV1	Ángulo inicial al medir en ángulo para el 1.er eje giratorio	
6			S_START_RA1 : ángulo inicial del 1.er eje giratorio	
7			S_END_RA1 : ángulo final del 1.er eje giratorio	
8			S_CMEA_RA1 : número de mediciones del 1.er eje giratorio, 3 para medir y calcular cinemática. Hasta 12 mediciones posibles para puntos de interpolación.	
9			S_POS_RA2 : posición del 2.º eje giratorio en la medición del 1.er eje giratorio [4]	
10	alfa 2	S_SETV2	Ángulo inicial al medir en ángulo para el 2.º eje giratorio [4]	
11			S_START_RA2 : ángulo inicial del 2.º eje giratorio [4]	
12			S_END_RA2 : ángulo final del 2.º eje giratorio [4]	
13			S_CMEA_RA2 : número de mediciones del 2.º eje giratorio, 3 para medir y calcular cinemática. Hasta 12 mediciones posibles para puntos de interpolación.[4]	
14			S_POS_RA1 : posición del 1.er eje giratorio en la medición del 2.º eje giratorio	
15	delta	S_SETV4	Valor de tolerancia de los vectores de offset	
16	DFA	S_FA	Trayecto de medición	
17	TSA	S_TSA	Zona de confianza	
18	Número	S_NMSP	Número de mediciones en el mismo lugar [5] (predeterminado = 1)	
19		_DMODE	Modo display	
			Valores:	UNIDADES: Plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
20		_AMODE	Modo alternativo	
			Valores:	UNIDADES: control de tolerancia sí/no 0 = no 1 = sí: evaluación de los valores de tolerancia de los vectores -> S_SETV4, S_SETV5
				DECENAS: calibrado automático 0 = sin calibrado automático 1 = calibrado automático
				CENTENAS: ángulo inicial automático 0 = se utiliza el ángulo inicial ajustado 1 = cálculo automatizado del ángulo inicial, en cada punto de medición
21		S_KNUM	Número de DO que se debe corregir en Medir cabezal de referencia	
			Valores	UNIDADES: la corrección se realiza siempre en el decalaje basto, el decalaje fino se borra
				DECENAS: 1..99 número de DO ajustables (1 = G54)
				MILLARES: 9 corrección de DO ajustables activos

[1] Todos los valores predeterminados = 0 o están identificados con predeterminado = xx

[2] Desplazamiento del cabezal en el sentido de conmutación del palpador si SD54760, bit 17 = 1

[3] Antes de la introducción se efectúa una consulta al usuario con M0. Los vectores se introducen solamente con Marcha CN. Si el programa de medida se cancela con RESET, no se introduce ningún vector calculado. Los vectores solo se introducen si no se supera la tolerancia de los vectores de offset en el cálculo.

[4] Eje giratorio 2 solo con cinemáticas con 2 ejes giratorios

[5] Indicación dependiente del dato general de operador SD54760 $SNS_MEA_FUNCTION_MASK_PIECE.

3.25.3.14 Parámetros de ciclo de medida 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)
```

Tabla 3-21 Parámetros de llamada CYCLE982 [1]

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
1		S_MVAR	Variante de medida	
			Valores:	UNIDADES: calibración/medición 0 = calibrar palpador de herramienta 1 = medición individual de herramienta [3] 2 = medición múltiple de herramienta, determinación de longitudes y radio de herramienta (con fresas)
				DECENAS: calibración o medición en MKS o WKS 0 = referido a la máquina [4] 1 = referido a la pieza
				CENTENAS: medición con y sin giro en fresas 0 = medición sin giro 1 = medición con giro
				MILLARES: objetivo de corrección en fresas 0 = determinar longitud o longitud y radio (ver 1.er dígito de S_MVAR) 1 = determinar radio si el 1.er dígito de S_MVAR = 1 2 = determinar longitud y radio (lado frontal) si el 1.er dígito de S_MVAR = 1 o 2 3 = fresa de plaquitas: determinar filo superior (lado posterior), longitud y radio [5]
				DECENAS DE MILLAR: posición de la fresa o de la broca 0 = posición axial de la fresa o la broca, radio en 2.º eje del plano (con G18 X) [7] 1 = posición radial de la fresa o la broca, radio en 1.er eje del plano (con G18 Z) [7]
				CENTENAS DE MILLAR: calibración o medición incremental 0 = sin indicación 1 = calibración o medición incremental
				UNIDADES DE MILLÓN: posicionamiento de cabezal en ángulo inicial S_STA1 (solo con la medición de fresas) 0 = sin posicionamiento del cabezal 1 = posicionamiento del cabezal en ángulo inicial S_STA1

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
2	Selección	S_KNUM	Variante de corrección [2]	
			Valores:	UNIDADES: corrección de herramienta 0 = sin indicación (corrección de herramienta en geometría) 1 = corrección de herramienta en desgaste
3	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número del palpador) (predeterminado = 1)	
4	X0	S_MA	Eje de medición	
			Valores:	1 = 1.er eje del plano (con G18 Z) 2 = 2.º eje del plano (con G18 X)
5	+-	S_MD	Sentido de medición	
			Valores:	0 = sin selección (el sentido de medición de determina a partir del valor real) 1 = positivo 2 = negativo
6	Z2	S_ID	Decalaje	
7	DFA	S_FA	Trayecto de medición	
8	TSA	S_TSA	Zona de confianza	
9	VMS	S_VMS	Velocidad de medición variable al calibrar [2]	
10	alfa1	S_STA1	Ángulo inicial al medir fresas	
11	alfa2	S_CORA	Ángulo de corrección al medir fresas con giro [8]	
12	TZL	S_TZL	Corrección cero al medir herramientas. S_TZL = 0 al calibrar.	
13	DIF	S_TDIF	Control de diferencia dimensional	
14	Mediciones	S_NMSP	Número de mediciones en la misma posición [2] (predeterminado = 1)	
15	EVN	S_EVNUM	Número para la memoria de valores medios empíricos [2], [9]	
16		S_MCBIT	reservado	
17		_DMODE	Modo display	
			Valores:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)
				DECENAS: posición de filo con fresas y herramientas de torneado (solo para visualización en las máscaras de entrada 1 a 9)
				CENTENAS: Tipo de herramienta 0 = herramienta de tornear 1 = fresa 2 = broca
				MILLARES: Estrategia de aproximación respecto al palpador de herramienta 0 = PLUS [X/Z]; X con posición de herramienta axial, Z con posición de herramienta radial 1 = MINUS [X/Z]; X con posición de herramienta axial, Z con posición de herramienta radial
18		_AMODE	Modo alternativo	
			Valores:	UNIDADES: reservado
				DECENAS: reservado

N.°	Parámetros de máscara	Parámetros de ciclo	Descripción	
				CENTENAS: reservado
				MILLARES: desplazamiento a la posición inicial tras proceso de medición para calibración y medición individual (ver S_MVAR, UNIDADES) 0 = la herramienta está desplazada la distancia DFA respecto al borde del palpador 1 = desplazamiento a la posición inicial

1) Todos los valores predeterminados = 0 o están identificados con predeterminado = x

2) Indicación dependiente del dato general de operador SD54762 _MEA_FUNCTION_MASK_TOOL

3) Medir herramienta de tornear, fresa o broca. Eje de medición en parámetro `S_MA`
Especificación con herramientas de tornear mediante posición de filo 1...8, con fresas mediante los dígitos de CENTENAS a MILLARES del parámetro `S_MVAR`.

4) La medición y la calibración tienen lugar en el sistema de coordenadas básico (MKS con transformación cinemática desactivada).

5) No para medición incremental.

6) Solo para medición múltiple `S_MVAR=x2x02` o `x3x02` (ejemplo fresa de plaquitas o fresa para ranuras)

7) Si el dato de operador específico de canal SD42950 $SC_TOOL_LENGTH_TYPE = 2, se efectúa la asignación de los componentes de longitud de herramienta como con herramientas de tornear

8) Solo en caso de medición con giro `S_MVAR=xx1x1`.

9) Cálculo de valor empírico
Rango de valores de la memoria de valores empíricos: 1 a 20 números (n) de la memoria de valores empíricos, ver dato de operador específico de canal SD55623 $SCS_MEA_EMPIRIC_VALUE[n-1].

3.25.3.15 Parámetros de ciclo de medida 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)
```

Tabla 3-22 Parámetros de llamada CYCLE971 [1)]

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
1		S_MVAR	Variante de medida	
			Valores:	UNIDADES: 0 = calibrar palpador de herramienta 1 = medir herramienta con cabezal parado (longitud o radio) 2 = medir herramienta con cabezal girando (longitud o radio), ver parámetros S_F1 a S_S4
				DECENAS: medición en MKS o WKS 0 = medición en MKS (referida a la máquina), medir herramienta o calibrar palpador de herramienta 1 = medición en WKS (referida a la pieza), medir herramienta o calibrar palpador de herramienta
				CENTENAS: probar dientes individualmente 0 = no 1 = sí
				MILLARES: 0 = 0
				DECENAS DE MILLAR: calibración o medición incremental 0 = sin indicación 1 = calibración o medición incremental
				CENTENAS DE MILLAR: calibrado automático del palpador de herramienta 0 = no calibrar automáticamente el palpador de herramienta 1 = calibrar automáticamente el palpador de herramienta
				UNIDADES DE MILLÓN: Calibración en el plano con giro del cabezal 0 = calibración en el plano sin giro del cabezal 1 = calibración en el plano con giro del cabezal
2	Selección	S_KNUM	Variante de corrección [2)]	
			Valores:	UNIDADES: corrección de herramienta 0 = sin indicación (corrección de herramienta en geometría) 1 = corrección de herramienta en desgaste
3	icono+ número	S_PRNUM	Número del campo de los parámetros de palpador (no número del palpador)	
4	X0	S_MA	Eje de medición, eje de desplazamiento [4)]	
			Valores:	UNIDADES: número del eje de medición 1 = 1.er eje del plano (con G17 X) 2 = 2.º eje del plano (con G17 Y) 3 = 3.er eje del plano (con G17 Z)
				DECENAS: 0 = 0
				CENTENAS: número del eje de desplazamiento 0 = sin eje de desplazamiento 1 = 1.er eje del plano (con G17 X) 2 = 2.º eje del plano (Y con G17)

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
5	+-	S_MD	Sentido de medición	
			Valo-res:	0 = sin selección (el sentido de medición de determina a partir del valor real) 1 = positivo 2 = negativo
6	V	S_ID	Decalaje	
			Valo-res:	0 = con herramientas sin decalaje >0 = • Calibrado: el decalaje afecta al 3.er eje del plano (con G17 Z) si el diámetro de la herramienta de calibrado es mayor que el diámetro superior del palpador. La herramienta se desplaza aquí desde el centro del palpador según el radio de herramienta menos el valor de S_ID. En S_MA se indica adicionalmente el eje de desplazamiento. • Medición: en herramientas con varios filos, se debe indicar el decalaje de la longitud de herramienta y el punto más alto del filo para la medición del radio, o bien el decalaje del radio de herramienta hasta el punto más alto del filo para la medición de la longitud de herramienta.
7	DFA	S_FA	Trayecto de medición	
8	TSA	S_TSA	Zona de confianza	
9	VMS	S_VMS	Velocidad de medición variable al calibrar [2)]	
10	TZL	S_TZL	Corrección cero (solo al medir la herramienta)	
11	DIF	S_TDIF	Control de diferencia dimensional al medir herramienta (S_MVAR=xx1 o S_MVAR=xx2)	
12	Mediciones	S_NMSP	Número de mediciones en la misma posición [2)]	
13	F1	S_F1	1.er avance para palpado con cabezal girando [2)]	
14	S1	S_S1	1.ª velocidad para palpado con cabezal girando [2)]	
15	F2	S_F2	2.º avance para palpado con cabezal girando [2)]	
16	S2	S_S2	2.ª velocidad para palpado con cabezal girando [2)]	
17	F3	S_F3	3.er avance para palpado con cabezal girando [2)]	
18	S3	S_S3	3.ª velocidad para palpado con cabezal girando [2)]	
19	EVN	S_EVNUM	Número para la memoria de valores empíricos [2)]	
20		S_MCBIT	Máscara de _CBIT o _CHBIT	
21		_DMODE	Modo display	
			Valo-res:	UNIDADES: plano de mecanizado G17/G18/G19 0 = compatibilidad, el plano activo antes de la llamada de ciclo permanece activo 1 = G17 (solo activo en el ciclo) 2 = G18 (solo activo en el ciclo) 3 = G19 (solo activo en el ciclo)

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
22		_AMODE	Modo alternativo	
			Valo-res:	UNIDADES: decalaje de herramienta al medir radio 1 = no 2 = sí
				DECENAS: sentido del decalaje de herramienta al medir radio en el 3.er eje del plano (con G17 Z) 1 = positivo 2 = negativo
				CENTENAS: decalaje de herramienta al medir longitud o al calibrar el palpador en el 3.er eje 0 = compatibilidad, auto 1 = no 2 = sí
				MILLARES: sentido del decalaje de herramienta al medir longitud en el eje de desplazamiento (ver S_MA, CENTENAS) 1 = positivo 2 = negativo

1) Todos los valores predeterminados = 0 o están identificados con predeterminado = x

2) Indicación dependiente del dato general de operador SD54762 MEA_FUNCTION_MASK_TOOL

3) Solo con corrección en herramienta y tolerancia dimensional "Sí"; de lo contrario, el parámetro = 0

4) Con medición automática (S_MVAR=1x00xx), sin visualización de eje de medición, eje de desplazamiento ⇒ S_MA=0.

3.25.3.16 Parámetros de ciclo de medida CYCLE150

```
PROC CYCLE150(INT S_PICT,INT S_PROT,STRING[160] S_PATH) SAVE
ACTBLOCNO DISPLOF
```

Tabla 3-23 Parámetros de llamada CYCLE150

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
1	Pantalla de resultados de medición	S_PICT	Selección de la visualización de resultados (predeterminado = 0)	
			Valo-res:	UNIDADES: 0 = pantalla de resultados de medición DESACTIVADA 1 = pantalla de resultados de medición ACTIVADA
				Decenas: selección de modo de visualización (valores como en SD55613) 1 = visualizar pantalla de resultados de medición - deselección automática tras 8 s 3 = visualizar pantalla de resultados de medición - confirmación con Marcha CN 4 = visualizar pantalla de resultados de medición - solo con las alarmas (61303 ... 61306)
2		S_PROT	Selección de documentación (protocolo) (predeterminado = 0)	

N.°	Parámetro de máscara	Parámetro de ciclo	Significado	
	Protocolo		Valo-res:	UNIDADES: selección protocolo des/con/última medición 0 = protocolo DESACTIVADO 1 = protocolo ACTIVADO 2 = documentar última medición
	Tipo de pro-tocolo			DECENAS: selección de tipo de protocolo 0 = protocolo estándar 1 = protocolo de usuario (de libre definición)
	Formato de protocolo			CENTENAS: selección de formato de protocolo 0 = formato de texto 1 = formato de tabla (para importación en Excel)
	Datos de protocolo			MILLARES: selección escribir nuevo/añadir 0 = nuevo 1 = añadir
	Almacena-miento protocolo			DECENAS DE MILLAR: selección de almacenamiento de protocolos 0 = como programa de pieza 1 = directorio
3		`S_PATH`	Ruta de fichero de protocolo conforme a la selección de almacenamiento de protocolos (indicación de ruta completa o solo nombre de fichero, p. ej., "//NC:/WKS.DIR/NAME.WPD o "MESSPROTOKOLL.TXT"	

Tablas

4

4.1 Instrucciones

Nota

Ciclos

La lista de instrucciones contiene todos los ciclos que pueden aparecer en el programa de CN (código G), es decir, que pueden programarse en el editor de programas mediante máscaras o que deben programarse sin ayuda para la programación durante el rectificado. Se omiten los ciclos que todavía existen en el controlador por motivos de compatibilidad, pero que ya no pueden editarse con el editor de programas de SINUMERIK Operate ("ciclos de compatibilidad").

Instrucciones A ... C

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
:	O	Número de secuencia principal CN, fin de marcas de salto, operador de concatenación		+		PM-NC
*	O	Operador de multiplicación		+		PM-NC
+	O	Operador de suma		+		PM-NC
-	O	Operador de resta		+		PM-NC
<	O	Operador de comparación, menor que		+		PM-NC
<<	O	Operador de concatenación para cadenas de caracteres		+		PM-NC
<=	O	Operador de comparación, menor o igual que		+		PM-NC
=	O	Operador de asignación		+		PM-NC
>=	O	Operador de comparación, mayor o igual que		+		PM-NC
/	O	Operador de división		+		PM-NC
/0 … … /7		La secuencia se omite (1er. nivel de omisión) ... ... La secuencia se omite (8.º nivel de omisión)		+		PM-NC
A	A	Nombre de eje	m/s	+		PM-NC
A2	A	Orient. herramienta: Ángulo euleriano o RPY	s	+		PM-NC
A3	A	Orient. herramienta: 1er componente del vector de dirección	s	+		PM-NC
A4	A	Orient. herramienta: 1er componente del vector normal a la superficie al inicio de la secuencia	s	+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
A5	A	Orient. herramienta: 1er componente del vector normal a la superficie al fin de la secuencia	s	+		PM-NC
A6	A	Orient. herramienta: 1er componentes del vector de dirección para el eje de giro del cono	s	+		PM-NC
A7	A	Orient. herramienta: 1er componente del vector para la orientación intermedia en la superficie envolvente del cono	s	+		PM-NC
ABS	F	Valor absoluto		+	+	PM-NC
AC	K	Acotado absoluto de coordenadas/posiciones	s	+		PM-NC
ACC	K	Influencia en la aceleración por eje actual	m	+	+	PM-NC
ACCLIMA	K	Influencia en la aceleración por eje máxima actual	m	+	+	PM-NC
ACN	K	Acotado absoluto para ejes giratorios, aproximación a la posición en sentido negativo	s	+		PM-NC
ACOS	F	Arcocoseno (función trigon.)		+	+	PM-NC
ACP	K	Acotado absoluto para ejes giratorios, aproximación a la posición en sentido positivo	s	+		PM-NC
ACTBLOCNO	P	Emisión del número de secuencia actual de una secuencia de alarma, incluso aunque la función "Suprimir la visualización de secuencia actual" (DISPLOF) esté activa.		+		PM-NC
ADDFRAME	F	Inclusión y, en su caso, activación de un frame medido		+	-	PM-NC, FM-B
ADIS	A	Distancia de matado de esquinas para funciones de contorneado G1, G2, G3, ...	m	+		PM-NC
ADISPOS	A	Distancia de matado de esquinas para desplazamiento rápido G0	m	+		PM-NC
ADISPOSA	P	Magnitud de la ventana de tolerancia para IPOBRKA	m	+	+	PM-NC
ALF	A	Ángulo de retirada rápida	m	+		PM-NC
AMIRROR	G	Simetría programable	s	+		PM-NC
AND	K	Y lógico		+		PM-NC
ANG	A	Ángulo de sucesión de contorno	s	+		PM-NC
AP	A	Ángulo polar	m/s	+		PM-NC
APR	K	Lectura/visualización de la protección de acceso		+		PM-NC
APRB	K	Lectura de derechos de acceso, BTSS		+		PM-NC
APRP	K	Lectura de derechos de acceso, programa de pieza		+		PM-NC
APW	K	Escritura de la protección de acceso		+		PM-NC
APWB	K	Escritura de derechos de acceso, BTSS		+		PM-NC
APWP	K	Escritura de derechos de acceso, programa de pieza		+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
APX	K	Definición de la protección de acceso para la ejecución del elemento de lenguaje indicado		+		PM-NC
AR	A	Ángulo en el vértice	m/s	+		PM-NC
AROT	G	Rotación programable	s	+		PM-NC
AROTS	G	Rotaciones de frames programables con ángulos espaciales	s	+		PM-NC
AS	K	Definición de macro		+		PM-NC
ASCALE	G	Escala programable	s	+		PM-NC
ASIN	F	Función de cálculo, arcoseno		+	+	PM-NC
ASPLINE	G	Spline de Akima	m	+		PM-NC
ATAN2	F	Arcotangente 2		+	+	PM-NC
ATOL	A	Tolerancia específica del eje para funciones de compresor, alisado de la orientación y tipos de matado de esquinas	m	+		PM-NC
ATRANS	G	Decalaje de origen programable aditivo	s	+		PM-NC
AUXFUDEL	P	Borrar función auxiliar específica de canal de la lista global		+	-	FM-B
AUXFUDELG	P	Borrar todas las funciones auxiliares de un grupo de funciones auxiliares específicas de canal de la lista global		+	-	FM-B
AUXFUMSEQ	P	Determinar orden de salida de funciones auxiliares M		+	-	FM-B
AUXFUSYNC	P	A partir de la lista global de funciones auxiliares, generar una secuencia de programa de pieza completa en forma de cadena para el SERUPRO-Ende-ASUP específico de canal		+	-	FM-B
AX	K	Identificador de eje variable	m/s	+		PM-NC
AXCTSWE	P	Rotación del contenedor de ejes		+	-	PM-NC
AXCTSWEC	P	Anular la habilitación para giro del contenedor de ejes		+	+	PM-NC
AXCTSWED	P	Girar contenedor de ejes (¡variante de comando para puesta en marcha!)		+	-	PM-NC
AXIS	K	Identificador de eje, dirección de eje		+		PM-NC
AXNAME	F	Conversión de la cadena de caracteres introducida en un identificador de eje		+	-	PM-NC
AXSTRING	F	Conversión de la cadena de caracteres en un número de cabezal		+	-	PM-NC
AXTOCHAN	P	Solicitar un eje para un determinado canal. Es posible desde el programa de CN y desde una acción síncrona.		+	+	PM-NC
AXTOSPI	F	Conversión del identificador de eje en un índice de cabezal		+	-	PM-NC
B	A	Nombre de eje	m/s	+		PM-NC
B2	A	Orient. herramienta: Ángulo euleriano o RPY	s	+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
B3	A	Orient. herramienta: componente del vector de dirección	s	+		PM-NC
B4	A	Orient. herramienta: 2.º componente del vector normal a la superficie al inicio de la secuencia	s	+		PM-NC
B5	A	Orient. herramienta: 2.º componente del vector normal a la superficie al fin de la secuencia	s	+		PM-NC
B6	A	Orient. herramienta: 2.º componentes del vector de dirección para el eje de giro del cono	s	+		PM-NC
B7	A	Orient. herramienta: 2.º componente del vector para la orientación intermedia en la superficie envolvente del cono	s	+		PM-NC
B_AND	O	Y bit a bit		+		PM-NC
B_OR	O	O bit a bit		+		PM-NC
B_NOT	O	Negación binaria		+		PM-NC
B_XOR	O	O exclusivo binario		+		PM-NC
BAUTO	G	Definición del primer segmento spline mediante los 3 puntos siguientes	m	+		PM-NC
BLOCK	K	Define junto con la palabra reservada TO la parte ejecutable del programa en una ejecución indirecta del subprograma		+		PM-NC
BLSYNC	K	La ejecución de la rutina de interrupción no debe comenzar hasta el siguiente cambio de secuencia		+		PM-NC
BNAT [6]	G	Transición natural a la primera secuencia spline	m	+		PM-NC
BOOL	K	Tipo de dato: Valor lógico TRUE/FALSE o bien 1/0		+		PM-NC
BOUND	F	Comprueba si el valor se encuentra dentro del margen de valores definido. En caso de igualdad se devuelve el valor de comprobación.		+	+	PM-NC
BRISK [6]	G	Aceleración de contorneado de forma escalonada	m	+		PM-NC
BRISKA	P	Activar la aceleración de contorneado en escalón para los ejes programados		+	-	PM-NC
BSPLINE	G	Spline B	m	+		PM-NC
BTAN	G	Transición tangencial a la primera secuencia spline	m	+		PM-NC
C	A	Nombre de eje	m/s	+		PM-NC
C2	A	Orient. herramienta: Ángulo euleriano o RPY	s	+		PM-NC
C3	A	Orient. herramienta: 3er componente del vector de dirección	s	+		PM-NC
C4	A	Orient. herramienta: 3er componente del vector normal a la superficie al inicio de la secuencia	s	+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
C5	A	Orient. herramienta: 3er componente del vector normal a la superficie al fin de la secuencia	s	+		PM-NC
C6	A	Orient. herramienta: 3er componentes del vector de dirección para el eje de giro del cono	s	+		PM-NC
C7	A	Orient. herramienta: 3er componente del vector para la orientación intermedia en la superficie envolvente del cono	s	+		PM-NC
CAC	K	Desplazamiento a una posición absoluta		+		PM-NC
CACN	K	Se efectúa una aproximación absoluta en sentido negativo hasta el valor consignado en la tabla		+		PM-NC
CACP	K	Se efectúa una aproximación absoluta en sentido positivo hasta el valor consignado en la tabla		+		PM-NC
CADAPTOF	P	Desactivar adaptación de carga		+	-	PM-NC
CADAPTON	P	Activar adaptación de carga		+	-	PM-NC
CALCDAT	F	Calcula el radio y el centro de una circunferencia a partir de 3 o 4 puntos		+	-	PM-NC
CALCFIR	P	Adaptación del filtro antitirones FIR al modo dinámico		+	-	PM-NC
CALCPOSI	F	Comprobación con respecto a vulneraciones de zonas protegidas, limitación del campo de trabajo y límites de software		+	-	PM-NC
CALL	K	Llamada indirecta de subprograma		+		PM-NC
CALLPATH	P	Ruta de búsqueda programable en llamadas de subprograma		+	-	PM-NC
CANCEL	P	Interrumpir la acción síncrona modal		+	-	FM-SA
CANCELSUB	P	Cancelar nivel de subprograma		+	+	FM-SA
CASE	K	Bifurcación de programa condicionada		+		PM-NC
CDC	K	Desplazamiento directo a una posición		+		PM-NC
CDOF [6]	G	Desactivar la vigilancia de colisiones	m	+		PM-NC
CDOF2	G	Desactivar la vigilancia de colisiones, en fresado periférico en 3D	m	+		PM-NC
CDON	G	Vigilancia de colisiones	m	+		PM-NC
CFC [6]	G	Avance constante en el contorno	m	+		PM-NC
CFIN	G	Avance constante solo con curvatura interna, no con curvatura externa	m	+		PM-NC
CFINE	F	Asignación del decalaje fino a una variable FRAME		+	-	PM-NC
CFTCP	G	Avance constante en el punto de referencia del filo de herramienta, trayectoria del centro	m	+		PM-NC
CHAN	K	Especificación del ámbito de validez de datos		+		PM-NC
CHANDATA	P	Ajustar el número de canal para accesos a datos de canal		+	-	PM-NC
CHAR	K	Tipo de dato: Caracteres ASCII		+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
CHF	A	Chaflán; valor = longitud del chaflán	s	+		PM-NC
CHKDM	F	Comprobación de la univocidad dentro de un almacén		+	-	FM-TM
CHKDNO	F	Prueba de unicidad de los números D		+	-	PM-NC
CHR	A	Chaflán; valor = longitud del chaflán en el sentido de desplazamiento		+		PM-NC
CIC	K	Desplazamiento a una posición incremental		+		PM-NC
CIP	G	Interpolación circular a través de punto intermedio	m	+		PM-NC
CLEARM	P	Resetear una/varias metas para la coordinación de canales		+	+	PM-NC
CLRINT	P	Cancelar interrupción		+	-	PM-NC
CMIRROR	F	Simetría respecto a un eje del sistema de coordenadas.		+	-	PM-NC
COARSEA	K	Fin de movimiento al alcanzar la "Parada precisa BASTA"	m	+		PM-NC
COLLPAIR	F	Comprobar la pertenencia a un par de colisión		+		PM-NC
COMPCAD	G	Activar la función de compresor COMPCAD	m	+		PM-NC
COMPCURV	G	Activar la función de compresor COMPCURV	m	+		PM-NC
COMPLETE		Instrucción de control para lectura y escritura de datos		+		PM-NC
COMPOF [6]	G	Desactivar la compresión de secuencias CN	m	+		PM-NC
COMPON	G	Activar la función de compresor COMPON	m	+		PM-NC
COMPSURF	G	Activar la función de compresor COMPSURF	m	+		PM-NC
CONTDCON	P	Codificación de contornos en forma de tabla		+	-	PM-NC
CONTPRON	P	Activar preparación de referencia		+	-	PM-NC
CORROF	P	Todos los movimientos superpuestos activos se cancelan.		+	-	PM-NC
CORRTC	F	Modificar los vectores de offset o los vectores de dirección de los portaherramientas orientables después de la medición de la máquina.		+	-	PM-NC
CORRTRAFO	F	Modificar los vectores desplazados o los vectores de dirección de los ejes de orientación en el modelo cinemático de la máquina		+	-	PM-NC
COS	F	Coseno (función trigon.)		+	+	PM-NC
COUPDEF	P	Definición del conjunto de reductores electrónicos/conjunto de cabezales síncronos		+	-	PM-NC
COUPDEL	P	Borrar conjunto de reductores electrónicos		+	-	PM-NC
COUPOF	P	Desactivar conjunto de reductores electrónicos/par de cabezales síncronos		+	-	PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
COUPOFS	P	Desactivación de un conjunto de reductores electrónicos/par de cabezales síncronos con parada del cabezal esclavo		+	-	PM-NC
COUPON	P	Activar conjunto de reductores electrónicos/ par de cabezales síncronos		+	-	PM-NC
COUPONC	P	Activación de un conjunto de reductores electrónicos/par de cabezales síncronos con aplicación de la programación anterior		+	-	PM-NC
COUPRES	P	Reseteo de un conjunto de reductores electrónicos		+	-	PM-NC
CP [6]	G	Desplazamiento de trayectoria	m	+		PM-NC
CPBC	K	Acoplamiento genérico: Criterio de cambio de secuencia		+	+	FM-A
CPDEF	K	Acoplamiento genérico: Creación de un módulo de acoplamiento		+	+	FM-A
CPDEL	K	Acoplamiento genérico: Borrado de un módulo de acoplamiento		+	+	FM-A
CPFMOF	K	Acoplamiento genérico: Comportamiento del eje esclavo con desactivación completa		+	+	FM-A
CPFMON	K	Acoplamiento genérico: Comportamiento del eje esclavo en la activación		+	+	FM-A
CPFMSON	K	Acoplamiento genérico: Modo de sincronización		+	+	FM-A
CPFPOS	K	Acoplamiento genérico: posición síncrona del eje esclavo		+	+	FM-A
CPFRS	K	Acoplamiento genérico: sistema de referencia de coordenadas		+	+	FM-A
CPLA	K	Acoplamiento genérico: definición de un eje maestro		+	-	FM-A
CPLCTID	K	Acoplamiento genérico: número de la tabla de levas		+	+	FM-A
CPLDEF	K	Acoplamiento genérico: definición de un eje maestro y creación de un módulo de acoplamiento		+	+	FM-A
CPLDEL	K	Acoplamiento genérico: borrado de un eje maestro de un módulo de acoplamiento		+	+	FM-A
CPLDEN	K	Acoplamiento genérico: denominador del factor de acoplamiento		+	+	FM-A
CPLINSC	K	Acoplamiento genérico: Factor de escala para el valor de entrada de un eje maestro		+	+	FM-A
CPLINTR	K	Acoplamiento genérico: valor de decalaje para el valor de entrada de un eje maestro		+	+	FM-A
CPLNUM	K	Acoplamiento genérico: numerador del factor de acoplamiento		+	+	FM-A
CPLOF	K	Acoplamiento genérico: desactivación de un eje maestro de un módulo de acoplamiento		+	+	FM-A

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
CPLON	K	Acoplamiento genérico: activación de un eje maestro de un módulo de acoplamiento		+	+	FM-A
CPLOUTSC	K	Acoplamiento genérico: factor de escala del valor de salida de un acoplamiento		+	+	FM-A
CPLOUTTR	K	Acoplamiento genérico: valor de decalaje para el valor de salida de un acoplamiento		+	+	FM-A
CPLPOS	K	Acoplamiento genérico: posición síncrona del eje maestro		+	+	FM-A
CPLSETVAL	K	Acoplamiento genérico: referencia de acoplamiento		+	+	FM-A
CPMALARM	K	Acoplamiento genérico: supresión de salidas de alarma especiales referidas a acoplamientos		+	+	FM-A
CPMBRAKE	K	Acoplamiento genérico: comportamiento del eje esclavo con determinados comandos y señales de parada		+	-	FM-A
CPMPRT	K	Acoplamiento genérico: Comportamiento de acoplamiento al inicio de programa de pieza con búsqueda de secuencia vía test de programa		+	+	FM-A
CPMRESET	K	Acoplamiento genérico: comportamiento de acoplamiento con RESET		+	+	FM-A
CPMSTART	K	Acoplamiento genérico: comportamiento de acoplamiento con inicio de programa de pieza		+	+	FM-A
CPMVDI	K	Acoplamiento genérico: comportamiento del eje esclavo con determinadas señales de interfaz CN/PLC		+	+	FM-A
CPOF	K	Acoplamiento genérico: desactivación de un módulo de acoplamiento		+	+	FM-A
CPON	K	Acoplamiento genérico: activación de un módulo de acoplamiento		+	+	FM-A
CPRECOF [6]	G	Desactivar Precisión de contorno programable	m	+		PM-NC
CPRECON	G	Activar Precisión de contorno programable	m	+		PM-NC
CPRES	K	Acoplamiento genérico: activa los datos configurados del acoplamiento de cabezales síncronos		+	-	FM-A
CPROT	P	Activar/desactivar las zonas protegidas específicas del canal		+	-	PM-NC
CPROTDEF	P	Definición de una zona protegida específica del canal		+	-	PM-NC
CPSETTYPE	K	Acoplamiento genérico: tipo de acoplamiento		+	+	FM-A
CPSYNCOP	K	Acoplamiento genérico: umbral para la marcha síncrona de posición "basta"		+	+	FM-A
CPSYNCOP2	K	Acoplamiento genérico: umbral para la marcha síncrona de posición "basta" 2		+	+	FM-A

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
CPSYNCOV	K	Acoplamiento genérico: umbral para la marcha síncrona de velocidad "basta"		+	+	FM-A
CPSYNFIP	K	Acoplamiento genérico: umbral para la marcha síncrona de posición "fina"		+	+	FM-A
CPSYNFIP2	K	Acoplamiento genérico: umbral para la marcha síncrona de posición "fina" 2		+	+	FM-A
CPSYNFIV	K	Acoplamiento genérico: umbral para la marcha síncrona de velocidad "fina"		+	+	FM-A
CR	A	Radio del círculo	s	+		PM-NC
CROT	F	Giro del sistema de coordenadas actual		+	-	PM-NC
CROTS	F	Rotaciones de frames programables con ángulos espaciales (rotación en los ejes indicados)	s	+	-	PM-NC
CRPL	F	Rotación de frame en un plano cualquiera		+	-	FM-B
CSCALE	F	Factor de escala para varios ejes		+	-	PM-NC
CSPLINE	F	Spline cúbico	m	+		PM-NC
CT	G	Arco con transición tangencial	m	+		PM-NC
CTAB	F	Averigua posición del eje esclavo a partir de la posición del eje maestro de la tabla de levas		+	+	PM-NC
CTABDEF	P	Activar la definición de tablas		+	-	PM-NC
CTABDEL	P	Borrar tabla de levas		+	-	PM-NC
CTABEND	P	Desactivar la definición de tablas		+	-	PM-NC
CTABEXISTS	F	Comprueba la tabla de levas con el número n		+	+	PM-NC
CTABFNO	F	Número de tablas de levas todavía posibles en la memoria		+	+	PM-NC
CTABFPOL	F	Número de polinomios todavía posibles en la memoria		+	+	PM-NC
CTABFSEG	F	Número de segmentos de curvas todavía posibles en la memoria		+	+	PM-NC
CTABID	F	Indica el número de la n.ª tabla de levas		+	+	PM-NC
CTABINV	F	Averigua posición del eje maestro a partir de la posición del eje esclavo de la tabla de levas		+	+	PM-NC
CTABISLOCK	F	Devuelve el estado de bloqueo de la tabla de levas con el número n		+	+	PM-NC
CTABLOCK	P	Borrado y sobrescritura, bloquear		+	+	PM-NC
CTABMEMTYP	F	Devuelve la memoria en la cual se ha creado la tabla de levas con el número n.		+	+	PM-NC
CTABMPOL	F	Número máximo de polinomios posibles en la memoria		+	+	PM-NC
CTABMSEG	F	Número máximo de segmentos de curvas posibles en la memoria		+	+	PM-NC
CTABNO	F	Número de tablas de levas definidas en la memoria SRAM o DRAM		+	+	FM-A
CTABNOMEM	F	Número de tablas de levas definidas en la memoria SRAM o DRAM		+	+	PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
CTABPERIOD	F	Devuelve la periodicidad de la tabla de levas con el número n		+	+	PM-NC
CTABPOL	F	Número de polinomios ya utilizados en la memoria		+	+	PM-NC
CTABPOLID	F	Número de polinomios de leva utilizados por la tabla de levas con el número n		+	+	PM-NC
CTABSEG	F	Número de segmentos de curva ya utilizados en la memoria		+	+	PM-NC
CTABSEGID	F	Número de segmentos de curva utilizados por la tabla de levas con el número n		+	+	PM-NC
CTABSEV	F	Suministra el valor final del eje esclavo de un segmento de la tabla de levas		+	+	PM-NC
CTABSSV	F	Suministra el valor inicial del eje esclavo de un segmento de la tabla de levas		+	+	PM-NC
CTABTEP	F	Suministra el valor del eje maestro al final de la tabla de levas		+	+	PM-NC
CTABTEV	F	Suministra el valor del eje esclavo al final de la tabla de levas		+	+	PM-NC
CTABTMAX	F	Suministra el valor máximo del eje esclavo de la tabla de levas		+	+	PM-NC
CTABTMIN	F	Suministra el valor mínimo del eje esclavo de la tabla de levas		+	+	PM-NC
CTABTSP	F	Suministra el valor del eje maestro al inicio de la tabla de levas		+	+	PM-NC
CTABTSV	F	Suministra el valor del eje esclavo al inicio de la tabla de levas		+	+	PM-NC
CTABUNLOCK	P	Cancelación del bloqueo de borrado y sobrescritura		+	+	PM-NC
CTOL	A	Tolerancia de contorno para funciones de compresor, alisado de la orientación y tipos de matado de esquinas	m	+	-	PM-NC
CTOLG0	A	Tolerancia de contorno con desplazamientos en rápido	m	+	-	PM-NC
CTRANS	F	Decalaje de origen para varios ejes		+	-	PM-NC
CUT2D [6]	G	WRK en 2D	m	+		PM-NC
CUT2DD	G	2½-D-WRK referida a una herramienta diferencial	m	+		PM-NC
CUT2DF	G	WRK en 2D, relativa al frame actual (plano inclinado)	m	+		PM-NC
CUT2DFD	G	2½-D-WRK referida a una herramienta diferencial, relativa al frame actual (plano inclinado)	m	+		PM-NC
CUT3DC	G	3D-WRK para fresado periférico	m	+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
CUT3DCC	G	3D-WRK para el fresado periférico considerando una superficie de limitación con corrección de radio 3D: Contorno en la superficie de mecanizado	m	+		PM-NC
CUT3DCCD	G	3D-WRK para fresado periférico considerando una superficie de limitación con herramienta diferencial en la trayectoria del centro de la herramienta: Penetración en la superficie de limitación	m	+		PM-NC
CUT3DCD	G	3D-WRK referida a una herramienta diferencial para fresado periférico	m	+		PM-NC
CUT3DF	G	3D-WRK para fresado frontal con cambio de orientación	m	+		PM-NC
CUT3DFD	G	3D-WRK referida a una herramienta diferencial para fresado frontal con cambio de orientación	m	+		PM-NC
CUT3DFF	G	3D-WRK para fresado frontal con orientación constante. La orientación de la herramienta es la dirección definida con G17 - G19 y, dado el caso, girada por un frame.	m	+		PM-NC
CUT3DFS	G	3D-WRK para fresado frontal con orientación constante. La orientación de herramienta se define con G17 - G19 y no depende de los frames.	m	+		PM-NC
CUTCONOF [6]	G	Desactivar la corrección de radio de herramienta	m	+		PM-NC
CUTCONON	G	Activar la corrección de radio de herramienta	m	+		PM-NC
CUTMOD	A	Activación de la modificación de los datos de corrección en herramientas girables (asociado a portaherramientas orientables)	m	+		PM-NC
CUTMODK	A	Activación de la modificación de los datos de corrección en herramientas girables (asociado a transformaciones de orientación definidas mediante cadenas cinemáticas)	m	+		PM-NC
CYCLE60	C (T)	Ciclo de grabado		+		PM-NC
CYCLE61	C (T)	Planeado		+		PM-NC
CYCLE62	C (T)	Llamada de contorno		+		PM-NC
CYCLE63	C (T)	Fresado de caja de contorno		+		PM-NC
CYCLE64	C (T)	Pretaladrado de caja de contorno		+		PM-NC
CYCLE70	C (T)	Fresado de roscas		+		PM-NC
CYCLE72	C (T)	Fresado en contorneado		+		PM-NC
CYCLE76	C (T)	Fresado de salientes rectangulares		+		PM-NC
CYCLE77	C (T)	Fresado de salientes circulares		+		PM-NC
CYCLE78	C (T)	Fresado de roscas		+		PM-NC
CYCLE79	C (T)	Poliedro		+		PM-NC
CYCLE81	C (T)	Taladrado, centrado (punteado)		+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
CYCLE82	C (T)	Taladrado, avellanado		+		PM-NC
CYCLE83	C (T)	Taladrado profundo		+		PM-NC
CYCLE84	C (T)	Roscado con macho sin mandril de compensación		+		PM-NC
CYCLE85	C (T)	Escariado		+		PM-NC
CYCLE86	C (T)	Mandrinado		+		PM-NC
CYCLE92	C (T)	Tronzado		+		PM-NC
CYCLE95	C (T)	Mecanizado de contorno		+		PM-NC
CYCLE98	C (T)	Cadena de roscas		+		PM-NC
CYCLE99	C (T)	Roscado		+		PM-NC
CYCLE116	C (M)	Cálculo de centro y radio de una circunferencia		+		PM-MC
CYCLE119	C (M)	Determinación de la posición en el espacio		+		PM-MC
CYCLE150	C (M)	Ver/documentar resultados de medición		+		PM-MC
CYCLE435	C (T)	Calcular posición del diamante		+		PM-NC
CYCLE495	C (T)	Perfilar		+		PM-NC
CYCLE750	C (A)	Ciclo de trabajo interno para CYCLE751 ... CYCLE759 (contiene el comando MMC encargado de la llamada de la función)		-		FM-A
CYCLE751	C (A)	Abrir/ejecutar/cerrar la sesión de optimización		M		FM-A
CYCLE752	C (A)	Agregar un eje a la sesión de optimización		M		FM-A
CYCLE753	C (A)	Seleccionar el modo de optimización		M		FM-A
CYCLE754	C (A)	Añadir/eliminar juego de datos		M		FM-A
CYCLE755	C (A)	Guardar/restablecer juego de datos		M		FM-A
CYCLE756	C (A)	Activar resultados de la optimización		M		FM-A
CYCLE757	C (A)	Guardar los datos de optimización		M		FM-A
CYCLE758	C (A)	Modificar valor de parámetro		M		FM-A
CYCLE759	C (A)	Leer valor de parámetro		M		FM-A
CYCLE782	C (T)	Adaptar a la carga		+		PM-NC
CYCLE800	C (T)	Orientación		+		PM-NC
CYCLE801	C (T)	Rejilla o marco		+		PM-NC
CYCLE802	C (T)	Cualquier posición		+		PM-NC
CYCLE830	C (T)	Taladrado profundo 2		+		PM-NC
CYCLE832	C (T)	High Speed Settings		+		PM-NC
CYCLE840	C (T)	Roscado con macho con mandril de compensación		+		PM-NC
CYCLE899	C (T)	Fresado de ranura abierta		+		PM-NC
CYCLE930	C (T)	Ranurado		+		PM-NC
CYCLE940	C (T)	Formas de garganta		+		PM-NC
CYCLE951	C (T)	Desbaste		+		PM-NC
CYCLE952	C (T)	Ranurado de contorno		+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
CYCLE961	C (M)	Determinar la posición de una esquina de una pieza (interior o exterior) y utilizarla como decalaje de origen		+		PM-MC
CYCLE971	C (M)	Calibrar el palpador de herramienta, medir la longitud o el radio de la herramienta (solo para la tecnología de fresado)		+		PM-MC
CYCLE973	C (M)	Calibrar un palpador de pieza en una superficie de la pieza o en una ranura (solo para la tecnología de torneado)		+		PM-MC
CYCLE974	C (M)	Determinar el origen de la pieza en el eje de medición seleccionado o una corrección de herramienta con la medición en 1 punto (solo para la tecnología de torneado)		+		PM-MC
CYCLE976	C (M)	Calibrar un palpador de pieza en un anillo o esfera de calibración, ya sea íntegramente en el plano de trabajo o bien en un borde, para un eje y sentido determinados		+		PM-MC
CYCLE977	C (M)	Determinar el centro del plano y también la anchura o el diámetro		+		PM-MC
CYCLE978	C (M)	Medir la posición de un borde en el sistema de coordenadas de pieza		+		PM-MC
CYCLE979	C (M)	Determinar el centro del plano y medir el radio de segmentos circulares		+		PM-MC
CYCLE982	C (M)	Calibrar el palpador de herramienta, medir herramientas de torneado, taladrado y fresado (solo para la tecnología de torneado)		+		PM-MC
CYCLE994	C (M)	Determinar el origen de la pieza en el eje de medición seleccionado con la medición en 2 puntos (solo para la tecnología de torneado)		+		PM-MC
CYCLE995	C (M)	Medir la angularidad del cabezal en una máquina herramienta		+		PM-MC
CYCLE996	C (M)	Determinar datos relevantes para transformadas cinemáticas con ejes giratorios		+		PM-MC
CYCLE997	C (M)	Determinar el centro y el diámetro de una esfera, medir los centros de tres esferas distribuidas		+		PM-MC
CYCLE998	C (M)	Determinar la posición angular de una superficie (plano) referida al plano de trabajo, así como el ángulo de bordes en el sistema de coordenadas de pieza		+		PM-MC
CYCLE4071	C (T)	Rectificado longitudinal con penetración en el punto de inversión		+		PM-NC
CYCLE4072	C (T)	Rectificado longitudinal con penetración en el punto de inversión y señal de cancelación		+		PM-NC
CYCLE4073	C (T)	Rectificado longitudinal con penetración continua		+		PM-NC
CYCLE4074	C (T)	Rectificado longitudinal con penetración continua y señal de cancelación		+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
CYCLE4075	C (T)	Rectificado plano con penetración en el punto de inversión		+		PM-NC
CYCLE4077	C (T)	Rectificado plano con penetración en el punto de inversión y señal de cancelación		+		PM-NC
CYCLE4078	C (T)	Rectificado plano con penetración continua		+		PM-NC
CYCLE4079	C (T)	Rectificado plano con penetración intermitente		+		PM-NC
CYCLE9960	C (M)	Medir cinemática completa		+		PM-MC

Instrucciones D ... F

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
D	A	Número de corrección de herramienta		+		PM-NC
D0	A	Con D0 se invalidan las correcciones para la herramienta		+		PM-NC
DAC	K	Programación por diámetros específica del eje, por secuencias, absoluta	s	+		PM-NC
DC	K	Acotado absoluto para ejes giratorios, aproximación directa a la posición	s	+		PM-NC
DCI	K	Asignación de clase de datos I (= individual) (solo SINUMERIK 828D)		+		PM-NC
DCM	K	Asignación de clase de datos M (= manufacturero) (solo SINUMERIK 828D)		+		PM-NC
DCU	K	Asignación de clase de datos U (= usuario) (solo SINUMERIK 828D)		+		PM-NC
DEF	K	Definición de variables		+		PM-NC
DEFAULT	K	Rama de la bifurcación CASE		+		PM-NC
DEFINE	K	Palabra reservada para definiciones de macros		+		PM-NC
DELAYFSTOF	P	Definir el fin de un área Stop-Delay	m	+	-	PM-NC
DELAYFSTON	P	Definir el inicio de un área Stop-Delay	m	+	-	PM-NC
DELDL	F	Borrar correcciones aditivas		+	-	PM-NC
DELDTG	P	Borrado de trayecto residual		-	+	FM-SA
DELETE	P	Borrar el fichero indicado. El nombre del fichero se puede indicar con la ruta y la identificación del fichero.		+	-	PM-NC
DELMLOWNER	F	Borrar puesto de almacén de propietario de la herramienta		+	-	FM-TM
DELMLRES	F	Borrar reserva de puesto de almacén		+	-	FM-TM
DELMT	P	Borrar Multitool		+	-	FM-TM

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1][2][3][4][5] Ver explicación en la leyenda (Página 1237).						
DELOBJ	F	Eliminación de elementos de cadenas cinemáticas, zonas protegidas, elementos de zonas protegidas, pares de colisión y datos de transformadas		+		PM-NC
DELT	P	Borrar herramienta		+	-	FM-TM
DELTC	P	Borrar juego de datos de portaherramientas		+	-	FM-TM
DELTOOLENV	F	Borrar bloques de datos para la descripción de entornos de herramienta		+	-	PM-NC
DIACYCOFA	K	Programación por diámetro específica del eje, modal: DES en ciclos	m	+		FM-A
DIAM90	G	Programación por diámetro para G90, programación por radio para G91	m	+		PM-NC
DIAM90A	K	Programación por diámetro específica del eje, modal, para G90 y AC, programación por radio para G91 e IC	m	+		PM-NC
DIAMCHAN	K	Adopción de todos los ejes del DM Funciones de eje en el estado de canal de la programación por diámetro		+		PM-NC
DIAMCHANA	K	Adopción del estado del canal de la programación por diámetro		+		PM-NC
DIAMCYCOF	G	Programación por diámetro específica del canal: DES en ciclos	m	+		FM-A
DIAMOF [6]	G	Programación por diámetro: DES Ajuste inicial: ver fabricante de la máquina	m	+		PM-NC
DIAMOFA	K	Programación por diámetro específica del eje, modal: DES Ajuste inicial: ver fabricante de la máquina	m	+		PM-NC
DIAMON	G	Programación por diámetro: CON	m	+		PM-NC
DIAMONA	K	Programación por diámetro específica del eje, modal: CON Habilitación: ver fabricante de la máquina	m	+		PM-NC
DIC	K	Programación por diámetro específica del eje, por secuencias, incremental	s	+		PM-NC
DILF	A	Distancia de retirada (longitud)	m	+		PM-NC
DISABLE	P	Desactivada la interrupción		+	-	PM-NC
DISC	A	Rebase circunferencia de transición; corrección de radio de herramienta	m	+		PM-NC
DISCL	A	Distancia entre el punto final del movimiento de penetración rápido y el plano de mecanizado		+		PM-NC
DISPLOF	PA	Suprimir la visualización de secuencia actual		+		PM-NC
DISPLON	PA	Anular la supresión de la visualización de secuencia actual		+		PM-NC
DISPR	A	Diferencia trayectoria reposicionamiento	s	+		PM-NC
DISR	A	Distancia reposicionamiento	s	+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1][2][3][4][5] Ver explicación en la leyenda (Página 1237).						
DISRP	A	Distancia entre el plano de retirada y el plano de mecanizado con aproximación y retirada suaves		+		PM-NC
DITE	A	Trayecto de salida de rosca	m	+		PM-NC
DITS	A	Trayecto de entrada en rosca	m	+		PM-NC
DIV	K	División entera		+		PM-NC
DL	A	Seleccionar una corrección de herramienta aditiva dependiente de la posición (DL, suma de correcciones de preparación)	m	+		PM-NC
DO	K	Acción síncrona: Disparo de acciones al cumplirse la condición		-	+	FM-SA
DRFOF	P	Desactivación de los desplazamientos con volante (DRF)	m	+	-	PM-NC
DRIVE	G	Aceleración de contorneado dependiente de la velocidad	m	+		PM-NC
DRIVEA	P	Activar aceleración con perfil discontinuo por tramos para los ejes programados		+	-	PM-NC
DYNFINISH	G	Dinámica para acabado	m	+		PM-NC
DYNNORM [6]	G	Dinámica normal	m	+		PM-NC
DYNPOS	G	Dinámica para el modo de posicionado, roscado de taladros	m	+		PM-NC
DYNPREC	G	Dinámica para acabado fino	m	+		PM-NC
DYNROUGH	G	Dinámica para desbaste	m	+		PM-NC
DYNSEMIFIN	G	Dinámica para preacabado	m	+		PM-NC
DZERO	P	Marca todos los números D de la unidad TO como no válidos		+	-	PM-NC
EAUTO	G	Definición del último segmento spline mediante los 3 últimos puntos	m	+		PM-NC
EGDEF	P	Definición de un reductor electrónico		+	-	PM-NC
EGDEL	P	Borrar la definición de acoplamiento para el eje esclavo		+	-	PM-NC
EGOFC	P	Desactivación continua del reductor electrónico		+	-	PM-NC
EGOFS	P	Desactivación selectiva del reductor electrónico		+	-	PM-NC
EGON	P	Activación del reductor electrónico		+	-	PM-NC
EGONSYN	P	Activación del reductor electrónico		+	-	PM-NC
EGONSYNE	P	Activación del reductor electrónico, con definición del modo de aproximación		+	-	PM-NC
ELSE	K	Programa CN: Bifurcación de programa en caso de no cumplirse la condición IF		+	-	PM-NC
ELSE	K	Acción síncrona: Disparo de acciones en caso de no cumplirse la condición		-	+	FM-SA
ENABLE	P	Interrupción activada		+	-	PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1][2][3][4][5] Ver explicación en la leyenda (Página 1237).						
ENAT [6]	G	Transición de curva natural a la siguiente secuencia de desplazamiento	m	+		PM-NC
ENDFOR	K	Línea final del bucle contador FOR		+		PM-NC
ENDIF	K	Línea final de la bifurcación IF		+		PM-NC
ENDLABEL	K	Marca final para repeticiones del programa de pieza mediante REPEAT		+		PM-NC, FM-B
ENDLOOP	K	Línea final del bucle de programa sin fin LOOP		+		PM-NC
ENDPROC	K	Línea final de un programa con la línea inicial PROC		+		
ENDWHILE	K	Línea final del bucle WHILE		+		PM-NC
ESRR	P	Parametrizar en accionamiento retirada ESR integrada en accionamiento		+		PM-NC
ESRS	P	Parametrizar en accionamiento parada ESR integrada en accionamiento		+		PM-NC
ETAN	G	Transición de curva tangencial a la próxima secuencia de desplazamiento para inicio spline	m	+		PM-NC
EVERY	K	Ejecutar una acción síncrona al pasar la condición de FALSE a TRUE		-	+	FM-SA
EX	K	Palabra reservada para la asignación de valores en notación exponencial		+		PM-NC
EXECSTRING	P	Transferencia de una variable string con la línea del programa de pieza que debe ejecutarse		+	-	PM-NC
EXECTAB	P	Ejecutar un elemento de una tabla de movimientos		+	-	PM-NC
EXECUTE	P	Ejecución del programa CON		+	-	PM-NC
EXP	F	Función exponencial ex		+	+	PM-NC
EXTCALL	A	Ejecutar subprograma externo		+	+	PM-NC
EXTCLOSE	P	Cerrar unidad/fichero externo abierto para escribir		+	-	PM-NC
EXTERN	K	Declaración de un subprograma con transferencia de parámetros		+		PM-NC
EXTOPEN	P	Abrir unidad/fichero externo para escribir en el canal		+	-	PM-NC
F	A	Valor de avance (en combinación con G4, el tiempo de espera se programa también con F)		+	+	PM-NC
FA	K	Avance por eje	m	+	+	PM-NC
FAD	A	Avance de penetración para aproximación y retirada suaves		+		PM-NC
FALSE	K	Constante lógica: falso		+	+	PM-NC
FB	A	Avance por secuencia		+		PM-NC
FCTDEF	P	Definición de la función polinomio		+	-	PM-NC
FCUB	G	Avance variable según spline cúbico	m	+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1][2][3][4][5] Ver explicación en la leyenda (Página 1237).						
FD	A	Avance de contorneado para corrección por volante	s	+		PM-NC
FDA	K	Avance por eje para corrección por volante	s	+		PM-NC
FENDNORM [6]	G	Deceleración en los dos vértices DES	m	+		PM-NC
FFWOF [6]	G	Mando anticipativo DES	m	+		PM-NC
FFWON	G	Mando anticipativo CON	m	+		PM-NC
FGREF	K	Radio de referencia en ejes giratorios o factores de referencia de trayectoria en ejes de orientación (interpolación vectorial)	m	+		PM-NC
FGROUP	P	Determinación de los ejes con avance de contorneado		+	-	PM-NC
FI	K	Parámetros para el acceso a datos frame: decalaje fino		+		PM-NC
FIFOCTRL	G	Control de la memoria (búfer) de pretratamiento	m	+		PM-NC
FILEDATE	P	Indica la fecha del último acceso de escritura al fichero		+	-	PM-NC
FILEINFO	P	Indica la suma de FILEDATE, FILESIZE, FILESTAT y FILETIME en total.		+	-	PM-NC
FILESIZE	P	Indica el tamaño actual del fichero.		+	-	PM-NC
FILESTAT	P	Indica el estado del fichero por lo que respecta a derechos de lectura, escritura, ejecución, visualización y borrado (rwxsd)		+	-	PM-NC
FILETIME	P	Indica la hora del último acceso de escritura al fichero		+	-	PM-NC
FINEA	K	Fin de movimiento al alcanzar la "Parada precisa fina"	m	+		PM-NC
FL	K	Velocidad límite para ejes síncronos	m	+		PM-NC
FLIN	G	Avance lineal variable	m	+		PM-NC
FMA	K	Varios avances por eje	m	+		PM-NC
FNORM [6]	G	Avance normal según DIN66025	m	+		PM-NC
FOC	K	Limitación de par/fuerza activa secuencia a secuencia	s	-	+	FM-SA
FOCOF	K	Desactivación de la limitación de par/fuerza modal	m	-	+	FM-SA
FOCON	K	Activación de la limitación de par/fuerza modal	m	-	+	FM-SA
FOR	K	Bucle contador con número fijo de pasadas		+		PM-NC
FP	A	Punto fijo: número del punto fijo que se debe alcanzar	s	+		PM-NC
FPO	K	Variación del avance programada mediante un polinomio		+		PM-NC
FPR	P	Identificación eje giratorio		+	-	PM-NC
FPRAOF	P	Desactivar avance por vuelta		+	-	PM-NC
FPRAON	P	Activar avance por vuelta		+	-	PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1][2][3][4][5] Ver explicación en la leyenda (Página 1237).						
FRAME	K	Tipo de datos para la determinación de sistemas de coordenadas		+		PM-NC
FRC	A	Avance para el radio y chaflán	s	+		PM-NC
FRCM	A	Avance modal para radio y chaflán	m	+		PM-NC
FROM	K	La acción se ejecuta si la condición se cumple una vez y mientras la acción síncrona esté activa		-	+	FM-SA
FTOC	P	Modificación de la corrección de herramienta fina		-	+	FM-SA
FTOCOF [6]	G	Corrección de herramienta fina activa online DES	m	+		PM-NC
FTOCON	G	Corrección de herramienta fina activa online CON	m	+		PM-NC
FXS	K	Desplazamiento a tope fijo	m	+	+	PM-NC
FXST	K	Límite de par para desplazamiento a tope fijo	m	+	+	PM-NC
FXSW	K	Ventana de vigilancia para desplazamiento a tope fijo		+	+	PM-NC
FZ	K	Avance por diente	m	+		PM-NC

Instrucciones G ... L

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
G0	G	Interpolación lineal con velocidad de desplazamiento rápido (movimiento en desplazamiento rápido)	m	+		PM-NC
G1 [6]	G	Interpolación lineal con avance (interpolación lineal)	m	+		PM-NC
G2	G	Interpolación circular en sentido horario	m	+		PM-NC
G3	G	Interpolación circular en sentido antihorario	m	+		PM-NC
G4	G	Tiempo de espera, temporizado	s	+		PM-NC
G5	G	Rectificado oblicuo de ranuras	s	+		PM-NC
G7	G	Movimiento de compensación en el rectificado oblicuo de ranuras	s	+		PM-NC
G9	G	Parada precisa, reducción de velocidad	s	+		PM-NC
G17 [6]	G	Selección del plano de trabajo X/Y	m	+		PM-NC
G18	G	Selección del plano de trabajo Z/X	m	+		PM-NC
G19	G	Selección del plano de trabajo Y/Z	m	+		PM-NC
G25	G	Limitación inferior del campo de trabajo	s	+		PM-NC
G26	G	Limitación superior del campo de trabajo	s	+		PM-NC
G33	G	Tallado de roscas con paso constante	m	+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
G34	G	Tallado de roscas con paso linealmente creciente	m	+		PM-NC
G35	G	Tallado de roscas con paso linealmente decreciente	m	+		PM-NC
G40 [6]	G	Corrección radio herramienta DES	m	+		PM-NC
G41	G	Corrección del radio de la herramienta a la izquierda del contorno	m	+		PM-NC
G42	G	Corrección del radio de la herramienta a la derecha del contorno	m	+		PM-NC
G53	G	Supresión del decalaje de origen actual (por secuencia)	s	+		PM-NC
G54	G	1.er decalaje de origen ajustable	m	+		PM-NC
G55	G	2.º decalaje de origen ajustable	m	+		PM-NC
G56	G	3.er decalaje de origen ajustable	m	+		PM-NC
G57	G	4.º decalaje de origen ajustable	m	+		PM-NC
G58 (840D sl)	G	Decalaje de origen absoluto programable (decalaje aproximado)	s	+		PM-NC
G58 (828D)	G	5.º decalaje de origen ajustable	m	+		PM-NC
G59 (840D sl)	G	Decalaje de origen programable aditivo (decalaje preciso)	s	+		PM-NC
G59 (828D)	G	6.º decalaje de origen ajustable	m	+		PM-NC
G60 [6]	G	Parada precisa, reducción de velocidad	m	+		PM-NC
G62	G	Deceleración en los dos vértices en esquinas interiores con corrección del radio de herramienta activa (G41, G42)	m	+		PM-NC
G63	G	Roscado con macho con mandril de compensación	s	+		PM-NC
G64	G	Modo de contorneado	m	+		PM-NC
G70	G	Acotado en pulgadas para datos geométricos (longitudes)	m	+	+	PM-NC
G71 [6]	G	Acotado métrico para datos geométricos (longitudes)	m	+	+	PM-NC
G74	G	Búsqueda de punto de referencia	s	+		PM-NC
G75	G	Desplazamiento a punto fijo	s	+		PM-NC
G90 [6]	G	Acotado absoluto	m/s	+		PM-NC
G91	G	Acotado incremental	m/s	+		PM-NC
G93	G	Avance inverso al tiempo en 1/min (rpm)	m	+		PM-NC
G94 [6]	G	Avance lineal F en mm/min o pulgadas/min y grados/min	m	+		PM-NC
G95	G	Avance por vuelta F en mm/vuelta o pulgadas/vuelta	m	+		PM-NC
G96	G	Avance por vuelta (como con G95) y velocidad de corte constante	m	+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
G97	G	Avance por vuelta y velocidad de giro del cabezal constante (velocidad de corte constante DES)	m	+		PM-NC
G110	G	Programación de polo relativa a la última posición nominal programada	s	+		PM-NC
G111	G	Programación de polo relativa al origen del sistema actual de coordenadas de pieza	s	+		PM-NC
G112	G	Programación de polo relativa al último polo activo	s	+		PM-NC
G140 [6]	G	Dirección para aprox./retirada suaves del contorno definida mediante G41/G42	m	+		PM-NC
G141	G	Dirección de aprox./retirada suaves a la izquierda del contorno	m	+		PM-NC
G142	G	Dirección de aprox./retirada suaves a la derecha del contorno	m	+		PM-NC
G143	G	Dirección de aprox./retirada suaves del contorno dependiente de la tangente	m	+		PM-NC
G147	G	Aproximación suave siguiendo una recta	s	+		PM-NC
G148	G	Retirada suave siguiendo una recta	s	+		PM-NC
G153	G	Supresión de frames actuales, incluido el frame básico	s	+		PM-NC
G247	G	Aproximación suave siguiendo un cuadrante de circunferencia	s	+		PM-NC
G248	G	Retirada suave siguiendo un cuadrante de circunferencia	s	+		PM-NC
G290 [6]	G	Conmutar al modo SINUMERIK CON	m	+		FM-TM
G291	G	Conmutar al modo ISO2/3 CON	m	+		FM-TM
G331	G	Roscado de taladros sin mandril de compensación, paso positivo, sentido de giro a derechas	m	+		PM-NC
G332	G	Roscado de taladros sin mandril de compensación, paso negativo, sentido de giro a izquierdas	m	+		PM-NC
G335	G	Torneado de una rosca convexa en sentido horario	m	+		PM-NC
G336	G	Torneado de una rosca convexa en sentido antihorario	m	+		PM-NC
G340 [6]	G	Secuencia de desplazamiento en el espacio (simultáneamente en profundidad y en el plano (hélice))	m	+		PM-NC
G341	G	Primero penetrar en el eje perpendicular (z); después desplazar en el plano	m	+		PM-NC
G347	G	Aproximación suave siguiendo una semicircunferencia	s	+		PM-NC
G348	G	Retirada suave siguiendo una semicircunferencia	s	+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
G450 [6]	G	Circunferencia de transición	m	+		PM-NC
G451	G	Intersección de equidistantes	m	+		PM-NC
G460 [6]	G	Activación de la vigilancia de colisión para las secuencias de aproximación y retirada	m	+		PM-NC
G461	G	Inserción de un círculo en la secuencia de corrección del radio de herramienta	m	+		PM-NC
G462	G	Inserción de una recta en la secuencia de corrección del radio de herramienta	m	+		PM-NC
G500 [6]	G	Desactivación de todos los frames ajustables; frames básicos activos	m	+		PM-NC
G505 ... G599	G	5. ... 99. Decalaje de origen ajustable	m	+		PM-NC
G601 [6]	G	Cambio de secuencia con parada precisa fina	m	+		PM-NC
G602	G	Cambio de secuencia con parada precisa basta	m	+		PM-NC
G603	G	Cambio de secuencia para interpolador - fin de secuencia	m	+		PM-NC
G621	G	Deceleración en los dos vértices en todas las esquinas	m	+		PM-NC
G641	G	Modo de contorneado con matado de esquinas según criterio de trayectoria (= distancia de matado programable)	m	+		PM-NC
G642	G	Modo de contorneado con matado de esquinas respetando las tolerancias definidas	m	+		PM-NC
G643	G	Modo de contorneado con matado de esquinas respetando las tolerancias definidas (internas de la secuencia)	m	+		PM-NC
G644	G	Modo de contorneado con matado de esquinas con la máxima dinámica posible	m	+		PM-NC
G645	G	Modo de contorneado con matado de esquinas y transiciones de secuencia tangenciales respetando las tolerancias definidas	m	+		PM-NC
G700	G	Acotado en pulgadas para datos geométricos y tecnológicos (longitudes, avance)	m	+	+	PM-NC
G710 [6]	G	Acotado métrico para datos geométricos y tecnológicos (longitudes, avance)	m	+	+	PM-NC
G810 [6], ..., G819	G	Grupo G reservado para el usuario OEM		+		PM-NC
G820 [6], ..., G829	G	Grupo G reservado para el usuario OEM		+		PM-NC
G931	G	Especificación del avance mediante el tiempo de desplazamiento, desactivar la velocidad de contorneado constante	m	+		
G942	G	Avance lineal y velocidad de corte constante o congelar velocidad de giro del cabezal	m	+		
G952	G	Avance por vuelta y velocidad de corte constante o congelar velocidad de giro del cabezal	m	+		
G961	G	Avance lineal (como con G94) y velocidad de corte constante	m	+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
G962	G	Avance lineal o avance por vuelta y velocidad de corte constante	m	+		PM-NC
G971	G	Avance lineal y velocidad de giro del cabezal constante (velocidad de corte constante DES)	m	+		PM-NC
G972	G	Avance lineal o avance por vuelta y velocidad de giro del cabezal constante (velocidad de corte constante DES)	m	+		PM-NC
G973	G	Avance por vuelta sin limitación de la velocidad de giro del cabezal y velocidad de giro del cabezal constante (G97 sin LIMS para modo ISO)	m	+		PM-NC
GEOAX	P	Asignar a los ejes geométricos 1 - 3 nuevos ejes del canal		+	-	PM-NC
GET	P	Cambiar el eje habilitado entre canales		+	+	PM-NC
GETACTT	F	Define la herramienta activa a partir de un grupo de herramientas con el mismo nombre		+	-	FM-TM
GETACTTD	F	Define el número T correspondiente para un número D absoluto		+	-	PM-NC
GETD	P	Cambiar el eje directamente entre canales		+	-	PM-NC
GETDNO	F	Indica el número D de un filo (CE) de una herramienta (T)		+	-	PM-NC
GETEXET	P	Lectura del número T cambiado		+	-	FM-TM
GETFREELOC	P	Buscar un puesto vacío en los almacenes para una determinada herramienta		+	-	FM-TM
GETSELT	P	Suministrar número T seleccionado		+	-	FM-TM
GETT	F	Determinar número T de un nombre de herramienta		+	-	FM-TM
GETTCOR	F	Leer longitudes de herramienta o componentes longitudinales de herramienta		+	-	PM-NC
GETTENV	F	Leer números T, D y DL		+	-	PM-NC
GETVARAP	F	Leer el derecho de acceso a una variable de sistema/usuario		+	-	PM-NC
GETVARDFT	F	Leer valor estándar de una variable de sistema/usuario		+	-	PM-NC
GETVARLIM	F	Leer umbrales de una variable de sistema/ usuario		+	-	PM-NC
GETVARPHU	F	Leer unidad física de una variable de sistema/ usuario		+	-	PM-NC
GETVARTYP	F	Leer tipo de dato de una variable de sistema/ usuario		+	-	PM-NC
GFRAME0 ... GFRAME100	G	Activación del frame de rectificado <n> de la gestión de datos del canal	m	+		PM-NC
GOTO	K	Instrucción de salto primero hacia delante y después hacia atrás (dirección primero hacia el fin del programa y después hacia el inicio del programa)		+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
GOTOB	K	Instrucción de salto hacia atrás (en dirección al inicio del programa)		+		PM-NC
GOTOC	K	Como GOTO, pero suprimiendo la alarma 14080 "Destino del salto no encontrado"		+		PM-NC
GOTOF	K	Instrucción de salto hacia delante (en dirección al final del programa)		+		PM-NC
GOTOS	K	Salto hacia atrás al inicio del programa		+		PM-NC
GP	K	Palabra reservada para la programación indirecta de atributos de posición		+		PM-NC
GROUP_ ADDEND	C (T)	Fin adición aprox.		+		PM-NC
GROUP_BEGIN	C (T)	- Inicio de grupo de programa		+		PM-NC
GROUP_END	C (T)	- Fin de grupo de programa		+		PM-NC
GWPSOF	P	Desactivar la velocidad periférica de muela constante (SUG)	s	+	-	PM-NC
GWPSON	P	Activar la velocidad periférica de muela constante (SUG)	s	+	-	PM-NC
H...	A	Emisión de funciones auxiliares al PLC		+	+	PM-NC, FM-B
HOLES1	C (T)	Fila de agujeros		+		PM-NC
HOLES2	C (T)	Agujeros en círculo		+		PM-NC
I	A	Parámetro de interpolación	s	+		PM-NC
I1	A	Coordenada del punto intermedio	s	+		PM-NC
IC	K	Acotado incremental	s	+		PM-NC
ICYCOF	P	Ejecución de todas las secuencias de un ciclo tecnológico según ICYCOF en un ciclo IPO		+	+	FM-SA
ICYCON	P	Ejecutar cada secuencia de un ciclo tecnológico según ICYCON en un ciclo IPO independiente		+	+	FM-SA
ID	K	Identifica acciones síncronas modales	m	-	+	FM-SA
IDS	K	Identifica acciones síncronas estáticas modales		-	+	FM-SA
IF	K	Introducción de un salto condicionado en el programa de pieza/ciclo tecnológico		+	+	PM-NC
INDEX	F	Definir un índice de un carácter en la cadena de caracteres introducida		+	-	PM-NC
INICF	K	Inicialización de las variables con NEWCONF		+		PM-NC
INIPO	K	Inicialización de las variables con PowerOn		+		PM-NC
INIRE	K	Inicialización de las variables con Reset		+		PM-NC
INIT	P	Selección de un determinado programa CN para la ejecución en un determinado canal		+	-	PM-NC
INITIAL		Creación de un fichero INI en todas las áreas		+		PM-NC
INT	K	Tipo de dato: valor entero con signo		+		PM-NC
INTERSEC	F	Calcular el punto de intersección entre dos elementos de contorno		+	-	PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
INVCCW	G	Desplazar evoluta en sentido antihorario	m	+		PM-NC
INVCW	G	Desplazar evoluta en sentido horario	m	+		PM-NC
INVFRAME	F	Calcular el frame inverso a partir de un frame		+	-	FM-B
IP	K	Parámetro de interpolación variable		+		PM-NC
IPOBRKA	P	Criterio de desplazamiento desde el punto de aplicación de la rampa de frenado	m	+	+	
IPOENDA	K	Fin de movimiento al alcanzar la "Parada IPO"	m	+		PM-NC
IPTRLOCK	P	Congelar inicio de la sección de programa sin posibilidad de búsqueda en la siguiente secuencia de función de la máquina.	m	+	-	PM-NC
IPTRUNLOCK	P	Ajustar fin de la sección de programa sin posibilidad de búsqueda a la secuencia actual en el momento de la interrupción.	m	+	-	PM-NC
IR	A	Coordenada del centro del círculo (dirección X) al tornear roscas convexas		+		PM-NC
ISAXIS	F	Comprobar si el eje geométrico indicado como parámetro es 1		+	-	PM-NC
ISD	A	Profundidad de penetración	m	+		PM-NC
ISFILE	F	Comprobar si existe un fichero en la memoria de aplicación CN		+	-	PM-NC
ISNUMBER	F	Comprobar si es posible convertir la cadena de caracteres introducida en un valor numérico		+	-	PM-NC
ISOCALL	K	Llamada indirecta de un programa programado en lenguaje ISO		+		PM-NC
ISVAR	F	Comprobar si el parámetro de transferencia contiene una variable conocida para el CN		+	-	PM-NC
J	A	Parámetro de interpolación	s	+		PM-NC
J1	A	Coordenada del punto intermedio	s	+		PM-NC
JERKA	P	Activar características de aceleración ajustadas en los datos de máquina para los ejes programados		+	-	
JERKLIM	K	Adaptación de tirón por eje máximo	m	+		PM-NC
JERKLIMA	K	Reducción o aumento del tirón máximo de un eje esclavo	m	+	+	PM-NC
JR	A	Coordenada del centro del círculo (dirección Y) al tornear roscas convexas		+		PM-NC
K	A	Parámetro de interpolación	s	+		PM-NC
K1	A	Coordenada del punto intermedio	s	+		PM-NC
KONT	G	Rodear el contorno en la corrección de herramienta	m	+		PM-NC
KONTC	G	Aproximación/retirada con polinomio de curvatura continua	m	+		PM-NC
KONTT	G	Aproximación/retirada con polinomio de tangente continua	m	+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
KR	A	Coordenada del centro del círculo (dirección Z) al tornear roscas convexas		+		PM-NC
L	A	Número de subprograma	s	+	+	PM-NC
LEAD	A	Ángulo de avance 1. Orientación de la herramienta 2. Polinomios de orientación	m	+		PM-NC
LEADOF	P	Acoplamiento axial de valores maestros DES		+	+	PM-NC
LEADON	P	Acoplamiento axial de valores maestros CON		+	+	PM-NC
LENTOAX	F	Proporciona información acerca de la asignación de las longitudes L1, L2 y L3 de la herramienta activa a los ejes de abscisas, ordenadas y aplicadas		+	-	PM-NC
LFOF [6]	G	Retirada rápida para el tallado de roscas DES	m	+		PM-NC
LFON	G	Retirada rápida para el tallado de roscas CON	m	+		PM-NC
LFPOS	G	Retirada del eje declarado con POLFMASK o POLFMLIN a la posición de eje absoluta programada con POLF	m	+		PM-NC
LFTXT [6]	G	El plano del movimiento de retroceso en la retirada rápida se determina a partir de la tangente de trayectoria y de la orientación de herramienta actual	m	+		PM-NC
LFWP	G	El plano del movimiento de retroceso en la retirada rápida se determina mediante el plano de trabajo actual (G17/G18/G19)	m	+		PM-NC
LIFTFAST	K	Retirada rápida		+		PM-NC
LIMS	K	Limitación de la velocidad de giro con G96/G961 y G97	m	+		PM-NC
LLI	K	Límite inferior de las variables		+		PM-NC
LN	F	Logaritmo neperiano (logaritmo natural)		+	+	PM-NC
LOCK	P	Bloquear acción síncrona con ID (parar ciclo tecnológico)		-	+	FM-SA
LONGHOLE	C (T)	Agujero rasgado		+		PM-NC
LOOP	K	Introducción de un bucle sin fin		+		PM-NC

Instrucciones M ... R

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
M0		Parada programada		+	+	PM-NC
M1		Parada opcional		+	+	PM-NC
M2		Fin del programa principal (como M30)		+	+	PM-NC
M3		Giro de cabezal a derechas		+	+	PM-NC
M4		Giro de cabezal a izquierdas		+	+	PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
M5		Parada del cabezal		+	+	PM-NC
M6		Cambio de herramienta		+	+	PM-NC
M17		Fin del subprograma		+	+	PM-NC
M19		Posicionado del cabezal en la posición ajustada en SD43240		+	+	PM-NC
M30		Fin del programa principal (como M2)		+	+	PM-NC
M40		Cambio automático de gama o escalón de reducción		+	+	PM-NC
M41 ... M45		Nivel de reducción 1 ... 5		+	+	PM-NC
M70		Transición a modo Eje		+	+	PM-NC
MASLDEF	P	Definir conjunto de ejes maestro/esclavo		+	+	PM-NC
MASLDEL	P	Separar conjunto de ejes maestro/esclavo y borrar definición del conjunto		+	+	PM-NC
MASLOF	P	Desconexión de un acoplamiento temporal		+	+	PM-NC
MASLOFS	P	Desconexión de un acoplamiento temporal con parada automática del eje esclavo		+	+	PM-NC
MASLON	P	Conexión de un acoplamiento temporal		+	+	PM-NC
MATCH	F	Búsqueda de una cadena de caracteres en el string		+	-	PM-NC
MAXVAL	F	Valor más grande de dos variables (función trigon.)		+	+	PM-NC
MCALL	K	Llamada de subprograma modal		+		PM-NC
MEAC	K	Medición continua por eje sin borrado de trayecto residual	s	+	+	PM-NC
MEAFRAME	F	Cálculo de frame a partir de puntos de medida		+	-	PM-NC
MEAS	A	Medición con borrado de trayecto residual	s	+		PM-NC
MEASA	K	Medición por eje con borrado de trayecto residual	s	+	+	PM-NC
MEASURE	F	Método de cálculo para la medición de pieza y de herramienta		+	-	FM-TE
MEAW	A	Medición sin borrado de trayecto residual	s	+		PM-NC
MEAWA	K	Medición por eje sin borrado de trayecto residual	s	+	+	PM-NC
MI	K	Acceso a datos de frame: Simetría especular		+		PM-NC
MINDEX	F	Definir un índice de un carácter en la cadena de caracteres introducida		+	-	PM-NC
MINVAL	F	Valor más pequeño de dos variables (función trigon.)		+	+	PM-NC
MIRROR	G	Simetría programable	s	+		PM-NC
MMC	P	Llamar ventana de diálogo de forma interactiva desde el programa de pieza en el HMI		+	-	PM-NC
MOD	K	División de módulo		+		PM-NC
MODAXVAL	F	Calcular la posición de módulo de un eje giratorio con módulo		+	-	PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
MOV	K	Iniciar eje de posicionado		-	+	FM-SA
MOVT	A	Indicar punto final de un desplazamiento en la dirección de la herramienta				FM-B
MSG	P	Avisos programables	m	+	-	PM-NC
MVTOOL	P	Instrucción de programación para el desplazamiento de una herramienta		+	-	FM-TM
N	A	Número de secuencia auxiliar CN		+		PM-NC
NAMETOINT	F	Determinar índice de variable de sistema		+		PM-NC
CN	K	Especificación del ámbito de validez de datos		+		PM-NC
NEWCONF	P	Aceptar los datos de máquina modificados (equivale a "activar los datos de máquina")		+	-	PM-NC
NEWMT	F	Crear nueva Multitool		+	-	FM-TM
NEWT	F	Crear nueva herramienta		+	-	FM-TM
NORM [6]	G	Ajuste normal en el punto inicial; final durante la corrección de herramienta	m	+		PM-NC
NOT	K	NO lógica (negación)		+		PM-NC
NPROT	P	Zona protegida específica de máquina CON/DES		+	-	PM-NC
NPROTDEF	P	Definición de una zona protegida específica de la máquina		+	-	PM-NC
NUMBER	F	Conversión de la cadena de caracteres introducida en un número		+	-	PM-NC
OEMIPO1	G	Interpolación OEM 1	m	+		PM-NC
OEMIPO2	G	Interpolación OEM 2	m	+		PM-NC
OF	K	Palabra reservada de la bifurcación CASE		+		PM-NC
OFFN	A	Demasía/creces para el contorno programado	m	+		PM-NC
OMA1	A	Dirección OEM 1	m	+		PM-NC
OMA2	A	Dirección OEM 2	m	+		PM-NC
OMA3	A	Dirección OEM 3	m	+		PM-NC
OMA4	A	Dirección OEM 4	m	+		PM-NC
OMA5	A	Dirección OEM 5	m	+		PM-NC
OR	K	Operador lógico, combinación O		+		PM-NC
ORIAXES	G	Interpolación lineal de los ejes de máquina o ejes de orientación	m	+		PM-NC
ORIAXPOS	G	Ángulo de orientación a través de ejes de orientación virtuales con posiciones de eje giratorio	m	+		PM-NC
ORIC [6]	G	Cambios de orientación en esquinas exteriores se superponen a la secuencia circular a insertar	m	+		PM-NC
ORICONCCW	G	Interpolación en una superficie envolvente de círculo en sentido antihorario	m	+		PM-NC, FM-TR
ORICONCW	G	Interpolación en una superficie envolvente de círculo en sentido horario	m	+		PM-NC, FM-TR

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
ORICONIO	G	Interpolación en una superficie envolvente de círculo con indicación de una orientación intermedia	m	+		PM-NC, FM-TR
ORICONTO	G	Interpolación en una superficie envolvente de círculo en la transición tangencial (indicación de la orientación final)	m	+		PM-NC, FM-TR
ORICURVE	G	Interpolación de la orientación con especificación del movimiento de dos puntos de contacto de la herramienta	m	+		PM-NC, FM-TR
ORID	G	Cambios de orientación se ejecutan antes de la secuencia circular	m	+		PM-NC
ORIEULER [6]	G	Ángulo de orientación mediante ángulos eulerianos	m	+		PM-NC
ORIMKS	G	Orientación de herramienta en el sistema de coordenadas de máquina	m	+		PM-NC
ORIPATH	G	Orientación de la herramienta con relación a la trayectoria	m	+		PM-NC
ORIPATHS	G	Orientación de la herramienta con relación a la trayectoria; se suaviza un acodamiento en el desarrollo de la orientación	m	+		PM-NC
ORIPLANE	G	Interpolación en un plano (corresponde a ORIVECT) interpolación circular de gran radio	m	+		PM-NC
ORIRESET	P	Estado inicial de la orientación de herramienta con hasta 3 ejes de orientación		+	-	PM-NC
ORIROTA [6]	G	Ángulo de rotación frente a un sentido de giro absoluto especificado	m	+		PM-NC
ORIROTC	G	Vector de giro tangencial a la trayectoria tangente	m	+		PM-NC
ORIROTR	G	Ángulo de rotación relativo al plano entre la orientación inicial y final	m	+		PM-NC
ORIROTT	G	Ángulo de rotación relativo a la modificación del vector de orientación	m	+		PM-NC
ORIRPY	G	Ángulo de orientación mediante ángulos RPY (XYZ)	m	+		PM-NC
ORIRPY2	G	Ángulo de orientación mediante ángulos RPY (ZYX)	m	+		PM-NC
ORIS	A	Cambio de orientación	m	+		PM-NC
ORISOF [6]	G	Alisado del desarrollo de la orientación DES	m	+		PM-NC
ORISOLH	F	Cálculo de orientaciones		+		PM-NC
ORISON	G	Alisado del desarrollo de la orientación CON	m	+		PM-NC
ORIVECT [6]	G	Interpolación circular de gran radio (idéntico a ORIPLANE)	m	+		PM-NC
ORIVIRT1	G	Ángulo de orientación mediante ejes de orientación virtuales (definición 1)	m	+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
ORIVIRT2	G	Ángulo de orientación mediante ejes de orientación virtuales (definición 1)	m	+		PM-NC
ORIWKS [6]	G	Orientación de herramienta en el sistema de coordenadas de pieza	m	+		PM-NC
OS	K	Activar/desactivar vaivén (oscilación)		+		PM-NC
OSB	K	Vaivén: punto inicial	m	+		PM-NC
OSC	G	Alisado de la orientación de herramienta constante	m	+		PM-NC
OSCILL	K	Axis: 1 - 3 ejes de penetración	m	+		PM-NC
OSCTRL	K	Opciones de vaivén	m	+		PM-NC
OSD	G	Matado de esquinas de la orientación de herramienta con especificación de la longitud de matado de esquina con DO	m	+		PM-NC
OSE	K	Punto final vaivén	m	+		PM-NC
OSNSC	K	Vaivén: número de ciclos de afinado	m	+		PM-NC
OSOF [6]	G	Alisado orientación de herramienta DES	m	+		PM-NC
OSP1	K	Vaivén: punto de inversión izquierdo	m	+		PM-NC
OSP2	K	Vaivén: punto de inversión derecho	m	+		PM-NC
OSS	G	Alisado de la orientación de herramienta al final de la secuencia	m	+		PM-NC
OSSE	G	Alisado de la orientación de herramienta al principio y al final de la secuencia	m	+		PM-NC
OST	G	Matado de esquinas de la orientación de herramienta con especificación de la tolerancia angular en grados con el DO (desviación máxima del desarrollo de la orientación programado)	m	+		PM-NC
OST1	K	Vaivén: punto de parada en el punto de inversión izquierdo	m	+		PM-NC
OST2	K	Vaivén: punto de parada en el punto de inversión derecho	m	+		PM-NC
OTOL	A	Tolerancia de orientación para funciones de compresor, alisado de la orientación y tipos de matado de esquinas	m	+	-	PM-NC
OTOLG0	A	Tolerancia de orientación con desplazamientos en rápido	m	+	-	PM-NC
OVR	K	Corrección de la velocidad de giro	m	+		PM-NC
OVRA	K	Corrección por eje de la velocidad de giro	m	+	+	PM-NC
OVRRAP	K	Corrección en rápido	m	+		PM-NC
P	A	Número de pasadas del subprograma		+		PM-NC
PAROT	G	Alinear sistema de coordenadas de pieza en la pieza	m	+		PM-NC
PAROTOF [6]	G	Desactivar la rotación de frame relativa a la pieza	m	+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
PCALL	K	Subprogramas con ruta de acceso absoluta y transferencia de parámetros		+		PM-NC
PDELAYOF	G	Retardo en el troquelado DES	m	+		PM-NC
PDELAYON [6]	G	Retardo en el troquelado CON	m	+		PM-NC
PHI	K	Ángulo de giro de la orientación alrededor del eje de dirección del cono		+		PM-NC
PHU	K	Unidad física de una variable		+		PM-NC
PL	A	1. B spline: distancia entre nodos 2. Interpolación polinómica duración del intervalo de parámetros en la interpolación polinómica	s	+		PM-NC
PM	K	Por minuto		+		PM-NC
PO	K	Coeficiente de polinomio en la interpolación polinómica	s	+		PM-NC
POCKET3	C (T)	Fresar caja rectangular		+		PM-NC
POCKET4	C (T)	Fresar caja circular		+		PM-NC
POLF	K	Posición de retirada LIFTFAST	m	+		PM-NC
POLFA	P	Iniciar posición de retirada de ejes individuales con $AA_ESR_TRIGGER	m	+	+	PM-NC
POLFMASK	P	Liberar ejes para la retirada sin relación entre los ejes	m	+	-	PM-NC
POLFMLIN	P	Liberar ejes para la retirada con relación lineal entre los ejes	m	+	-	PM-NC
POLY	G	Interpolación de polinomios	m	+		PM-NC
POLYPATH	P	Interpolación polinómica seleccionable para los grupos de ejes AXIS o VECT	m	+	-	PM-NC
PON	G	Troquelado CON	m	+		PM-NC
PONS	G	Troquelado CON en el ciclo IPO	m	+		PM-NC
POS	K	Posicionar eje		+	+	PM-NC
POSA	K	Posicionar eje salvando límites de secuencia		+	+	PM-NC
POSM	P	Posicionar almacén		+	-	FM-TM
POSMT	P	Posicionar Multitool en número de puesto de portaherramientas		+	-	FM-TM
POSP	K	Posicionado en tramos (vaivén)		+		PM-NC
POSRANGE	F	Calcular si la posición teórica interpolada actualmente de un eje se encuentra en una ventana en torno a una posición de referencia predefinida		+	+	FM-SA
POT	F	Cuadrado (función aritmética)		+	+	PM-NC
PR	K	Por vuelta		+		PM-NC
PREPRO	PA	Identificación de subprogramas con preparación		+		PM-NC
PRESETON	P	Ajuste de valores reales con pérdida del estado de referencia		+	+	PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
PRESETONS	P	Ajuste de valores reales sin pérdida del estado de referencia		+	+	PM-NC
PRIO	K	Palabra reservada para definir la prioridad en el tratamiento de interrupciones		+		PM-NC
PRLOC	K	Inicialización de las variables con Reset solo tras modificación local		+		PM-NC
PROC	K	Primera instrucción de un programa		+		PM-NC
PROTA	P	Solicitar nuevo cálculo del modelo de colisión		+		PM-NC
PROTD	F	Calcular la distancia entre dos zonas protegidas		+		PM-NC
PROTS	P	Asignar estado a la zona protegida		+		PM-NC
PSI	K	Ángulo en el vértice del cono		+		PM-NC
PTP	G	Desplazamiento punto a punto	m	+		PM-NC
PTPG0	G	Desplazamiento punto a punto solo con G0; si no, desplazamiento de trayectoria CP	m	+		PM-NC
PTPWOC	G	Desplazamiento punto a punto son movimientos de compensación causados por cambios de orientación	m	+		PM-NC
PUNCHACC	P	Aceleración dependiente del recorrido durante el punzonado		+	-	PM-NC
PUTFTOC	P	Corrección de herramienta fina para diamantado paralelo		+	-	PM-NC
PUTFTOCF	P	Corrección de herramienta fina dependiente de una función definida con FCTDEF para diamantado paralelo		+	-	PM-NC
PW	A	B-spline, peso del punto	s	+		PM-NC
QU	K	Emisión rápida de funciones (auxiliares) adicionales		+		PM-NC
R...	A	Parámetro de cálculo también como identificador de direcciones ajustable y con extensión numérica		+		PM-NC
RAC	K	Programación por radios específica del eje, por secuencias, absoluta	s	+		PM-NC
RDISABLE	P	Bloqueo de lectura		-	+	FM-SA
READ	P	Lee una o varias líneas en el fichero indicado y guarda información leída en la matriz		+	-	PM-NC
REAL	K	Tipo de dato: variable de coma flotante con signo (números reales)		+		PM-NC
REDEF	K	Redefinición de variables de sistema, variables de usuario e instrucciones de programación del CN		+		PM-NC
RELEASE	P	Habilitar ejes de máquina para el intercambio		+	+	PM-NC
REP	K	Palabra reservada para inicializar todos los elementos de una matriz con el mismo valor		+		PM-NC
REPEAT	K	Repetición de un bucle de programa		+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
REPEATB	K	Repetición de una línea de programa		+		PM-NC
REPOSA	G	Reposicionamiento en el contorno lineal con todos los ejes	s	+		PM-NC
REPOSH	G	Reposicionamiento en el contorno en semicírculo	s	+		PM-NC
REPOSHA	G	Reposicionamiento en el contorno con todos los ejes; ejes geométricos en semicírculo	s	+		PM-NC
REPOSL	G	Reposicionamiento en el contorno lineal	s	+		PM-NC
REPOSQ	G	Reposicionamiento en el contorno en cuadrante	s	+		PM-NC
REPOSQA	G	Reposicionamiento en el contorno lineal con todos los ejes; ejes geométricos en cuadrante	s	+		PM-NC
RESETMON	P	Instrucción de programación para la activación de valores de consigna		+	-	FM-TM
RET	P	Fin de subprograma		+	+	PM-NC
RETB	P	Fin de subprograma		+	+	PM-NC
RIC	K	Programación por radio específica del eje, por secuencias, incremental	s	+		PM-NC
RINDEX	F	Definir un índice de un carácter en la cadena de caracteres introducida		+	-	PM-NC
RMB	G	Reposicionamiento en el punto inicial de la secuencia	m	+		PM-NC
RMBBL	G	Reposicionamiento en el punto inicial de la secuencia	s	+		PM-NC
RME	G	Reposicionamiento en el punto final de la secuencia	m	+		PM-NC
RMEBL	G	Reposicionamiento en el punto final de la secuencia	s	+		PM-NC
RMI [6]	G	Reposicionamiento en el punto de interrupción	m	+		PM-NC
RMIBL [6]	G	Reposicionamiento en el punto de interrupción	s	+		PM-NC
RMN	G	Reposicionamiento en el punto de trayectoria más cercano	m	+		PM-NC
RMNBL	G	Reposicionamiento en el punto de trayectoria más cercano	s	+		PM-NC
RND	A	Redondear esquina de contorno	s	+		PM-NC
RNDM	A	Redondeo modal	m	+		PM-NC
ROT	G	Rotación programable	s	+		PM-NC
ROTS	G	Rotaciones de frame programables con ángulos espaciales	s	+		PM-NC
ROUND	F	Redondeo de decimales		+	+	PM-NC
ROUNDUP	F	Redondeo al alza de un valor de entrada		+	+	PM-NC
RP	A	Radio polar	m/s	+		PM-NC
RPL	A	Rotación en el plano	s	+		PM-NC
RT	K	Parámetros para el acceso a datos frame: Rotación		+		PM-NC

Instrucción	Clase [1]	Significado	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
RTLIOF	G	G0 sin interpolación lineal (interpolación de ejes individuales)	m	+		PM-NC
RTLION [6]	G	G0 con interpolación lineal	m	+		PM-NC

Instrucciones S ... Z

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
S	A	Velocidad de giro del cabezal (otro significado con G4, G96/G961)	m/s	+	+	PM-NC
SAVE	PA	Atributo para salvaguardar información de llamadas de subprograma		+		PM-NC
SBLOF	P	Supresión de secuencia individual		+	-	PM-NC
SBLON	P	Cancelación de la supresión de secuencia individual		+	-	PM-NC
SC	K	Parámetros para el acceso a datos frame: Escalado		+		PM-NC
SCALE	G	Escala programable	s	+		PM-NC
SCC	K	Asignación selectiva de un eje de refrentado a G96/G961/G962. Los identificadores de eje pueden ser ejes geométricos, ejes de canal o de máquina		+		PM-NC
SCPARA	K	Programación de la secuencia de parámetros servo		+	+	PM-NC
SD	A	Grado del spline	s	+		PM-NC
SET	K	Palabra reservada para inicializar todos los elementos de una matriz con valores listados		+		PM-NC
SETAL	P	Activar alarma		+	+	PM-NC
SETDNO	F	Asignación del número D del filo (CE) de una herramienta (T)		+	-	PM-NC
SETINT	K	Determinación de qué rutina de interrupción se debe activar cuando aparece una entrada CN		+		PM-NC
SETM	P	Activación de metas en el canal propio		+	+	PM-NC
SETMS	P	Retornar al cabezal maestro especificado en el dato de máquina		+	-	PM-NC
SETMS(n)	P	Cabezal n debe actuar como cabezal maestro		+		PM-NC
SETMTH	P	Definir número de portaherramientas maestro		+	-	FM-TM
SETPIECE	P	Define el número de piezas para todas las herramientas asignadas al cabezal		+	-	FM-TM
SETTA	P	Definir herramienta como activa en conjunto de desgaste		+	-	FM-TM
SETTCOR	F	Modificación de componentes de herramienta teniendo en cuenta todas las limitaciones		+	-	PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
SETTIA	P	Definir herramienta como inactiva en conjunto de desgaste		+	-	FM-TM
SF	A	Decalaje del punto inicial para el tallado de roscas	m	+		PM-NC
SIN	F	Seno (función trigon.)		+	+	PM-NC
SIRELAY	F	Activación de las funciones de seguridad parametrizadas con SIRELIN, SIRELOUT y SIRELTIME		-	+	FM-SI
SIRELIN	P	Inicialización de las magnitudes de entrada del bloque de función		+	-	FM-SI
SIRELOUT	P	Inicialización de las magnitudes de salida del bloque de función		+	-	FM-SI
SIRELTIME	P	Inicialización del temporizador del bloque de función		+	-	FM-SI
SLOT1	C (T)	Ranura longitudinal		+		PM-NC
SLOT2	C (T)	Ranura circular		+		PM-NC
SOFT	G	Aceleración de contorneado con limitación de tirones	m	+		PM-NC
SOFTA	P	Activar la aceleración con limitación de tirones de los ejes programados		+	-	PM-NC
SON	G	Punzonado CON	m	+		PM-NC
SONS	G	Punzonado CON en el ciclo IPO	m	+		PM-NC
SPATH [6]	G	La trayectoria de referencia para los ejes FGROUP es la longitud de un arco	m	+		PM-NC
SPCOF	P	Conmutar cabezal maestro o cabezales de regulación de posición a regulación de velocidad	m	+	-	PM-NC
SPCON	P	Conmutar cabezal maestro o cabezales de regulación de velocidad a regulación de posición	m	+	-	PM-NC
SPI	F	Conversión del número de cabezal en un identificador de eje		+	-	PM-NC
SPIF1 [6]	G	Entradas/salidas CN rápidas para troquelado/punzonado, byte 1	m	+		FM-TE
SPIF2	G	Entradas/salidas CN rápidas para troquelado/punzonado, byte 2	m	+		FM-TE
SPLINEPATH	P	Determinar conjunto spline		+	-	PM-NC
SPN	A	Cantidad de trayectos parciales por secuencia	s	+		PM-NC
SPOF [6]	G	Carrera DES, troquelado, punzonado DES	m	+		PM-NC
SPOS	K	Posición del cabezal	m	+	+	PM-NC
SPOSA	K	Pos. del cabezal salvando límites de secuencia	m	+		PM-NC
SPP	A	Longitud de un trayecto parcial	m	+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
SPRINT	F	Devuelve un string de entrada formateado		+		PM-NC
SQRT	F	Raíz cuadrada (función aritmética) (square root)		+	+	PM-NC
SR	A	Distancia de retirada de vaivén para acción síncrona	s	+		PM-NC
SRA	K	Distancia de retirada de vaivén en entrada externa por eje para acción síncrona	m	+		PM-NC
ST	A	Tiempo de afinado de vaivén para acción síncrona	s	+		PM-NC
STA	K	Tiempo de afinado de vaivén por eje para acción síncrona	m	+		PM-NC
START	P	Arranque de los programas seleccionados simultáneamente por varios canales desde el programa actualmente en ejecución		+	-	PM-NC
STARTFIFO [6]	G	Mecanizado; paralelamente llenado de la memoria (búfer) de pretratamiento	m	+		PM-NC
STAT		Posición de las articulaciones	s	+		PM-NC
STOLF	A	Factor de tolerancia G0	m	+		PM-NC
STOPFIFO	G	Parada del mecanizado; llenado de la memoria (búfer) de pretratamiento hasta que se reconozca STARTFIFO; memoria (búfer) de pretratamiento llena o fin de programa	m	+		PM-NC
STOPRE	P	Parada de decodificación previa hasta que se hayan ejecutado todas las secuencias preparadas del proceso principal		+	-	PM-NC
STOPREOF	P	Suprimir parada de decodificación previa		-	+	FM-SA
STRING	K	Tipo de dato: Cadena de caracteres		+		PM-NC
STRINGIS	F	Comprueba para el repertorio del lenguaje CN disponible y, de forma especial para este comando, si existen, son válidos, están definidos o activos nombres de ciclo CN, variables de usuario, macros y nombres de label		+	-	PM-NC
STRLEN	F	Definir la longitud de un string		+	-	PM-NC
SUBSTR	F	Definir un índice de un carácter en la cadena de caracteres introducida		+	-	PM-NC
SUPA	G	Supresión del decalaje de origen actual, incluidos los decalajes programados, frames de sistema, decalajes con volante (DRF), decalaje de origen externo y desplazamiento superpuesto	s	+		PM-NC
SUPD	G	Supresión de las correcciones de herramienta activas	s	+	-	PM-NC
SVC	K	Velocidad de corte de la herramienta	m	+		PM-NC
SYNFCT	P	Evaluación de un polinomio dependiente de una condición en la acción síncrona de desplazamiento		-	+	FM-SA

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
SYNR	K	La lectura de la variable se realiza de forma síncrona, es decir, en el momento de ejecución		+		PM-NC
SYNRW	K	La lectura y la escritura de la variable se realizan de forma síncrona, es decir, en el momento de ejecución		+		PM-NC
SYNW	K	La escritura de la variable se realiza de forma síncrona, es decir, en el momento de ejecución		+		PM-NC
T	A	Llamar herramienta (cambiar solo si se especifica en dato de máquina; de lo contrario, se requiere el comando M6)		+		PM-NC
TAN	F	Tangente (función trigon.)		+	+	PM-NC
TANG	P	Control tangencial: Definir acoplamiento		+	-	PM-NC
TANGDEL	P	Control tangencial: Borrar acoplamiento		+	-	PM-NC
TANGOF	P	Control tangencial: Desactivar acoplamiento		+	-	PM-NC
TANGON	P	Control tangencial: Activar acoplamiento		+	-	PM-NC
TCA (828D: _TCA)	P	Selección/cambio de herramienta independiente del estado de la herramienta		+	-	FM-TM
TCARR	A	Solicitar portaherramientas (número "m")		+		PM-NC
TCI	P	Cambio de herramienta desde la memoria intermedia al almacén		+	-	FM-TM
TCOABS [6]	G	Calcular las componentes longitudinales de la herramienta a partir de la orientación actual de la herramienta	m	+		PM-NC
TCOFR	G	Determinar las componentes longitudinales de la herramienta a partir de la orientación del frame activo	m	+		PM-NC
TCOFRX	G	Determinar la orientación de la herramienta de un frame activo en la selección de herramientas; la herramienta apunta en dirección X	m	+		PM-NC
TCOFRY	G	Determinar la orientación de la herramienta de un frame activo en la selección de herramientas; la herramienta apunta en dirección Y	m	+		PM-NC
TCOFRZ	G	Determinar la orientación de la herramienta de un frame activo en la selección de herramientas; la herramienta apunta en dirección Z	m	+		PM-NC
THETA	A	Ángulo de giro	s	+		PM-NC
TILT	A	Ángulo lateral	m	+		PM-NC
TLIFT	P	Control tangencial: Activar generación de secuencia intermedia		+	-	PM-NC
TML	P	Selección de herramienta con número de puesto de almacén		+	-	FM-TM
TMOF	P	Desactivar vigilancia de herramienta		+	-	PM-NC
TMON	P	Activar vigilancia de herramienta		+	-	PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
TO	K	Designa el valor final de un bucle contador FOR		+		PM-NC
TOFF	A	Offset de longitud de herramienta en la dirección de la componente longitudinal de herramienta que actúa paralelamente al eje geométrico indicado en el índice	m	+		PM-NC
TOFFL	A	Offset de longitud de herramienta en la dirección de la componente longitudinal de herramienta L1, L2 o L3	m	+		PM-NC
TOFFLR	A	Offset simultáneo de longitud de herramienta y radio de herramienta	m	+		PM-NC
TOFFOF	P	Reset corrección de longitud de herramienta online		+	-	PM-NC
TOFFON	P	Activación de la corrección longitudinal de herramienta online		+	-	PM-NC
TOFFR	A	Offset del radio de herramienta	m	+		PM-NC
TOFRAME	G	Orientar el eje Z del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	m	+		PM-NC
TOFRAMEX	G	Orientar el eje X del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	m	+		PM-NC
TOFRAMEY	G	Orientar el eje Y del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	m	+		PM-NC
TOFRAMEZ	G	Como TOFRAME	m	+		PM-NC
TOLOWER	F	Conversión de las letras de una cadena de caracteres en letras minúsculas		+	-	PM-NC
TOOLENV	F	Guardar todos los estados actuales importantes para la evaluación de los datos de herramienta guardados en la memoria		+	-	PM-NC
TOOLGNT	F	Determinar número de herramientas de un grupo de herramientas		+	-	FM-TM
TOOLGT	F	Determinar número T de una herramienta de un grupo de herramientas		+	-	FM-TM
TOROT	G	Orientar el eje Z del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	m	+		PM-NC
TOROTOF [6]	G	Rotaciones de frame en dirección a la herramienta DES	m	+		PM-NC
TOROTX	G	Orientar el eje X del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	m	+		PM-NC
TOROTY	G	Orientar el eje Y del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	m	+		PM-NC
TOROTZ	G	Como TOROT	m	+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
TOUPPER	F	Conversión de las letras de una cadena de caracteres en letras mayúsculas		+	-	PM-NC
TOWBCS	G	Valores de desgaste en el sistema de coordenadas básico (BKS)	m	+		PM-NC
TOWKCS	G	Valores de desgaste en el sistema de coordenadas del cabezal de herramienta con transformación cinética (se distingue del MKS por el giro de la herramienta)	m	+		PM-NC
TOWMCS	G	Valores de desgaste en el sistema de coordenadas de máquina (MKS)	m	+		PM-NC
TOWSTD [6]	G	Ajuste básico para correcciones en la longitud de la herramienta	m	+		PM-NC
TOWTCS	G	Valores de desgaste en el sistema de coordenadas de herramienta (punto de referencia del portaherramientas T en el alojamiento del portaherramientas)	m	+		PM-NC
TOWWCS	G	Valores de desgaste en el sistema de coordenadas de pieza (WKS)	m	+		PM-NC
TR	K	Componente de decalaje de una variable frame		+		PM-NC
TRAANG	P	Transformada eje oblicuo		+	-	PM-NC
TRACON	P	Transformada en cascada		+	-	PM-NC
TRACYL	P	Cilindro: Transformada de superficie envolvente		+	-	PM-NC
TRAFOOF	P	Desactivación de las transformadas activas en el canal		+	-	PM-NC
TRAFOON	P	Activar transformada definida con cadenas cinemáticas		+	-	PM-NC
TRAILOF	P	Arrastre sincronizado de eje DES		+	+	PM-NC
TRAILON	P	Arrastre sincronizado de ejes CON		+	+	PM-NC
TRANS	G	Decalaje de origen programable absoluto	s	+		PM-NC
TRANSMIT	P	Transformada polar (mecanizado de superficies frontales)		+	-	PM-NC
TRAORI	P	Transformada de 4, 5 ejes, transformada genérica		+	-	PM-NC
TRUE	K	Constante lógica: verdadero		+		PM-NC
TRUNC	F	Supresión de decimales		+	+	PM-NC
TU		Ángulo del eje	s	+		PM-NC
TURN	A	N.º de vueltas para hélices o espirales	s	+		PM-NC
ULI	K	Límite superior de las variables		+		PM-NC
UNLOCK	P	Liberar acción síncrona con ID (continuar ciclo tecnológico)		-	+	FM-SA
UNTIL	K	Condición para terminar un bucle REPEAT		+		PM-NC
UPATH	G	La trayectoria de referencia para los ejes FGROUP es el parámetro de curva	m	+		PM-NC

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
VAR	K	Palabra reservada: modo de transferencia de parámetros		+		PM-NC
VELOLIM	K	Adaptación de la velocidad máxima de eje o cabezal	m	+	-	PM-NC
VELOLIMA	K	Reducción o aumento de la velocidad máxima de un eje esclavo	m	+	+	PM-NC
WAITC	P	Esperar hasta que se haya cumplido el criterio de cambio de secuencia de acoplamiento para los ejes/cabezales		+	-	PM-NC
WAITE	P	Esperar el fin del programa en otro canal		+	-	PM-NC
WAITENC	P	Esperar a posiciones de eje sincronizadas o restauradas		+	-	PM-NC
WAITM	P	Esperar la meta en el canal indicado; terminar la secuencia anterior con parada precisa		+	-	PM-NC
WAITMC	P	Esperar meta en el canal indicado; parada precisa solo si los otros canales no han alcanzado aún la meta		+	-	PM-NC
WAITP	P	Esperar a que termine el desplazamiento del eje de posicionado		+	-	PM-NC
WAITS	P	Esperar a que se alcance la posición del cabezal		+	-	PM-NC
WALCS0 [6]	G	Limitación del campo de trabajo WKS anulada	m	+	-	PM-NC
WALCS1	G	Grupo de limitación del campo de trabajo WKS 1 activo	m	+	-	PM-NC
WALCS2	G	Grupo de limitación del campo de trabajo WKS 2 activo	m	+	-	PM-NC
WALCS3	G	Grupo de limitación del campo de trabajo WKS 3 activo	m	+	-	PM-NC
WALCS4	G	Grupo de limitación del campo de trabajo WKS 4 activo	m	+	-	PM-NC
WALCS5	G	Grupo de limitación del campo de trabajo WKS 5 activo	m	+	-	PM-NC
WALCS6	G	Grupo de limitación del campo de trabajo WKS 6 activo	m	+	-	PM-NC
WALCS7	G	Grupo de limitación del campo de trabajo WKS 7 activo	m	+	-	PM-NC
WALCS8	G	Grupo de limitación del campo de trabajo WKS 8 activo	m	+	-	PM-NC
WALCS9	G	Grupo de limitación del campo de trabajo WKS 9 activo	m	+	-	PM-NC
WALCS10	G	Grupo de limitación del campo de trabajo WKS 10 activo	m	+	-	PM-NC
WALIMOF	G	Limitación del campo de trabajo BKS DES	m	+	-	PM-NC
WALIMON [6]	G	Limitación del campo de trabajo BKS CON	m	+	-	PM-NC
WHEN	K	La acción se ejecuta una vez si se cumple la condición una vez		-	+	FM-SA

Instrucción	Clase [1]	Descripción	W [2]	TP [3]	SA [4]	Consultar la descripción en [5]
[1] [2] [3] [4] [5] Ver explicación en la leyenda (Página 1237).						
WHENEVER	K	La acción se ejecuta durante cada ciclo de interpolación si se cumple la condición		-	+	FM-SA
WHILE	K	Inicio del bucle de programa WHILE		+		PM-NC
WRITE	P	Escribir texto en el sistema de ficheros. Inserta una secuencia al final del fichero indicado		+	-	PM-NC
WRTPR	P	Escribir cadena de caracteres en variable BTSS		+	-	PM-NC
X	A	Nombre de eje	m/s	+	+	PM-NC
XOR	O	O exclusivo lógico		+		PM-NC
Y	A	Nombre de eje	m/s	+	+	PM-NC
Z	A	Nombre de eje	m/s	+	+	PM-NC

Leyenda

1) Tipo de instrucción:

A Dirección

Identificador al que se asigna un valor (p. ej., OVR=10). Existen también algunas direcciones que activan o desactivan una función sin asignación de valores (p. ej., CPLON y CPLOF).

C (A) Ciclo AST

Programa CN predefinido para la reoptimización automática con AST (= optimización automática servo). La adaptación a la situación concreta de optimización se realiza mediante parámetros que se transfieren al ciclo al llamarlos.

C (M) Ciclo de medida

Programa CN predefinido en el que está programada con validez general una determinada medición, p. ej., la determinación del diámetro interior de una pieza cilíndrica. La adaptación a la situación concreta de medida se realiza mediante parámetros que se transfieren al ciclo al llamarlos.

C (T) Ciclo tecnológico

Programa CN predefinido en el que se ha programado con carácter general un determinado proceso de mecanizado, p. ej., el taladrado de una rosca o el fresado de una caja. La adaptación a la situación concreta de mecanizado se realiza mediante parámetros que se transfieren al ciclo al llamarlos.

F Función predefinida (proporciona un valor de retorno)

La llamada a la función predefinida puede estar en la expresión como operando.

G Comando G

Los comandos G están divididos en grupos. En una secuencia solo se puede escribir un comando G. Un comando G puede actuar de forma modal (hasta que sea anulado por otro comando del mismo grupo) o aplicarse solo a la secuencia en la que se encuentra (válido por secuencias).

K Palabra reservada

Identificador que define la sintaxis de una secuencia. A una palabra reservada no se le asigna ningún valor y con ella tampoco es posible activar/desactivar ninguna función CN.

Ejemplos: estructuras de control (IF, ELSE, ENDIF, WHEN...), ejecución de programa (GOTOB, GOTO, RET…)

O Operador

Operador de una operación matemática, de comparación o lógica

P Procedimiento predefinido (no proporciona ningún valor de retorno)

PA Atributo de programa

Los atributos de programa figuran al final de la línea de definición de un subprograma:
`PROC <Nombre de programa>(...) <Atributo de programa>`

Determinan el comportamiento de ejecución del subprograma.

2) Efecto de la instrucción:

m modal

s secuencia a secuencia

3) Programabilidad en el programa de pieza:

\+ programable

\- no programable

M solo programable por el fabricante de la máquina

[4] Programabilidad en acciones síncronas:

+	programable
-	no programable
T	programable solo en ciclos tecnológicos

[5] Referencia al documento que contiene la descripción detallada de la instrucción:

FM-A	Manual de funciones Ejes y cabezales
FM-B	Manual de funciones Funciones básicas
FM-SA	Manual de funciones Acciones síncronas
FM-SI	Manual de funciones Safety Integrated
FM-TE	Manual de funciones Tecnologías
FM-TM	Manual de funciones Gestión de herramientas
FM-TR	Manual de funciones Transformadas
PM-MC	Manual de programación Ciclos de medición
PM-NC	Manual de programación Programación CN

[6] Ajuste por defecto al principio del programa (así viene de fábrica el control numérico, si no hay nada diferente programado)

4.2 Direcciones

4.2.1 Letras para direcciones

Letra	Descripción	Extensión numérica
A	Identificador de direcciones ajustable	x
B	Identificador de direcciones ajustable	x
C	Identificador de direcciones ajustable	x
D	Activación/desactivación de corrección de herramienta, filo de la herramienta	
E	Identificador de direcciones ajustable	x
F	Avance Tiempo de espera en segundos	x
G	Comando G	
H	Función H	x
I	Identificador de direcciones ajustable	x
J	Identificador de direcciones ajustable	x
K	Identificador de direcciones ajustable	x
L	Nombre de subprograma, llamada a un subprograma	
M	Función M	x
N	Número de secuencia auxiliar	
O	Libre	
P	Cantidad de pasadas del programa	
Q	Identificador de direcciones ajustable	x
R	Identificador de variables (parámetros R) Identificador de direcciones ajustable (sin extensión numérica)	x
S	Valor de cabezal Tiempo de espera en giros de cabezal	x x
T	Número de herramienta	x
U	Identificador de direcciones ajustable	x
V	Identificador de direcciones ajustable	x
W	Identificador de direcciones ajustable	x
X	Identificador de direcciones ajustable	x
Y	Identificador de direcciones ajustable	x
Z	Identificador de direcciones ajustable	x
%	Carácter inicial y carácter separador para la transferencia de ficheros	
:	Número de secuencia principal	
/	Identificador para secuencia opcional	

4.2.2 Direcciones fijas

Direcciones fijas sin extensión de eje

Identificador de dirección	Tipo de dirección	Modal/ por secuencia	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Tipo de dato del valor asignado
D	Número de corrección	m								x	sin signo INT
F	Avance, tiempo de espera	m, s	x							x	sin signo REAL
G	Comando G	ver lista de las funciones G									sin signo INT
H	Funciones auxiliares	s								x	M: sin signo INT H: REAL
L	Número de subprograma	s									sin signo INT
M	Funciones auxiliares	s								x	M: sin signo INT H: REAL
N	Número de secuencia	s									sin signo INT
OVR	Override	m									sin signo REAL
OVRRAP	Corrección para velocidad en rápido	m									sin signo REAL
P	Cantidad de pasadas del subprograma	s									sin signo INT
S	Cabezal, tiempo de espera	m, s								x	sin signo REAL

Identificador de dirección	Tipo de dirección	Modal/ por secuencia	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Tipo de dato del valor asignado
SCC	Asignación de un eje de refrentado a G96/G961/ G962.	m									REAL
SPOS	Posición del cabezal	m				x	x	x			REAL
SPOSA	Pos. del cabezal salvando límites de secuencia	m				x	x	x			REAL
T	Número de herramienta	m								x	sin signo INT

Direcciones fijas con extensión de eje

Identificador de dirección	Tipo de dirección	Modal/ por secuencia	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Tipo de dato del valor asignado
ACC	Aceleración por eje	m									sin signo REAL
ACCLIMA	Limitación de aceleración por eje esclavo	m									sin signo REAL
AX	Identificador de eje variable	[1]	x	x	x	x	x	x			REAL
FA	Avance por eje	m	x							x	sin signo REAL
FDA	Avance por eje para corrección de volante	s	x								sin signo REAL
FGREF	Radio de referencia	m	x	x							sin signo REAL
FL	Límite para avance por eje	m	x								sin signo REAL
FMA	Avance por eje síncrono	m									sin signo REAL

Identificador de dirección	Tipo de dirección	Modal/ por secuencia	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Tipo de dato del valor asignado
FOC	Desplazamiento secuencial con par limitado	s									REAL
FOCOF	Desplazamiento modal con par limitado desactivado	m									REAL
FOCON	Desplazamiento modal con par limitado activado	m									REAL
FXS	Desplazamiento a tope fijo activado	m									sin signo INT
FXST	Límite de par para desplazamiento a tope fijo	m									REAL
FXSW	Ventana de vigilancia para desplazamiento a tope fijo	m									REAL
IP	Parámetro de interpolación variable	s	x	x	x	x	x				REAL
JERKLIM	Limitación de tirones por eje	m									sin signo REAL
JERKLIMA	Limitación de tirones por eje esclavo	m									sin signo REAL
MEAC	Medición cíclica	s									INT Modo y 1 - 4 eventos de disparo
MEASA	Medición por eje con borrado de trayecto residual	s									INT Modo y 1 - 4 eventos de disparo

Identificador de dirección	Tipo de dirección	Modal/ por secuencia	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Tipo de dato del valor asignado
MEAWA	Medición por eje sin borrado de trayecto residual	s									INT Modo y 1 - 4 eventos de disparo
MOV	Iniciar eje de posicionado	m	x	x	x	x	x	x	x		REAL
OS	Activar/ desactivar vaivén (oscilación)	m									sin signo INT
OSB	Punto inicial oscilación	m	x	x	x	x	x	x			REAL
OSCILL	Asignación de ejes para vaivén, activar vaivén	m									Axis: 1 - 3 ejes de penetración
OSCTRL	Opciones de vaivén	m									sin signo INT: opciones de activación, sin signo INT: Opciones de desactivación
OSE	Punto final vaivén	m	x	x	x	x	x	x			REAL
OSNSC	Número de ciclos de afinado vaivén	m									sin signo INT
OSP1	Punto de inversión izquierdo (vaivén)	m	x	x	x	x	x	x			REAL
OSP2	Punto de inversión derecho (vaivén)	m	x	x	x	x	x	x			REAL
OST1	Tiempo de parada en pto. de inversión izquierdo (vaivén)	m									REAL
OST2	Tiempo de parada en pto. de inversión derecho (vaivén)	m									REAL
OVRA	Corrección por eje	m	x								sin signo REAL

Identificador de dirección	Tipo de dirección	Modal/ por secuencia	G70/ G71	G700/ G710	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	QU	Tipo de dato del valor asignado
PO	Coeficiente de polinomio	s	x	x		x	x	x			sin signo REAL
POLF	Posicionar LIFTFAST	m	x	x							sin signo REAL
POS	Eje de posicionado	m	x	x	x	x	x	x	x		REAL
POSA	Eje de posicionado salvando límites de secuencia	m	x	x	x	x	x	x	x		REAL
POSP	Posicionado en tramos (vaivén)	m	x	x	x	x	x	x			REAL: posición final Real: longitud parcial INT: Opción
STA	Tiempo de afinado (por eje)	m									sin signo REAL
SRA	Trayecto de retroceso con entrada externa (por eje)	m									sin signo REAL
VELOLIM	Limitación de velocidad por eje	m									sin signo REAL
VELOLIMA	Limitación de velocidad por eje esclavo	m									sin signo REAL

[1] Puntos finales absolutos: modal; puntos finales incrementales: por secuencia, o bien modal/por secuencia en función del comando G que define la sintaxis

4.2.3 Direcciones ajustables

Identificador de dirección (ajuste estándar)	Tipo de dirección	modal/ por secuencia	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Cant. máx.	Tipo de dato del valor asignado
Valores de ejes y puntos finales											
X, Y, Z, A, B, C	Eje	[1]	x	x	x	x				8	REAL
AP	Ángulo polar	m/s [1]	x	x	x					1	REAL
RP	Radio polar	m/s [1]	x	x	x					1	sin signo REAL
Orientación de la herramienta											
A2, B2, C2	Ángulo euleriano o ángulo RPY	s								3	REAL
A3, B3, C3	Componentes del vector de dirección	s								3	REAL
A4, B4, C4	Componentes del vector normal a la superficie al inicio de la secuencia	s								3	REAL
A5, B5, C5	Componentes del vector normal a la superficie al final de la secuencia	s								3	REAL
A6, B6, C6	Componentes del vector de dirección para el eje de giro del cono	s								3	REAL
A7, B7, C7	Componentes de vector para la orientación intermedia en la superficie del cono	s								3	REAL
LEAD	Ángulo de avance	m								1	REAL
THETA	Ángulo de giro - giro en la dirección de herramienta	m		x	x					1	REAL
TILT	Ángulo lateral	m								1	REAL

Identificador de dirección (ajuste estándar)	Tipo de dirección	modal/ por secuencia	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Cant. máx.	Tipo de dato del valor asignado
ORIS	Modificación de la orientación (referido a la trayectoria)	m								1	REAL
Parámetro de interpolación											
I, J, K	Parámetros de interpolación Coordenada del punto intermedio	s		x [2)]	x [2)]					3	REAL
I1, J1, K1		s	x	x	x					3	REAL
RPL	Rotación en el plano	s								1	REAL
CR	Radio del círculo	s								1	sin signo REAL
AR	Ángulo en el vértice	s								1	sin signo REAL
TURN	N.º de vueltas para hélices o espirales	s								1	sin signo INT
PL	Longitud de intervalo de parámetros	s								1	sin signo REAL
PW	Peso del punto	s								1	sin signo REAL
SD	Grado del spline	m								1	sin signo INT
TU	Ángulo del eje	s								1	sin signo INT
STAT	Posición de las articulaciones	m								1	sin signo INT
SF	Decalaje del punto inicial para el tallado de roscas	m								1	REAL
DISCL	Distancia de seguridad WAB	s								1	sin signo REAL
DISR	Distancia repos./ distancia WAB	s								1	sin signo REAL
DISPR	Diferencia trayectoria reposicionamiento	s								1	sin signo REAL
ALF	Ángulo de retirada rápida	m								1	sin signo INT
DILF	Long. de retirada rápida	m								1	REAL

Identificador de dirección (ajuste estándar)	Tipo de dirección	modal/ por secuencia	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Cant. máx.	Tipo de dato del valor asignado
FP	Punto fijo: Número del punto fijo que se debe alcanzar	s								1	sin signo INT
RNDM	Redondeo modal	m								1	sin signo REAL
RND	Redondeo por secuencia	s								1	sin signo REAL
CHF	Chaflán por sec.	s								1	sin signo REAL
CHR	Fase en el sentido original de desplazamiento	s								1	sin signo REAL
ANG	Ángulo de sucesión de contorno	s								1	REAL
ISD	Profundidad de penetración	m								1	REAL
DISC	Rebase circunferencia de transición; corrección de radio de herramienta	m								1	sin signo REAL
OFFN	Offset contorno normal	m								1	REAL
DITS	Trayecto de entrada en rosca	m								1	REAL
DITE	Trayecto de salida de rosca	m								1	REAL
Criterios de matado de esquinas											
ADIS	Distancia de matado de esquinas	m								1	sin signo REAL
ADISPOS	Distancia de matado de esquinas en rápido	m								1	sin signo REAL
Medición											
MEAS	Medida con palpador de contacto	s								1	sin signo INT
MEAW	Medición con palpador de contacto sin borrado de trayecto residual	s								1	sin signo INT

Identificador de dirección (ajuste estándar)	Tipo de dirección	modal/ por secuencia	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Cant. máx.	Tipo de dato del valor asignado
Comportamiento de ejes, cabezales											
LIMS	Limitación de la velocidad del cabezal	m								1	sin signo REAL
COARSEA	Comportamiento en cambio de secuencia: Parada precisa basta de eje	m									
FINEA	Comportamiento en cambio de secuencia: Parada precisa fina de eje	m									
IPOENDA	Comportamiento en cambio de secuencia: Parada interpolador de eje	m									
DIACYCOFA	Eje de refrentado: programación por diámetro de eje desactivada en ciclos	m									
DIAM90A	Eje de refrentado: Programación por diámetro de eje en G90	m									
DIAMCHAN	Eje de refrentado: Aplicación de todos los ejes de refrentado en el estado del canal de la programación por diámetro	m									
DIAMCHANA	Eje de refrentado: Aplicación del estado del canal de la programación por diámetro	m									

Identificador de dirección (ajuste estándar)	Tipo de dirección	modal/ por secuencia	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Cant. máx.	Tipo de dato del valor asignado
DIAMOFA	Eje de refrentado: Programación por diámetro de eje desactivada	m									
DIAMONA	Eje de refrentado: Programación por diámetro de eje activada	m									
GP	Posición: Programación indirecta de atributos de posición	m									
Avances											
FAD	Velocidad del desplazamiento de penetración lento	s						x		1	sin signo REAL
FD	Avance de contorneado para corrección por volante	s								1	sin signo REAL
FRC	Avance para el radio y chaflán	s								1	sin signo REAL
FRCM	Avance modal para radio y chaflán	m								1	sin signo REAL
FB	Avance por secuencia	s								1	sin signo REAL
Punzonado/troquelado											
SPN	Cantidad de trayectos parciales por secuencia	s								1	INT
SPP	Longitud de un trayecto parcial	m								1	REAL
Rectificado											
ST	Tiempo de afinado	s								1	sin signo REAL
SR	Distancia de retirada	s								1	sin signo REAL
Selección de herramienta											

Identificador de dirección (ajuste estándar)	Tipo de dirección	modal/ por secuencia	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Cant. máx.	Tipo de dato del valor asignado
TCARR	Portaherramientas	m								1	INT
Gestión de herramientas											
DL	Número de corrección de herramienta	m								1	INT
Direcciones OEM											
OMA1	Dirección OEM 1	m		x	x	x				1	REAL
OMA2	Dirección OEM 2	m		x	x	x				1	REAL
OMA3	Dirección OEM 3	m		x	x	x				1	REAL
OMA4	Dirección OEM 4	m		x	x	x				1	REAL
OMA5	Dirección OEM 5	m		x	x	x				1	REAL
Varios											
CUTMOD	Modificación de los datos de corrección en herramientas giratorias (en combinación con portaherramientas orientables)	m									INT
CUTMODK	Modificación de los datos de corrección en herramientas giratorias (en combinación con transformadas de orientación definidas mediante cadenas cinemáticas)	m									STRING
TOFF	Offset de longitud de herramienta paralelo al eje geométrico indicado	m									REAL

Identificador de dirección (ajuste estándar)	Tipo de dirección	modal/ por secuencia	G90/ G91	IC	AC	DC, ACN, ACP	CIC, CAC, CDC, CACN, CACP	PR, PM	QU	Cant. máx.	Tipo de dato del valor asignado
TOFFL	Offset de longitud de herramienta en la dirección de la componente longitudinal de herramienta L1, L2 o L3	m									REAL
TOFFR	Offset del radio de herramienta	m									REAL
TOFFLR	Offset simultáneo de longitud de herramienta y radio de herramienta	m									REAL

1) Puntos finales absolutos: modal; puntos finales incrementales: por secuencia, o bien modal/por secuencia en función del comando G que determina la sintaxis

2) Para el centro de la circunferencia se tienen en cuenta parámetros IPO incrementales. En el control adaptativo se pueden programar de forma absoluta. Con otros significados (p. ej.: paso de rosca) se ignora la modificación de dirección

4.3 Comandos G

4.3.1 Comandos G

Los comandos G se dividen en grupos G. En los programas de pieza o en acciones síncronas, en una secuencia solo puede escribirse un comando G de un grupo G. Un comando G puede tener una validez modal o secuencial.

Modal: hasta la programación de otro comando G del mismo grupo G.

4.3.2 Grupo G 1: Comandos de desplazamiento modales

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G0	1	Desplazamiento en rápido	+	m		
G1	2	Interpolación lineal (interpolación lineal)	+	m	x	
G2	3	Interpolación circular en sentido horario	+	m		
G3	4	Interpolación circular en sentido antihorario	+	m		
CIP	5	Interpolación circular a través de punto intermedio	+	m		
ASPLINE	6	Spline de Akima	+	m		
BSPLINE	7	Spline B	+	m		
CSPLINE	8	Spline cúbico	+	m		
POLY	9	Interpolación de polinomios	+	m		
G33	10	Tallado de roscas con paso constante	+	m		
G331	11	Roscado	+	m		
G332	12	Retroceso (roscado)	+	m		
OEMIPO1	13	Reservado	+	m		
OEMIPO2	14	Reservado	+	m		
CT	15	Arco con transición tangencial	+	m		
G34	16	Tallado de roscas con paso linealmente creciente	+	m		
G35	17	Tallado de roscas con paso linealmente decreciente	+	m		
INVCW	18	Interpolación de evolutas en sentido horario	+	m		
INVCCW	19	Interpolación de evolutas en sentido antihorario	+	m		
G335	20	Torneado de una rosca convexa en sentido horario	+	m		
G336	21	Torneado de una rosca convexa en sentido antihorario	+	m		

4.3.3 Grupo G 2: Desplazamientos por secuencia, tiempo de espera

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G4	1	Tiempo de espera, temporizado	-	s		
G63	2	Roscado sin sincronización	-	s		
G74	3	Búsqueda del punto de referencia con sincronización	-	s		
G75	4	Desplazamiento a punto fijo	-	s		
REPOSL	5	Reposicionamiento en el contorno lineal	-	s		
REPOSQ	6	Reposicionamiento en el contorno en cuadrante	-	s		
REPOSH	7	Reposicionamiento en el contorno en semicircunferencia	-	s		
REPOSA	8	Reposicionamiento en el contorno lineal con todos los ejes	-	s		
REPOSQA	9	Reposicionamiento en el contorno con todos los ejes; ejes geométricos en cuadrante	-	s		
REPOSHA	10	Reposicionamiento en el contorno con todos los ejes; ejes geométricos en semicírculo	-	s		
G147	11	Aproximación al contorno siguiendo una recta	-	s		
G247	12	Aproximación al contorno siguiendo un cuadrante de circunferencia	-	s		
G347	13	Aproximación al contorno siguiendo una semicircunferencia	-	s		
G148	14	Retirada del contorno siguiendo una recta	-	s		
G248	15	Retirada del contorno siguiendo un cuadrante de circunferencia	-	s		
G348	16	Retirada del contorno siguiendo una semicircunferencia	-	s		
G5	17	Rectificado oblicuo de ranuras	-	s		
G7	18	Movimiento de compensación en el rectificado oblicuo de ranuras	-	s		

4.3.4 Grupo G 3: Frame programable, limitación del campo de trabajo y programación del polo

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
TRANS	1	TRANSLATION: Decalaje programable	-	s		
ROT	2	ROTATION: Rotación programable	-	s		
SCALE	3	SCALE: Escala programable	-	s		
MIRROR	4	MIRROR: Simetría programable	-	s		
ATRANS	5	Additive TRANSLATION: Decalaje aditivo programable	-	s		
AROT	6	Additive ROTATION: Rotación programable	-	s		
ASCALE	7	Additive SCALE: Escala programable	-	s		

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
AMIRROR	8	Additive MIRROR: Simetría programable	-	s		
-	9	Libre	-	-		
G25	10	Límite inferior para zona de trabajo/limitación de velocidad del cabezal	-	s		
G26	11	Límite superior para zona de trabajo/limitación de velocidad del cabezal	-	s		
G110	12	Programación de polo relativa a la última posición nominal programada	-	s		
G111	13	Programación del polo respecto al origen del sistema de coordenadas actual de la pieza	-	s		
G112	14	Programación de polo relativa al último polo activo	-	s		
G58	15	Decalaje de origen programable absoluto	-	s		
G59	16	Decalaje de origen programable aditivo	-	s		
ROTS	17	Rotación con ángulo espacial	-	s		
AROTS	18	Rotación aditiva con ángulo espacial	-	s		

4.3.5 Grupo G 4: FIFO

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
STARTFIFO	1	Start FIFO Mecanizado y paralelamente llenado de la memoria (búfer) de pretratamiento	+	m	x	
STOPFIFO	2	STOP FIFO, parada del mecanizado; llenado de la memoria de pretratamiento hasta que se reconozca STARTFIFO, memoria de pretratamiento llena o fin de programa	+	m		
FIFOCTRL	3	Activación del control automático de la memoria (búfer) de pretratamiento	+	m		

4.3.6 Grupo G 6: Selección del plano

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G17	1	Selección del plano del 1.er - 2.º eje geométrico	+	m	x	
G18	2	Selección del plano del 3.er - 1.er eje geométrico	+	m		
G19	3	Selección del plano del 2.º - 3.er eje geométrico	+	m		

4.3.7 Grupo G 7: Corrección de radio de herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G40	1	No se corrige el radio de la herramienta	+	m	x	
G41	2	Corrección del radio de la herramienta a la izquierda del contorno	-	m		
G42	3	Corrección del radio de la herramienta a la derecha del contorno	-	m		

4.3.8 Grupo G 8: Decalaje de origen ajustable

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G500	1	Desactivación del decalaje de origen ajustable (G54 ... G57, G505 ... G599)	+	m	x	
G54	2	1.er decalaje de origen ajustable	+	m		
G55	3	2.º decalaje de origen ajustable	+	m		
G56	4	3.er decalaje de origen ajustable	+	m		
G57	5	4.º decalaje de origen ajustable	+	m		
G505	6	5.º decalaje de origen ajustable	+	m		
...	...	...	+	m		
G599	100	99.º decalaje de origen ajustable	+	m		
Con los distintos comandos G de este grupo G se activa un frame de usuario ajustable $P_UIFR[ ]. G54 equivale al frame $P_UIFR[1], G505 equivale al frame $P_UIFR[5]. El número de frames de usuario ajustables y, por tanto, el número de comandos G de este grupo G puede parametrizarse a través del dato de máquina MD28080 $MC_MM_NUM_USER_FRAMES.						

4.3.9 Grupo G 9: Supresión de la corrección de frames y de herramientas

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G53	1	Supresión de los frames actuales: frame programable, incluido frame de sistema para TOROT y TOFRAME y frame ajustable activo (G54 ... G57, G505 ... G599).	-	s		
SUPA	2	Como G153, incluida la supresión de los frames de sistema para poner valor real, aproximación con contacto, decalaje de origen externo, PAROT con decalajes con volante (DRF), [decalaje de origen externo], movimiento superpuesto	-	s		
G153	3	Como G53, incluida la supresión de todos los frames básicos específicos del canal o globales de NCU	-	s		
SUPD	4	Supresión de las correcciones de herramienta activas	-	s		

4.3.10 Grupo G 10: Parada precisa - Modo Contorneado

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G60	1	Parada precisa	+	m	x	
G64	2	Modo de contorneado	+	m		
G641	3	Modo de contorneado con matado de esquinas según criterio de trayectoria (= distancia de matado programable)	+	m		
G642	4	Modo de contorneado con matado de esquinas respetando las tolerancias definidas	+	m		
G643	5	Modo de contorneado con matado de esquinas respetando las tolerancias definidas (internas de la secuencia)	+	m		
G644	6	Modo de contorneado con matado de esquinas con la máxima dinámica posible	+	m		
G645	7	Modo de contorneado con matado de esquinas y transiciones de secuencia tangenciales respetando las tolerancias definidas	+	m		

4.3.11 Grupo G 11: Parada precisa por secuencia

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G9	1	Parada precisa	-	s		

4.3.12 Grupo G 12: Criterios de cambio de secuencia para parada precisa (G60/G9)

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G601	1	Cambio de secuencia con parada precisa fina	+	m	x	
G602	2	Cambio de secuencia con parada precisa basta	+	m		
G603	3	Cambio de secuencia para interpolador - fin de secuencia	+	m		

4.3.13 Grupo G 13: Acotado de piezas en métrico/pulgadas

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G70	1	Sistema de unidades en pulgadas (longitudes)	+	m		
G71	2	Sistema de unidades métrico en mm (longitudes)	+	m	x	
G700	3	Sistema de unidades en pulgadas, pulgadas/min (longitudes + velocidad + variable de sistema)	+	m		
G710	4	Sistema de unidades métrico en mm, mm/min (longitudes + velocidad + variable de sistema)	+	m		

4.3.14 Grupo G 14: Acotado de piezas absoluto/incremental

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G90	1	Acotado absoluto	+	m	x	
G91	2	Acotado incremental	+	m		

4.3.15 Grupo G 15: Tipo de avance

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G93	1	Avance inverso al tiempo en 1/min (rpm)	+	m		
G94	2	Avance lineal en mm/min, pulgadas/min	+	m	x	
G95	3	Avance por vuelta en mm/vuelta pulgadas/vuelta	+	m		
G96	4	Avance por vuelta (como con G95) y velocidad de corte constante	+	m		
G97	5	Avance por vuelta y velocidad de giro del cabezal constante (velocidad de corte constante DES)	+	m		
G931	6	Especificación del avance mediante el tiempo de desplazamiento, desactivar la velocidad de contorneado constante	+	m		
G961	7	Avance lineal (como con G94) y velocidad de corte constante	+	m		
G971	8	Avance lineal y velocidad de giro del cabezal constante (velocidad de corte constante DES)	+	m		
G942	9	Avance lineal y velocidad de corte constante o velocidad del cabezal constante	+	m		
G952	10	Avance por vuelta y velocidad de corte constante o velocidad del cabezal constante	+	m		
G962	11	Avance lineal o avance por vuelta y velocidad de corte constante	+	m		
G972	12	Avance lineal o avance por vuelta y velocidad de giro del cabezal constante (velocidad de corte constante DES)	+	m		
G973	13	Avance por vuelta sin limitación de la velocidad de giro del cabezal y velocidad de giro del cabezal constante (G97 sin LIMS para modo ISO)	+	m		

4.3.16 Grupo G 16: Corrección de avance en curvatura interna y externa

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CFC	1	Avance constante en el contorno activo con curvatura interna y externa	+	m	x	
CFTCP	2	Avance constante en el punto de referencia del filo de herramienta (trayectoria del centro)	+	m		
CFIN	3	Avance constante con curvatura interna, aceleración con curvatura externa	+	m		

4.3.17 Grupo G 17: Comportamiento en aproximación y retirada, corrección de herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
NORM	1	Posición de normales en los puntos inicial y final	+	m	x	
KONT	2	Rodear el contorno en los puntos inicial y final	+	m		
KONTT	3	Aproximación/retirada con tangente continua	+	m		
KONTC	4	Aproximación/retirada de curvatura continua	+	m		

4.3.18 Grupo G 18: Comportamiento en esquinas, corrección de herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G450	1	Circunferencia de transición (la herramienta pasa por las esquinas de la pieza siguiendo una trayectoria circular)	+	m	x	
G451	2	Punto de intersección de las equidistantes (la herramienta sale en el ángulo de la pieza)	+	m		

4.3.19 Grupo G 19: Transición de curva al inicio de Spline

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
BNAT	1	Transición de curva natural a la primera secuencia spline	+	m	x	
BTAN	2	Transición de curva tangencial a la primera secuencia spline	+	m		
BAUTO	3	Definición del primer segmento spline mediante los 3 puntos siguientes	+	m		

4.3.20 Grupo G 20: Transición de curva al final de Spline

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ENAT	1	Transición de curva natural a la siguiente secuencia de desplazamiento	+	m	x	
ETAN	2	Transición de curva tangencial a la siguiente secuencia de desplazamiento	+	m		
EAUTO	3	Definición del último segmento spline mediante los 3 últimos puntos	+	m		

4.3.21 Grupo G 21: Perfil de aceleración

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
BRISK	1	Aceleración de contorneado de forma escalonada	+	m	x	
SOFT	2	Aceleración sin tirones en la trayectoria	+	m		
DRIVE	3	Aceleración de contorneado dependiente de la velocidad	+	m		

4.3.22 Grupo G 22: Tipo de corrección de herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CUT2D	1	2½-D-WRK	+	m	x	
CUT2DF	2	2½-D-WRK, relativa al frame actual (plano inclinado)	+	m		
CUT3DC	3	3D-WRK para fresado periférico	+	m		
CUT3DF	4	3D-WRK para fresado frontal con cambio de orientación	+	m		
CUT3DFS	5	3D-WRK para fresado frontal con orientación constante. La orientación de herramienta se define con G17 - G19 y no depende de los frames.	+	m		
CUT3DFF	6	3D-WRK para fresado frontal con orientación constante. La orientación de la herramienta es la dirección definida con G17 - G19 y, dado el caso, girada por un frame.	+	m		
CUT3DCC	7	3D-WRK para el fresado periférico considerando una superficie de limitación con corrección de radio 3D: Contorno en la superficie de mecanizado	+	m		
CUT3DCCD	8	3D-WRK para fresado periférico considerando una superficie de limitación con herramienta diferencial en la trayectoria del centro de la herramienta: Penetración en la superficie de limitación	+	m		

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CUT2DD	9	2½-D-WRK referida a una herramienta diferencial	+	m		
CUT2DFD	10	2½-D-WRK referida a una herramienta diferencial, relativa al frame actual (plano inclinado)	+	m		
CUT3DCD	11	3D-WRK referida a una herramienta diferencial para fresado periférico	+	m		
CUT3DFD	12	3D-WRK referida a una herramienta diferencial para fresado frontal con cambio de orientación	+	m		

4.3.23 Grupo G 23: Vigilancia de colisión en contornos internos

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CDOF	1	Vigilancia de colisión DES	+	m	x	
CDON	2	Vigilancia de colisión CON	+	m		
CDOF2	3	Vigilancia de colisión DES con fresado periférico 3D	+	m		

4.3.24 Grupo G 24: Control anticipativo

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
FFWOF	1	Mando anticipativo DES	+	m	x	
FFWON	2	Mando anticipativo CON	+	m		

4.3.25 Grupo G 25: Relación orientación de herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIWKS	1	Orientación de herramienta en el sistema de coordenadas de pieza (WKS)	+	m	x	
ORIMKS	2	Orientación de herramienta en el sistema de coordenada de máquina (MKS)	+	m		

4.3.26 Grupo G 26: Modo de reposicionamiento para REPOS (de efecto modal)

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
RMB	1	Reposicionamiento en el punto inicial de la secuencia	-	m		
RMI	2	Reposicionamiento en el punto de interrupción	-	m	x	
RME	3	Reposicionamiento en el punto final de la secuencia	-	m		
RMN	4	Reposicionamiento en el punto de trayectoria más cercano	-	m		

4.3.27 Grupo G 27: Corrección de herramienta en cambios de orientación en esquinas exteriores

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIC	1	Cambios de orientación en esquinas exteriores se superponen a la secuencia circular a insertar	+	m	x	
ORID	2.	Cambios de orientación se ejecutan antes de la secuencia circular	+	m		

4.3.28 Grupo G 28: Limitación del campo de trabajo

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
WALIMON	1	Limitación del campo de trabajo CON	+	m	x	
WALIMOF	2	Limitación del campo de trabajo DES	+	m		

4.3.29 Grupo G 29: Programación por radio/diámetro

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
DIAMOF	1	Programación por diámetro específica del canal modalmente activa DES Con la desactivación se hace efectiva la programación por radio específica del canal.	+	m	x	
DIAMON	2	Programación por diámetro independiente específica del canal modalmente activa CON El efecto es independiente del modo de acotado programado (G90/G91).	+	m		

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
DIAM90	3	Programación por diámetro dependiente específica del canal modalmente activa CON El efecto depende del modo de acotado programado (G90/G91).	+	m		
DIAMCYCOF	4	Programación por diámetro específica del canal modalmente activa durante la ejecución de ciclos DES	+	m		

4.3.30 Grupo G 30: Compresión de secuencias CN

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
COMPOF	1	Compresión de secuencias CN DES	+	m	x	
COMPON	2	Función de compresor COMPON CON	+	m		
COMPCURV	3	Función de compresor COMPCURV CON	+	m		
COMPCAD	4	Función de compresor COMPCAD CON	+	m		
COMPSURF	5	Función de compresor COMPSURF CON	+	m		

4.3.31 Grupo G 31: Comandos G OEM

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G810	1	Comando G OEM	-	m		
G811	2	Comando G OEM	-	m		
G812	3	Comando G OEM	-	m		
G813	4	Comando G OEM	-	m		
G814	5	Comando G OEM	-	m		
G815	6	Comando G OEM	-	m		
G816	7	Comando G OEM	-	m		
G817	8	Comando G OEM	-	m		
G818	9	Comando G OEM	-	m		
G819	10	Comando G OEM	-	m		
Se reservan dos grupos G para el usuario OEM. De esta manera se habilita la programación de las funciones implementadas.						

4.3.32 Grupo G 32: Comandos G OEM

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G820	1	Comando G OEM	-	m		
G821	2	Comando G OEM	-	m		
G822	3	Comando G OEM	-	m		
G823	4	Comando G OEM	-	m		
G824	5	Comando G OEM	-	m		
G825	6	Comando G OEM	-	m		
G826	7	Comando G OEM	-	m		
G827	8	Comando G OEM	-	m		
G828	9	Comando G OEM	-	m		
G829	10	Comando G OEM	-	m		
Se reservan dos grupos G para el usuario OEM. De esta manera se habilita la programación de las funciones implementadas.						

4.3.33 Grupo G 33: Corrección de herramienta fina ajustable

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
FTOCOF	1	Corrección de herramienta fina activa online DES	+	m	x	
FTOCON	2	Corrección de herramienta fina activa online CON	-	m		

4.3.34 Grupo G 34: Alisado de la orientación de herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
OSOF	1	Alisado de la orientación de herramienta DES	+	m	x	
OSC	2	Alisado de la orientación de herramienta constante	+	m		
OSS	3	Alisado de la orientación de herramienta al final de la secuencia	+	m		
OSSE	4	Alisado de la orientación de herramienta al principio y al final de la secuencia	+	m		
OSD	5	Matado de esquinas interno de la secuencia con especificación de la longitud del trayecto	+	m		
OST	6	Matado de esquinas interno de la secuencia con especificación de la tolerancia angular	+	m		

4.3.35 Grupo G 35: Troquelado y punzonado

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
SPOF	1	Carrera DES, troquelado y punzonado DES	+	m	x	
SON	2	Punzonado CON	+	m		
PON	3	Troquelado CON	+	m		
SONS	4	Punzonado CON en el ciclo IPO	-	m		
PONS	5	Troquelado CON en el ciclo IPO	-	m		

4.3.36 Grupo G 36: Troquelado con retardo

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
PDELAYON	1	Retardo en el troquelado CON	+	m	x	
PDELAYOF	2	Retardo en el troquelado DES	+	m		

4.3.37 Grupo G 37: Perfil de avance

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
FNORM	1	Avance normal según DIN66025	+	m	x	
FLIN	2	Avance lineal variable	+	m		
FCUB	3	Avance variable según spline cúbico	+	m		

4.3.38 Grupo G 38: Asignación de entradas/salidas rápidas para punzonado/troquelado

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
SPIF1	1	Entradas/salidas CN rápidas para troquelado/punzonado, byte 1	+	m	x	
SPIF2	2	Entradas/salidas CN rápidas para troquelado/punzonado, byte 2	+	m		

4.3.39 Grupo G 39: Precisión de contorno programable

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CPRECOF	1	Precisión de contorno programable DES	+	m	x	
CPRECON	2	Precisión de contorno programable CON	+	m		

4.3.40 Grupo G 40: Corrección de radio de corte/herramienta constante

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CUTCONOF	1	Corrección constante del radio de herramienta DES	+	m	x	
CUTCONON	2	Corrección constante del radio de herramienta CON	+	m		

4.3.41 Grupo G 41: Tallado de roscas con posibilidad de interrupción

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
LFOF	1	Tallado de roscas con posibilidad de interrupción DES	+	m	x	
LFON	2	Tallado de roscas con posibilidad de interrupción CON	+	m		

4.3.42 Grupo G 42: Portaherramientas

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
TCOABS	1	Calcular las componentes longitudinales de la herramienta a partir de la orientación actual de la herramienta	+	m	x	
TCOFR	2	Determinar las componentes longitudinales de la herramienta a partir de la orientación del frame activo	+	m		
TCOFRZ	3	Determinar la orientación de la herramienta de un frame activo en la selección de herramientas; la herramienta apunta en dirección Z	+	m		
TCOFRY	4	Determinar la orientación de la herramienta de un frame activo en la selección de herramientas; la herramienta apunta en dirección Y	+	m		
TCOFRX	5	Determinar la orientación de la herramienta de un frame activo en la selección de herramientas; la herramienta apunta en dirección X		m		

4.3.43 Grupo G 43: Dirección de aproximación/retirada suaves

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G140	1	Dirección para aprox./retirada suaves del contorno definida mediante G41/G42	+	m	x	
G141	2	Dirección de aprox./retirada suaves a la izquierda del contorno	+	m		
G142	3	Dirección de aprox./retirada suaves a la derecha del contorno	+	m		
G143	4	Dirección de aprox./retirada suaves del contorno dependiente de la tangente	+	m		

4.3.44 Grupo G 44: División de segmentos aprox./retirada suaves

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G340	1	Secuencia de desplazamiento en el espacio, es decir, penetración en profundidad y aproximación en el plano dentro de una secuencia	+	m	x	
G341	2	Primero, penetración en el eje perpendicular (Z); después, aproximación en el plano	+	m		

4.3.45 Grupo G 45: Trayectoria de referencia de los ejes FGROUP

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
SPATH	1	La trayectoria de referencia para los ejes FGROUP es la longitud de un arco	+	m	x	
UPATH	2	La trayectoria de referencia para los ejes FGROUP es el parámetro de curva	+	m		

4.3.46 Grupo G 46: Selección de planos para retirada rápida

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
LFTXT	1	El plano se determina a partir de la tangente de trayectoria y de la orientación de herramienta actual	+	m	x	
LFWP	2	El plano viene determinado por el plano de trabajo actual (G17/G18/G19)	+	m		
LFPOS	3	Levantamiento axial a una posición	+	m		

4.3.47 Grupo G 47: Conmutación de modo para código CN externo

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G290	1	Activación del modo de lenguaje SINUMERIK	+	m	x	
G291	2	Activación del modo de lenguaje ISO	+	m		

4.3.48 Grupo G 48: Comportamiento en aproximación y retirada en la corrección de radio de herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
G460	1	Vigilancia de colisión para la secuencia de aproximación y retirada CON	+	m	x	
G461	2	Si no existe ningún punto de intersección en la secuencia de corrección del radio de herramienta, prolongar la secuencia de borde con un arco de circunferencia	+	m		
G462	3	Si no existe ningún punto de intersección en la secuencia de corrección del radio de herramienta, prolongar la secuencia de borde con una recta	+	m		

4.3.49 Grupo G 49: Desplazamiento punto a punto

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
CP	1	Desplazamiento de trayectoria	+	m	x	
PTP	2	Desplazamiento punto a punto (movimiento de eje síncrono)	+	m		

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
PTPG0	3	Desplazamiento punto a punto sólo con G0; si no, desplazamiento de trayectoria CP	+	m		
PTPWOC	4	Movimiento punto a punto sin movimientos de compensación provocados por modificaciones de la orientación	+	m		

4.3.50 Grupo G 50: Programación de la orientación de la herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIEULER	1	Ángulo de orientación mediante ángulos eulerianos	+	m	x	
ORIRPY	2	Ángulo de orientación mediante ángulos RPY (secuencia de giro XYZ)	+	m		
ORIVIRT1	3	Ángulo de orientación mediante ejes de orientación virtuales (definición 1)	+	m		
ORIVIRT2	4	Ángulo de orientación mediante ejes de orientación virtuales (definición 2)	+	m		
ORIAXPOS	5	Ángulo de orientación a través de ejes de orientación virtuales con posiciones de eje giratorio	+	m		
ORIRPY2	6	Ángulo de orientación mediante ángulos RPY (secuencia de giro ZYX)	+	m		

4.3.51 Grupo G 51: Tipo de interpolación en la programación de la orientación

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIVECT	1	Interpolación circular de gran radio (idéntico a ORIPLANE)	+	m	x	
ORIAXES	2	Interpolación lineal de los ejes de máquina o ejes de orientación	+	m		
ORIPATH	3	Ruta de orientación de la herramienta con relación a la trayectoria	+	m		
ORIPLANE	4	Interpolación en el plano (idéntico a ORIVECT)	+	m		
ORICONCW	5	Interpolación en una superficie de cono en sentido horario	+	m		
ORICONCCW	6	Interpolación en una superficie de cono en sentido antihorario	+	m		
ORICONIO	7	Interpolación en una superficie de cono con indicación de una orientación intermedia	+	m		
ORICONTO	8	Interpolación en una superficie envolvente de cono en la transición tangencial	+	m		

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORICURVE	9	Interpolación con curva espacial adicional para la orientación	+	m		
ORIPATHS	10	Orientación de la herramienta con relación a la trayectoria; se suaviza un acodamiento en el desarrollo de la orientación	+	m		

4.3.52 Grupo G 52: Rotación de frame relativa a la pieza

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
PAROTOF	1	Rotación de frame relativa a la pieza DES	+	m	x	
PAROT	2	Rotación de frame relativa a la pieza CON El sistema de coordenadas de pieza se orienta respecto a la pieza.	+	m		

4.3.53 Grupo G 53: Rotación de frame relativa a la herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
TOROTOF	1	Rotación de frame relativa a la herramienta DES	+	m	x	
TOROT	2	Orientar el eje Z del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	+	m		
TOROTZ	3	Como TOROT	+	m		
TOROTY	4	Orientar el eje Y del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	+	m		
TOROTX	5	Orientar el eje X del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	+	m		
TOFRAME	6	Orientar el eje Z del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	+	m		
TOFRAMEZ	7	Como TOFRAME	+	m		
TOFRAMEY	8	Orientar el eje Y del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	+	m		
TOFRAMEX	9	Orientar el eje X del sistema de coordenadas de pieza paralelamente a la orientación de la herramienta mediante rotación del frame	+	m		

4.3.54 Grupo G 54: Rotación vectorial en la programación polinómica

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORIROTA	1	Rotación vectorial absoluta	+	m	x	
ORIROTR	2	Rotación vectorial relativa	+	m		
ORIROTT	3	Rotación vectorial tangencial	+	m		
ORIROTC	4	Vector de giro tangencial a la trayectoria tangente	+	m		

4.3.55 Grupo G 55: Desplazamiento en rápido con/sin interpolación lineal

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
RTLION	1	Desplazamiento en rápido con interpolación lineal CON	+	m	x	
RTLIOF	2	Desplazamiento en rápido con interpolación lineal DES El desplazamiento en rápido se lleva a cabo con interpolación de ejes individuales.	+	m		

4.3.56 Grupo G 56: Inclusión del desgaste de la herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
TOWSTD	1	Ajuste básico para correcciones en la longitud de la herramienta	+	m	x	
TOWMCS	2	Valores de desgaste en el sistema de coordenadas de máquina (MKS)	+	m		
TOWWCS	3	Valores de desgaste en el sistema de coordenadas de pieza (WKS)	+	m		
TOWBCS	4	Valores de desgaste en el sistema de coordenadas básico (BKS)	+	m		
TOWTCS	5	Valores de desgaste en el sistema de coordenadas de herramienta (punto de referencia de portaherramientas T en el alojamiento del portaherramientas)	+	m		
TOWKCS	6	Valores de desgaste en el sistema de coordenadas del cabezal de herramienta con transformación cinética (se distingue del MKS por el giro de la herramienta)	+	m		

4.3.57 Grupo G 57: Deceleración en los dos vértices

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
FENDNORM	1	Deceleración en los dos vértices DES	+	m	x	
G62	2	Deceleración en los dos vértices en esquinas interiores con corrección activa del radio de herramienta (G41/ G42)	+	m		
G621	3	Deceleración en los dos vértices en todas las esquinas	+	m		

4.3.58 Grupo G 59: Modo dinámico para interpolación de la trayectoria

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
DYNNORM	1	Dinámica normal	+	m	x	
DYNPOS	2	Modo de posicionado, roscado	+	m		
DYNROUGH	3	Desbaste	+	m		
DYNSEMIFIN	4	Preacabado	+	m		
DYNFINISH	5	Acabado	+	m		
DYNPREC	6	Acabado fino	+	m		

4.3.59 Grupo G 60: Limitación del campo de trabajo

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
WALCS0	1	Limitación del campo de trabajo WKS DES	+	m	x	
WALCS1	2	Grupo de limitación del campo de trabajo WKS 1 activo	+	m		
WALCS2	3	Grupo de limitación del campo de trabajo WKS 2 activo	+	m		
WALCS3	4	Grupo de limitación del campo de trabajo WKS 3 activo	+	m		
WALCS4	5	Grupo de limitación del campo de trabajo WKS 4 activo	+	m		
WALCS5	6	Grupo de limitación del campo de trabajo WKS 5 activo	+	m		
WALCS6	7	Grupo de limitación del campo de trabajo WKS 6 activo	+	m		
WALCS7	8	Grupo de limitación del campo de trabajo WKS 7 activo	+	m		
WALCS8	9	Grupo de limitación del campo de trabajo WKS 8 activo	+	m		
WALCS9	10	Grupo de limitación del campo de trabajo WKS 9 activo	+	m		
WALCS10	11	Grupo de limitación del campo de trabajo WKS 10 activo	+	m		

4.3.60 Grupo G 61: Alisado de la orientación de herramienta

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
ORISOF	1	Alisado de la orientación de herramienta DES	+	m	x	
ORISON	2	Alisado de la orientación de herramienta CON	+	m		

4.3.61 Grupo G 62: Modo de reposicionamiento para REPOS (efectivo por secuencia)

Comando G	N.º [1]	Descripción	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
RMBBL	1	Reposicionamiento en el punto inicial de la secuencia	-	s		
RMIBL	2	Reposicionamiento en el punto de interrupción	-	s	x	
RMEBL	3	Reposicionamiento en el punto final de la secuencia	-	s		
RMNBL	4	Reposicionamiento en el punto de trayectoria más cercano	-	s		

4.3.62 Grupo G 64: Frames de rectificado

Comando G	N.º [1]	Descripción Frame de rectificado activo en el canal $P_GFRAME =	MD20150 [2]	W [3]	STD [4]	
					SAG	MH
GFRAME[0]	1	Frame de rectificado de la gestión de datos $P_GFR[0] (frame cero)	+	m	x	
GFRAME[1]	2	Frame de rectificado de la gestión de datos $P_GFR[1]	+	m		
GFRAME[2]	3	Frame de rectificado de la gestión de datos $P_GFR[2]	+	m		
...	...		+	m		
GFRAME[100]	101	Frame de rectificado de la gestión de datos $P_GFR[100]	+	m		

Leyenda

1) Número interno (p. ej., para interfaz PLC)

2) Capacidad de configuración del comando G como posición de borrado del grupo G en el arranque, el reset o el fin del programa de pieza (con MD20150 $MC_GCODE_RESET_VALUES):

+ Configurable

- No configurable

3) Efecto del comando G:

m Modal (para todas las secuencias)

s Secuencial

4) Posición de borrado, ver los datos de máquina siguientes:

- MD20149 $MC_GCODE_RESET_S_VALUES (posición de borrado de los grupos G (fija))
- MD20150 $MC_GCODE_RESET_VALUES (posición de borrado de los grupos G)
- MD20151 $MC_GCODE_RESET_S_MODE (comportamiento de reset de grupos G (fijo))
- MD20152 $MC_GCODE_RESET_MODE (comportamiento de reset de los grupos G)
- MD20154 $MC_EXTERN_GCODE_RESET_VALUES (posición de borrado de los grupos G en modo ISO)
- MD20156 $MC_EXTERN_GCODE_RESET_MODE (comportamiento de reset de los grupos G externos)

SAG Ajuste por defecto de **S**iemens **AG**

MH Ajuste por defecto del **f**abricante de la **m**áquina (ver los datos del fabricante de la máquina)

4.4 Procedimientos predefinidos

Mediante la llamada a un procedimiento predefinido se inicia la ejecución de una función CN predefinida. Un procedimiento predefinido, a diferencia de una función predefinida, **no** proporciona un valor de retorno.

Sistema de coordenadas					
Identificador	**Parámetro**				**Explicación**
	1.	**2.**	**3. - 15.**	**4. - 16.**	
PRESETON	AXIS *): Identificador de eje Eje de máquina	REAL: Decalaje de Preset G700/G710 Contexto	Como 1...	Como 2...	Ajuste de valor real para los ejes programados con pérdida del estado de referenciado
PRESETONS	AXIS *): Identificador de eje Eje de máquina	REAL: Decalaje de Preset G700/G710 Contexto	Como 1...	Como 2...	Ajuste de valor real para los ejes programados sin pérdida del estado de referenciado
DRFOF					Borrado del decalaje DRF para todos los ejes pertenecientes al canal

*) En lugar del identificador del eje de máquina se pueden introducir identificadores para ejes geométricos o ejes adicionales, siempre que se trate de una configuración inequívoca.

Conjuntos de ejes					
Identificador	**Parámetro**				**Explicación**
GEOAX	**1.**	**2.**	**3./5.**	**4./6.**	Selección de un sistema de coordenadas paralelo
	INT: Número de eje geométrico 1 - 3	AXIS: Identificador de eje de canal	Como 1	Como 2	
FGROUP	**1. – 8.**				Referencia F variable: Define los ejes a los que se refiere el avance de contorneado Número máximo de ejes: 8 El ajuste estándar para el valor de referencia F se activa con FGROUP ( ) sin indicación de parámetros.
	AXIS: Identificador de eje de canal				
SPLINEPATH	**1.**	**2. - 9.**			Determinación del conjunto spline Número máximo de ejes: 8
	INT: Conjunto spline (tiene que ser 1)	AXIS: Identificador geométrico o adicional			

Conjuntos de ejes				
Identificador	**Parámetro**			**Explicación**
POLYPATH	**1.**	**2.**		Activación de la interpolación polinómica para grupos de ejes selectivos
	STRING	STRING		

Arrastre							
Identificador	**Parámetro**						**Explicación**
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
TANG	AXIS: Nombre de eje Eje esclavo	AXIS: Eje maestro 1	AXIS: Eje maestro 2	REAL: Factor de acoplamiento	CHAR: Opción: "B": Seguimiento en BKS "W": Seguimiento en WKS	CHAR Optimización: "S": estándar "P": autom. con recorrido de matado de esquinas, tolerancia angular	Control tangencial: Definir acoplamiento A partir de los dos ejes maestro indicados se determina la tangente para el arrastre. El factor de acoplamiento indica la relación entre el cambio angular de la tangente y el cambio angular del eje arrastrado. Generalmente es 1.
TANGON	AXIS: Nombre de eje Eje esclavo	REAL: Ángulo de offset	REAL: Recorrido de matado de esquinas	REAL: Tolerancia angular			Control tangencial: Activar acoplamiento
TANGOF	AXIS: Nombre de eje Eje esclavo						Control tangencial: Desactivar acoplamiento
TLIFT	AXIS: Eje arrastrado						Control tangencial: Activar generación de secuencia intermedia
TRAILON	AXIS: Eje arrastrado o esclavo	AXIS: Eje maestro	REAL: Factor de acoplamiento				Activar arrastre sincronizado de eje
TRAILOF	AXIS: Eje arrastrado o esclavo	AXIS: Eje maestro					Desactivar arrastre sincronizado de eje
TANGDEL	AXIS: Eje arrastrado o esclavo						Control tangencial: Borrar acoplamiento

Tablas de levas						
Identificador	**Parámetro**					**Explicación**
	1.	**2.**	**3.**	**4.**	**5.**	
CTABDEF	AXIS: Eje esclavo	AXIS: Eje maestro	INT: Número de tabla	INT: Comportamiento en los márgenes del área def.	STRING: Indicación de ubicación	Activación definición de tabla Las secuencias de desplazamiento siguientes determinan la tabla de levas.
CTABEND	AXIS: Eje esclavo	AXIS: Eje maestro	INT: Número de tabla	INT: Comportamiento en los márgenes del área def.		Desactivación definición de tabla
CTABDEL	INT: Número de tabla n	INT: Número de tabla m	STRING: Indicación de ubicación			Borrar tabla de levas
CTABLOCK	INT: Número de tabla n					Bloquea la tabla de levas con el número n, es decir, esta tabla no podrá borrarse ni sobrescribirse.
CTABUNLOCK	INT: Número de tabla n					Vuelve a habilitar la tabla protegida con CTABLOCK con el número n
LEADON	AXIS: Eje esclavo	AXIS: Eje maestro	INT: Número de tabla			Acoplamiento de valor maestro CON
LEADOF	AXIS: Eje esclavo	AXIS: Eje maestro				Acoplamiento de valor maestro DES

Perfil de aceleración por eje		
Identificador	**Parámetro**	**Explicación**
	1. – 8.	
BRISKA	AXIS	Activación de la aceleración "escalonada" para los ejes programados
SOFTA	AXIS	Activar la aceleración con limitación de tirones de los ejes programados
DRIVEA	AXIS	Activar aceleración con perfil discontinuo por tramos para los ejes programados
JERKA	AXIS	Activación para los ejes programados del comportamiento en aceleración definido mediante el dato de máquina $MA_AX_JERK_ENABLE.

Avance por vuelta			
Identificador	**Parámetro**		**Explicación**
FPRAON	**1.**	**2.**	Avance por vuelta por eje CON
	AXIS: Eje para el que se activa el avance por vuelta	AXIS: Eje/cabezal, guía para el avance por vuelta. Si no se programa ningún eje, el avance por vuelta se toma del cabezal maestro.	
FPRAOF	**1. - n.**		Avance por vuelta por eje DES El avance por vuelta puede desactivarse simultáneamente para varios ejes. Se pueden programar tantos ejes como sean admisibles en una secuencia.
	AXIS: Ejes para los que se desactiva el avance por vuelta		
FPR	**1.**		Selección de un eje giratorio/cabezal que sirve de guía para la velocidad de avance por vuelta con G95. El ajuste indicado con FPR es válido de forma modal.
	AXIS: Eje/cabezal, guía para el avance por vuelta. Si no se programa ningún eje, el avance por vuelta se toma del cabezal maestro.		

Transformadas				
Identificador	**Parámetro**			**Explicación**
	1.	**2.**	**3.**	
TRACYL	REAL: Diámetro de trabajo	INT: Número de la transformada		Cilindro: Transformada de superficie envolvente Por cada canal se pueden ajustar varias transformadas. El número de la transformada introducido define la transformada que se desea activar. Si se omite el 2º parámetro, entonces se activa la transformada definida mediante datos de máquina.
TRANSMIT	INT: Número de la transformada			Transmit: Transformada polar Por cada canal se pueden ajustar varias transformadas. El número de la transformada introducido define la transformada que se desea activar. Si se omite el parámetro, entonces se activa la transformada definida mediante datos de máquina.

Transformadas				
Identificador	**Parámetro**			**Explicación**
	1.	**2.**	**3.**	
TRAANG	REAL: Ángulo	INT: Número de la transformada		Transformada eje oblicuo Por cada canal se pueden ajustar varias transformadas. El número de la transformada introducido define la transformada que se desea activar. Si se omite el 2º parámetro, entonces se activa la transformada definida mediante datos de máquina. Si no se programa el ángulo TRAANG (,2) o TRAANG , el último ángulo introducido es válido de forma modal.
TRAORI	INT: Número de la transformada			Transformada de 4, 5 ejes Por cada canal se pueden ajustar varias transformadas. El número de la transformada introducido define la transformada que se desea activar.
TRACON	INT: Número de la transformada	REAL: Otros parámetros dependientes de DM.		Transformada en cascada El significado de los parámetros depende del tipo de concatenación en cascada.
TRAFOOF				Desactivar transformada
TRAFOON	STRING: Nombre del juego de datos de la transformada	REAL: Diámetro de referencia y de trabajo (solo TRACYL)	BOOL: Con/sin corrección de pared de ranura (solo TRACYL)	Activar transformada definida con cadenas cinemáticas

Cabezal			
Identificador	**Parámetro**		**Explicación**
	1	**2. - n.**	
SPCON	INT: Número de cabezal	INT: Número de cabezal	Conmutación al modo de cabezal regulado por posición
SPCOF	INT: Número de cabezal	INT: Número de cabezal	Conmutación al modo de cabezal regulado por velocidad
SETMS	INT: Número de cabezal		Definición para el canal actual de un cabezal maestro Con SETMS( ) sin introducción de parámetros se activa el ajuste predeterminado mediante datos de máquina.

Rectificado		
Identificador	**Parámetro**	**Explicación**
	1.	
GWPSON	INT: Número de cabezal	Velocidad periférica de muela constante CON Si se omite el número de cabezal, se activa la velocidad periférica de muela constante para el cabezal de la herramienta activa.
GWPSOF	INT: Número de cabezal	Velocidad periférica de muela constante DES Si se omite el número de cabezal, se desactiva la velocidad periférica de muela constante para el cabezal de la herramienta activa.
TMON	INT: Número T	Vigilancia específica de muelas CON Si se omite el número T, se activa dicha vigilancia para la herramienta actualmente seleccionada.
TMOF	INT: Número T	Vigilancia de herramienta DES Si se omite el número T, se desactiva dicha vigilancia para la herramienta actualmente seleccionada.

Desbaste					
Identificador	**Parámetro**				**Explicación**
	1.	**2.**	**3.**	**4.**	
CONTPRON	REAL [,11]: Tabla de contorno	CHAR: Tipo de mecanizado	INT: Número de destalonados	INT: Estado del cálculo	Activar preparación de referencia Los programas del contorno o secuencias CN que se van a llamar se dividen en desplazamientos individuales y se almacenan en la tabla de contornos. Se devuelve el número de puntos de destalonado.
CONTDCON	REAL [, 6]: Tabla de contorno	INT: Sentido de mecanizado			Codificación de contornos Las secuencias de un contorno se codifican en una tabla con una fila de la tabla por cada secuencia dada para ahorrar memoria.
EXECUTE	INT: Estado de error				Activar ejecución de programa De esta manera, se puede volver desde el modo de preparación de referencia, o bien tras generar una zona protegida, a la ejecución normal del programa.

Ejecutar tabla		
Identificador	**Parámetro**	**Explicación**
	1.	
EXECTAB	REAL [11]: Elemento de la tabla de movimientos	Ejecutar un elemento de una tabla de movimientos

Zonas protegidas						
Identificador	**Parámetro**					**Explicación**
	1.	**2.**	**3.**	**4.**	**5.**	
CPROTDEF	INT: Número de la zona protegida	BOOL: TRUE: Zona protegida orientada a la herramienta	INT: 0: El 4.º y el 5.º parámetros no se evalúan 1: El 4.º parámetro se evalúa 2: El 5.º parámetro se evalúa 3: El 4.º y el 5.º parámetros se evalúan	REAL: Límite en sentido positivo	REAL: Límite en sentido negativo	Definición de una zona protegida específica de canal
NPROTDEF	INT: Número de la zona protegida	BOOL: TRUE: Zona protegida orientada a la herramienta	INT: 0: El 4.º y el 5.º parámetros no se evalúan 1: El 4.º parámetro se evalúa 2: El 5.º parámetro se evalúa 3: El 4.º y el 5.º parámetros se evalúan	REAL: Límite en sentido positivo	REAL: Límite en sentido negativo	Definición de una zona protegida específica de máquina

Zonas protegidas						
Identificador	**Parámetro**					**Explicación**
	1.	**2.**	**3.**	**4.**	**5.**	
CPROT	INT: Número de la zona protegida	INT: Opción 0: Zona protegida DES 1: Preactivar zona protegida 2: Zona protegida CON 3: Preactivar zona protegida con parada condicionada, sólo con zonas protegidas activas	REAL: Decalaje de la zona protegida en el primer eje geométrico	REAL: Decalaje de la zona protegida en el segundo eje geométrico	REAL: Decalaje de la zona protegida en el tercer eje geométrico	Zona protegida específica de canal CON/DES
NPROT	INT: Número de la zona protegida	INT: Opción 0: Zona protegida DES 1: Preactivar zona protegida 2: Zona protegida CON 3: Preactivar zona protegida con parada condicionada, sólo con zonas protegidas activas	REAL: Decalaje de la zona protegida en el primer eje geométrico	REAL: Decalaje de la zona protegida en el segundo eje geométrico	REAL: Decalaje de la zona protegida en el tercer eje geométrico	Zona protegida específica de máquina ON/DES

Decodificación previa/secuencia a secuencia		
Identificador	**Parámetro**	**Explicación**
STOPRE		Parada de decodificación previa hasta que se hayan ejecutado todas las secuencias preparadas del proceso principal
SBLOF		Suprimir ejecución de secuencia individual
SBLON		Cancelar supresión de la ejecución de secuencia individual

Interrupciones		
Identificador	**Parámetro**	**Explicación**
	1.	
DISABLE	INT: N.º de la entrada de interrupción	Se desactiva la rutina de interrupción asociada a la entrada hardware indicada. Tampoco se ejecutan las retiradas rápidas del contorno. La asignación entre la entrada hardware y la rutina de interrupción realizada mediante SETINT permanece válida y puede reactivarse otra vez con ENABLE
ENABLE	INT: N.º de la entrada de interrupción	Reactivación de la asignación de rutinas de interrupción desactivada con DISABLE
CLRINT	INT: N.º de la entrada de interrupción	Cancela las asignaciones de rutinas de interrupción así como los atributos en una entrada de interrupción. La rutina de interrupción se desactiva. No se produce ninguna reacción en el momento en que se genere la interrupción.

Acciones síncronas		
Identificador	**Parámetro**	**Explicación**
	1. – n.	
CANCEL	INT: Número de acción síncrona	Interrupción de la acción síncrona modal con el ID indicado También pueden indicarse varios ID separados por comas.
CANCELSUB		Cancelar nivel de subprograma

Definición de funciones					
Identificador	**Parámetro**				**Explicación**
	1.	**2.**	**3.**	**4. - 7.**	
FCTDEF	INT: Número de función	REAL: Límite inferior	REAL: Límite superior	REAL: Coeficientes a0-a3	Definición de la función polinomio Se evalúa con las funciones SYNFCT o PUTFTOCF.

Comunicación			
Identificador	**Parámetro**		**Explicación**
	1.	**2.**	
MMC	STRING: Comando	CHAR: Modo de acuse*) "N": Sin acuse "S": Acuse síncrono "A": Acuse asíncrono	Comando al intérprete de comandos HMI para la parametrización de ventanas mediante programa CN

*)Los comandos se confirman a petición del componente que se está ejecutando (canal, CN...).

Coordinación de programas					
Identificador	**Parámetro**				**Explicación**
INIT	**1.**	**2.**	**3.**		Selección de un programa CN para su ejecución en un canal
	INT: Número de canal **o** nombre de canal según MD20000*)	STRING: Ruta de acceso	CHAR: Modo de acuse**)		
	1. - n.				
START	INT: Número de canal **o** nombre de canal según MD20000*)				Arranque de los programas seleccionados simultáneamente por varios canales desde el programa actualmente en ejecución Este comando no tiene ningún efecto en el canal propio.
WAITE	INT: Número de canal **o** nombre de canal según MD20000*)				Esperar el fin del programa en otro canal u otros canales.
	1.	**2. - n.**			
WAITM	INT: Número de meta	INT: Número de canal **o** nombre de canal según MD20000*)			Esperar a que se alcance una meta en los canales indicados. La secuencia anterior finaliza con parada precisa.
WAITMC	INT: Número de meta	INT: Número de canal **o** nombre de canal según MD20000*)			Esperar a que se alcance una meta en los canales indicados. La parada precisa sólo se activa si los otros canales no han alcanzado aún las metas.
	1. - n.				
SETM	INT: Número de meta				Activar una o varias metas para la coordinación de canales El mecanizado en el canal propio no se verá afectado por esto.
CLEARM	INT: Número de meta				Borrar una o varias metas para la coordinación de canales El mecanizado en el canal propio no se verá afectado por esto.
	1. - n.				

Coordinación de programas					
Identificador	**Parámetro**				**Explicación**
WAITP	AXIS: Identificador de eje				Esperar hasta que los ejes de posicionado indicados que se habían programado anteriormente con POSA alcancen su punto final programado
WAITS	INT: Número de cabezal				Esperar hasta que los cabezales indicados que se habían programado anteriormente con SPOSA alcancen su punto final programado
RET	**1.**	**2.**	**3.**	**4.**	Fin de subprograma sin emisión de funciones al PLC Al indicar el 1.er parámetro (destino de salto), el salto atrás no se produce hasta la secuencia posterior a la secuencia de llamada. A continuación se busca el destino en función de la programación (RET o RETB) según la estrategia siguiente: • RET: Búsqueda hacia el final del programa. Si la búsqueda no tiene éxito, a continuación se busca hacia el inicio del programa. • RETB: Búsqueda hacia el principio del programa. Si la búsqueda no tiene éxito, a continuación se busca hacia el final del programa.
	INT (o STRING): Destino del salto (n.º de sec./ meta) para el salto hacia atrás	INT: 0: Salto hacia atrás al destino del salto del 1.er par. > 0: Salto hacia atrás a la secuencia siguiente	INT: Número de niveles de subprograma que hay que saltar	BOOL: Salto hacia atrás a la primera secuencia en el programa principal	
RETB	INT (o STRING): Destino del salto (n.º de sec./ meta) para el salto hacia atrás	INT: 0: Salto hacia atrás al destino del salto del 1.er par. > 0: Salto hacia atrás a la secuencia siguiente	INT: Número de niveles de subprograma que hay que saltar	BOOL: Salto hacia atrás a la primera secuencia en el programa principal	
	1. - n.				
GET	AXIS: Identificador de eje ***)				Ocupar ejes de máquina Se deben liberar los ejes indicados con RELEASE en otro canal.
GETD	AXIS: Identificador de eje ***)				Ocupar directamente ejes de máquina Los ejes indicados **no** deben liberarse con RELEASE.
RELEASE	AXIS: Identificador de eje ***)				Liberar ejes de máquina
	1.	**2.**	**3.**	**4.**	

Coordinación de programas					
Identificador	**Parámetro**				**Explicación**
PUTFTOC	REAL: Valor de corrección	INT: Número de parámetro	INT: Número de canal **o** nombre de canal según MD20000*)	INT: Número de cabezal	Modificación de la corrección de herramienta fina
PUTFTOCF	INT: N.º de función	VAR REAL: Valor de ref.	INT: Número de parámetro	INT: Número de canal **o** nombre de canal según MD20000*)	Modificación de la corrección de herramienta fina dependiendo de una función prefijada con FCTDEF (polinomio de hasta 3.er grado) Con FCTDEF debe indicarse el núm. aquí utilizado.
AXTOCHAN	**1.**	**2.**	**3. - n.**	**4. - m.**	Transferir ejes a otros canales
	AXIS: Identificador de eje	INT: Número de canal **o** nombre de canal según MD20000*)	Como 1...	Como 2...	

*) En lugar de los números de canal, también pueden programarse los nombres de canal definidos con MD20000 $MC_CHAN_NAME.

**) Los comandos se confirman a petición del componente que se está ejecutando (canal, CN...).

***) En lugar de un eje se puede programar un cabezal mediante la función SPI: p. ej. GET(SPI(1))

Accesos a los datos		
Identificador	**Parámetro**	**Explicación**
CHANDATA	**1.** INT: Número de canal	Ajustar número de canal para los accesos a datos de canal (solo se permite en el bloque de inicialización). Los accesos siguientes se refieren al canal ajustado con CHANDATA.
NEWCONF		Aceptar datos de máquina modificados

Avisos			
Identificador	**Parámetro**		**Explicación**
	1.	**2.**	
MSG	STRING: Aviso	INT: Ejecución	Salida de una cadena de caracteres cualquiera como aviso en la interfaz de usuario
WRTPR	STRING: Cadena de caracteres	INT: Ejecución	Escribir cadena de caracteres en variable BTSS

Accesos a ficheros						
Identificador	**Parámetro**					**Explicación**
READ	**1.**	**2.**	**3.**	**4.**	**5.**	Leer secuencias del sistema de ficheros
	VAR INT: Error	CHAR[160]: Nombre de fichero	INT: Línea inicial del área de fichero que se va a leer	INT: Número de líneas que hay que leer	VAR CHAR[255]: Campo de variables en el que se guarda la información leída	
WRITE	**1.**	**2.**	**3.**	**4.**		Escribir secuencia en el sistema de ficheros (o en una unidad/fichero externo)
	VAR INT: Error	CHAR[160]: Nombre de fichero	STRING: Unidad/fichero para salida externa	CHAR[200]: Secuencia		
DELETE	**1.**	**2.**				Borrar fichero
	VAR INT: Error	CHAR[160]: Nombre de fichero				

Alarmas			
Identificador	**Parámetro**		**Explicación**
	1.	**2.**	
SETAL	INT: Número de alarma (alarmas de ciclo)	STRING: Cadena de caracteres	Activar alarma Se puede introducir, además del número de alarma, una cadena de caracteres con hasta 4 parámetros. Se encuentran disponibles los siguientes parámetros predefinidos: %1 = número de canal %2 = número de secuencia, label %3 = índice de texto para alarmas de ciclo %4 = parámetro de alarma adicional

Gestión de herramientas							
Identificador	**Parámetro**						**Explicación**
	1.	**2.**					
DELDL	INT: N.º T	INT: N.º D					Borrar la suma de todas las correcciones de un filo (o de una hta. cuando no se indique un número D).
DELT	STRING[32]: Identificador de herramienta	INT: N.º Duplo					Borrar herramienta Se puede omitir el número duplo.
DELTC	INT: N.º de secuencia n	INT: N.º de secuencia m					Borrar número de secuencia de portaherramientas n a m
DZERO							Activar como inválidos los números D de todas las herramientas de la unidad TO asignada al canal
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
GETFREELOC	VAR INT: N.º de almacén (valor de retorno)	VAR INT: N.º de puesto (valor de retorno)	INT: N.º T	INT: N.º referencia de almacén	CHAR: Dato depend. del 4.º parámetro	INT: Modo de reserva	Buscar puesto vacío para una herramienta
	1.	**2.**					
GETSELT	VAR INT: N.º T (valor de retorno)	INT: N.º de cabezal					Proporciona el número T de la herramienta preseleccionada para el cabezal.
GETEXET	VAR INT: N.º T (valor de retorno)	INT: N.º de cabezal					Proporciona el número T de la herramienta activa desde la perspectiva del programa CN
GETTENV	STRING: Nombre del entorno de la herramienta	INT ARRAY[3]: Valores de retorno					Lee los números T, D y DL guardados en un entorno de herramienta
	1.	**2.**	**3.**	**4.**			

Gestión de herramientas						
Identificador	**Parámetro**					**Explicación**
POSM	INT: N.º del puesto en el que se debe posicionar	INT: N.º del almacén que se debe mover	INT: N.º de puesto del almacén interno	INT: N.º del almacén interno		Posicionar almacén
RESETMON	VAR INT: Estado = resultado de la operación (valor de retorno)	INT: interno N.º T	INT: N.º D de la herramienta	INT: Parámetro opcional codificado en bits		Fijar el valor real de la herramienta al valor de consigna
SETDNO	**1.** INT: N.º T	**2.** INT: N.º de filo	**3.** INT: N.º D			Fijar número de corrección (D) del filo de la herramienta (T)
SETMTH	**1.** INT: N.º de portaherramientas					Poner Nº de portaherramientas.
SETPIECE	**1.** INT: Valor por el que se decrementa	**2.** INT: N.º de cabezal				Decrementar contador de piezas del cabezal De esta forma el usuario puede actualizar los datos de vigilancia del número de piezas de las herramientas involucradas en el proceso de mecanizado.
	1.	**2.**	**3.**	**4.**		
SETTA	VAR INT: Estado = resultado de la operación (valor de retorno)	INT: N.º de almacén	INT: N.º de conjunto de desgaste	INT: Subgrupo de herramientas		Definir herramienta como activa en conjunto de desgaste
SETTIA	VAR INT: Estado = resultado de la operación (valor de retorno)	INT: N.º de almacén	INT: N.º de conjunto de desgaste	INT: Subgrupo de herramientas		Definir herramienta como inactiva en conjunto de desgaste
TCA	**1.** STRING [32]: Identificador de herramienta	**2.** INT: N.º Duplo	**3.** INT: N.º de portaherramientas			Elección/cambio de herramienta independiente de su estado

Gestión de herramientas							
Identificador	**Parámetro**						**Explicación**
TCI	**1.**	**2.**					Cambiar herramienta desde la memoria intermedia en el almacén
	INT: N.º de memoria intermedia	INT: N.º de portaherramientas					
MVTOOL	**1.**	**2.**	**3.**	**4.**	**5.**		Instrucción de programación para el desplazamiento de una herramienta
	INT: Estado	INT: N.º de almacén	INT: N.º de puesto	INT: N.º de almacén después del movimiento	INT: N.º de puesto de destino después del movimiento		

Orientación de la herramienta				
Identificador	**Parámetro**			**Explicación**
	1.	**2.**	**3.**	
ORIRESET	REAL: Posición básica 1.er eje geom.	REAL: Posición básica 2.º eje geom.	REAL: Posición básica 3.er eje geom.	Posición básica de la orientación de herramienta

Cabezal síncrono							
Identificador	**Parámetro**						**Explicación**
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
COUPDEF	AXIS: Cabezal esclavo	AXIS: Cabezal maestro	REAL: Numerador de relación de transmisión	REAL: Denominador de relación de transmisión	STRING[8]: Comportamiento en cambio de secuencia	STRING[2]: Tipo de acoplamiento	Definir conjunto de cabezales síncronos
COUPDEL	AXIS: Cabezal esclavo	AXIS: Cabezal maestro					Borrar conjunto de cabezales síncronos
COUPRES	AXIS: Cabezal esclavo	AXIS: Cabezal maestro					Resetear los parámetros de acoplamiento a los valores DM y SD configurados

Cabezal síncrono							
Identificador	**Parámetro**						**Explicación**
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
COUPON	AXIS: Cabezal esclavo	AXIS: Cabezal maestro	REAL: Posición de acoplamiento del cabezal esclavo				Activación del acoplamiento del cabezal síncrono Si se indica para el cabezal esclavo una posición de acoplamiento (decalaje angular entre cabezal esclavo y maestro referida, de forma absoluta o incremental, a la posición de cero grados del cabezal maestro en sentido de giro positivo), el acoplamiento no se activará hasta que se rebase la posición indicada.
COUPONC	AXIS: Cabezal esclavo	AXIS: Cabezal maestro					Activación del acoplamiento del cabezal síncrono Con COUPONC se aplica la velocidad efectiva actual del cabezal esclavo (M3/M4 S...) al activarse el acoplamiento.
COUPOF	AXIS: Cabezal esclavo	AXIS: Cabezal maestro	REAL: Posición de desconexión del cabezal esclavo (absoluta)	REAL: Posición de desconexión del cabezal maestro (absoluta)			Desactivación del acoplamiento del cabezal síncrono Si se definen posiciones para el desacoplamiento, éste se produce cuando todas las posiciones indicadas han sido alcanzadas. El cabezal esclavo gira con la velocidad programada antes de la activación del acoplamiento.

Cabezal síncrono							
Identificador	**Parámetro**						**Explicación**
	1.	**2.**	**3.**	**4.**	**5.**	**6.**	
COUPOFS	AXIS: Cabezal esclavo	AXIS: Cabezal maestro	REAL: Posición de desconexión del cabezal esclavo (absoluta)				Desactivación del acoplamiento del cabezal síncrono con parada del cabezal esclavo Si se define una posición, el desacoplamiento no se producirá hasta que no se haya rebasado la posición indicada.
WAITC	AXIS: Cabezal esclavo	STRING [8]: Comportamiento en cambio de secuencia	AXIS: Cabezal esclavo	STRING[8]: Comportamiento en cambio de secuencia			Esperar hasta que se haya cumplido el criterio de cambio de secuencia de acoplamiento para los cabezales (máx. 2) Si no se indica el comportamiento de cambio de secuencia, se aplicará el comportamiento indicado en la definición con COUPDEF.

Reductor electrónico					
Identificador	**Parámetro**				**Explicación**
EGDEL	**1.**				Borrar la definición de acoplamiento para el eje esclavo
	AXIS: Eje esclavo				
EGDEF	**1.**	**2./4./6./8./10.**	**3./5./7./9./11.**		Definición de un reductor electrónico
	AXIS: Eje esclavo	AXIS: Eje maestro	INT: Tipo de acoplamiento		

Reductor electrónico									
Identificador	**Parámetro**								**Explicación**
EGON	**1.**	**2.**	**3./6./9./12./15.**	**4./7./10./13./16.**	**5./8./11./14./17.**				Reductor electrónico CON sin sincronización
	AXIS: Eje esclavo	STRING: Comportamiento de cambio de secuencia	AXIS: Eje maestro	REAL: Numerador del factor de acoplamiento	REAL: Denominador del factor de acoplamiento				
EGONSYN	**1.**	**2.**	**3.**	**4./8./12./16./20.**	**5./9./13./17./21.**	**6./10./14./18./22.**	**7./11./15./19./23.**		Reductor electrónico CON con sincronización
	AXIS: Eje esclavo	STRING: Comportamiento de cambio de secuencia	REAL: Posición síncrona del eje esclavo	AXIS: Eje maestro	REAL: Posición síncrona del eje maestro	REAL: Numerador del factor de acoplamiento	REAL: Denominador del factor de acoplamiento		
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.**	Reductor electrónico CON con sincronización y especificación del modo de aproximación
	AXIS: Eje esclavo	STRING: Comportamiento de cambio de secuencia	REAL: Posición síncrona del eje esclavo	STRING: Modo de aproximación	AXIS: Eje maestro	REAL: Posición síncrona del eje maestro	REAL: Numerador del factor de acoplamiento	REAL: Denominador del factor de acoplamiento	
EGOFS	**1.**	**2. - n.**							Desactivación selectiva del reductor electrónico
	AXIS: Eje esclavo	AXIS: Eje maestro							
EGOFC	**1.**								Desactivar reductor electrónico (variante solo para cabezales)
	AXIS: Cabezal esclavo								

Punzonado					
Identificador	**Parámetro**				**Explicación**
	1.	**2.**	**3.**	**4.**	
PUNCHAAC	REAL: Menor distancia entre agujeros	REAL: Aceleración inicial	REAL: Mayor distancia entre agujeros	REAL: Aceleración final	Activar la aceleración dependiente del recorrido

Funciones de información en sistema de ficheros pasivo				
Identificador	Parámetro			Explicación
	1.	2.	3.	
FILEDATE	VAR INT: Aviso de error	CHAR[160]: Nombre de fichero	VAR CHAR[8]: Fecha en formato "dd.mm.aa"	Indica la fecha del último acceso de escritura a un fichero
FILETIME	VAR INT: Aviso de error	CHAR[160]: Nombre de fichero	VAR CHAR[8]: Hora en formato "hh.mm.ss"	Indica la hora del último acceso de escritura a un fichero
FILESIZE	VAR INT: Aviso de error	CHAR[160]: Nombre de fichero	VAR INT: Tamaño de fichero	Suministra el tamaño actual de un fichero
FILESTAT	VAR INT: Aviso de error	CHAR[160]: Nombre de fichero	VAR CHAR[5]: Fecha en formato "rwxsd"	Suministra el estado de un fichero con respecto a los permisos siguientes: • Lectura (r: read) • Escritura (w: write) • Ejecutar (x: execute) • Mostrar (s: show) • Borrar (d: delete)
FILEINFO	VAR INT: Aviso de error	CHAR[160]: Nombre de fichero	VAR CHAR[32]: Fecha en formato "rwxsd nnnnnnnn dd.mm.yy hh:mm:ss"	Proporciona para un fichero la suma de las informaciones que pueden leerse mediante FILEDATE, FILETIME, FILESIZE y FILESTAT

Contenedor de ejes		
Identificador	Parámetro	Explicación
	1. - n.	
AXCTSWE	AXIS: Contenedor de ejes	Rotación del contenedor de ejes
AXCTSWED	AXIS: Contenedor de ejes	Girar contenedor de ejes (¡variante de comando para puesta en marcha!)
AXCTSWEC:	AXIS: Contenedor de ejes	Anular la habilitación para giro del contenedor de ejes

Acoplamiento maestro/esclavo		
Identificador	**Parámetro**	**Explicación**
	1. - n.	
MASLON	AXIS: Identificador de eje	Activación acoplamiento maestro/esclavo
MASLOF	AXIS: Identificador de eje	Separación acoplamiento maestro/esclavo
MASLOFS	AXIS: Identificador de eje	Separar acoplamiento maestro/esclavo y frenar automáticamente cabezales esclavos
MASLDEF	AXIS: Identificador de eje	Definir acoplamiento maestro/esclavo El último eje es el eje maestro.
MASLDEL	AXIS: Identificador de eje	Separar acoplamiento maestro/esclavo y borrar definición del conjunto

Corrección longitudinal de herramienta online			
Identificador	**Parámetro**		**Explicación**
	1.	**2.**	
TOFFON	AXIS: Sentido de la corrección	REAL: Valor offset en el sentido de la corrección	Activar corrección de longitud de la herramienta online en el sentido especificado
TOFFOF	AXIS: Sentido de la corrección		Resetear corrección de longitud de la herramienta online en el sentido especificado

SERUPRO		
Identificador	**Parámetro**	**Explicación**
IPTRLOCK		Inicio de la sección de programa no apta para la búsqueda
IPTRUNLOCK		Fin de la sección de programa no apta para la búsqueda

Retirada		
Identificador	**Parámetro**	**Explicación**
	1. - n.	
POLFMASK	AXIS: Nombre de eje geométrico o de máquina	Liberar ejes para la retirada rápida (sin relación entre los ejes)
POLFMLIN	AXIS: Nombre de eje geométrico o de máquina	Liberar ejes para la retirada rápida lineal

Retirada				
Identificador	**Parámetro**			**Explicación**
POLFA	**1.**	**2.**	**3.**	Posición de retirada para ejes individuales
	AXIS: Identificador de eje de canal	INT: Tipo	REAL: Valor	

Prevención de colisiones			
Identificador	**Parámetro**		**Explicación**
	1.		
PROTA	STRING: "R"		Solicitar nuevo cálculo del modelo de colisión
PROTS	**1.**	**2. - n.**	Asignar estado a zona protegida
	CHAR: Estado	STRING: Nombre de la zona protegida	

Adaptación inteligente de la carga					
Identificador	**Parámetro**				**Explicación**
CADAPTON	**1.**	**2.**	**3.**	**4.**	Activar adaptación de carga
	INT: Estado	AXIS: Nombre de eje de máquina	INT: Magnitud de entrada	REAL: Valor de entrada (opcional)	
CADAPTOF	**1.**	**2.**	**3.**		Desactivar adaptación de carga
	INT: Estado	AXIS: Nombre de eje de máquina	INT: Magnitud de entrada		

Adaptación de tirón		
Identificador	**Parámetro**	**Explicación**
CALCFIR		Adaptación del filtro antitirones FIR al modo dinámico Se utiliza CALCFIR en el ciclo de fabricante CUST_832.

4.5 Procedimientos predefinidos en acciones síncronas

Los siguientes procedimientos predefinidos están disponibles únicamente en las acciones síncronas.

Procedimientos síncronos		
Identificador	**Parámetro**	**Explicación**
STOPREOF		Suprimir parada de decodificación previa Una acción síncrona con el comando STOPREOF genera una parada de decodificación previa tras la siguiente salida de la secuencia (secuencia del proceso principal). La parada de pretratamiento se cancela con el final de la salida de la secuencia o cuando se ha cumplido la condición de STOPREOF. Todas las instrucciones síncronas al comando STOPREOF se interpretan y se tienen en cuenta como si ya se hubiesen ejecutado.
RDISABLE		Bloqueo de lectura
DELDTG	**1.** AXIS: Eje para el borrado del trayecto residual (opcional). Si se omite el eje, se produce el borrado del trayecto residual para la trayectoria.	Borrado de trayecto residual Una acción síncrona con el comando DELDTG genera una parada de decodificación previa tras la siguiente salida de la secuencia (secuencia del proceso principal). La parada de pretratamiento se cancela con el final de la salida de la secuencia o cuando se ha cumplido la primera condición de DELDTG. En $AA_DELT[Eje] se encuentran las distancias axiales al punto de destino a la hora de realizar borrados del trayecto residual por eje en $AC_DELT la trayectoria residual.

Coordinación de programas ciclos tecnológicos		
Identificador	**Parámetro**	**Explicación**
	1.	
LOCK	INT: ID de la acción síncrona que debe bloquearse	Bloquear acción síncrona con ID o parar ciclo tecnológico Se pueden programar uno o varios ID.
UNLOCK	INT: ID de la acción síncrona que debe habilitarse	Habilitar acción síncrona con ID o continuar ciclo tecnológico Se pueden programar uno o varios ID.
ICYCON		Ejecutar cada secuencia de un ciclo tecnológico según ICYCON en un ciclo IPO independiente
ICYCOF		Ejecución de todas las secuencias de un ciclo tecnológico según ICYCOF en un ciclo IPO

Funciones polinomio						
Identificador	**Parámetro**					**Explicación**
SYNFCT	**1.**	**2.**	**3.**			Si se cumple la condición de la acción síncrona al desplazamiento, el polinomio determinado por la primera expresión se evalúa como variable de entrada. Los valores superior e inferior del rango están limitados, y se asigna a las variables resultantes.
	INT: Número de la función polinómica definida para la función FCTDEF	VAR REAL: Variable de resultado *)	VAR REAL: Variable de resultado **)			
FTOC	**1.**	**2.**	**3.**	**4.**	**5.**	Modificación de la corrección fina de herramienta dependiendo de una función prefijada mediante FCTDEF (polinomio de hasta 3.er grado). Si se usa la función FCTDEF, se debe introducir el número utilizado aquí.
	INT: Número de la función polinómica definida para la función FCTDEF	VAR REAL: Variable de entrada **)	INT: Longitud 1, 2, 3	INT: Número de canal	INT: Número de cabezal	

*) Como variables de resultado se permiten únicamente variables de sistema especiales (ver Manual de funciones Acciones síncronas).

**) Como variables de entrada se permiten únicamente variables de sistema especiales (ver Manual de funciones Acciones síncronas).

4.6 Funciones predefinidas

Mediante la llamada a una función predefinida se impide la ejecución de un comando CN predefinido que, a diferencia del procedimiento predefinido, proporciona un valor de retorno. La llamada a la función predefinida puede estar en la expresión como operando.

Sistema de coordenadas						
Identificador	**Valor de retorno**	**Parámetro**				**Explicación**
		1.	**2.**	**3. - 15.**	**4. - 16.**	
CTRANS	FRAME	AXIS: Identificador de eje	REAL: Decalaje	AXIS: Identificador de eje	REAL: Decalaje	Translation: decalaje de origen GRUESO para varios ejes
CFINE	FRAME	AXIS: Identificador de eje	REAL: Decalaje	AXIS: Identificador de eje	REAL: Decalaje	Translation: decalaje de origen FINO para varios ejes
CSCALE	FRAME	AXIS: Identificador de eje	REAL: Factor de escala	AXIS: Identificador de eje	REAL: Factor de escala	Scale: Factor de escala para varios ejes
		1.	**2.**	**3. y 5.**	**4. y 6.**	
CROT	FRAME	AXIS: Identificador de eje	REAL: Rotación	AXIS: Identificador de eje	REAL: Rotación	Rotation: Giro del sistema de coordenadas actual Número máximo de parámetros: 6 (un identificador de eje y un valor para cada eje geométrico).
CROTS	FRAME	AXIS: Identificador de eje	REAL: Giro con ángulo espacial	AXIS: Identificador de eje	REAL: Giro con ángulo espacial	Rotation: Giro del sistema de coordenadas actual con ángulo espacial Número máximo de parámetros: 6 (un identificador de eje y un valor para cada eje geométrico).
CMIRROR		**1.**	**2. - 8.**			Mirror: Simetría respecto a un eje del sistema de coordenadas.
	FRAME	AXIS	AXIS			

Sistema de coordenadas						
Identificador	**Valor de retorno**	**Parámetro**				**Explicación**
		1.	**2.**			
CRPL	FRAME	INT: Eje giratorio	REAL: Ángulo de giro			Rotación de frame en un plano cualquiera
ADDFRAME	INT: 0: OK 1: El destino indicado (string) es erróneo 2: El frame de destino no está configurado 3: La rotación en el frame no está permitida	FRAME: frame aditivo medido o calculado	STRING: frame de destino especificado			Calcula el frame de destino que está especificado por el string El frame de destino se calcula de forma que el nuevo frame total se obtiene a partir de la concatenación del antiguo frame total con el frame transferido.
INVFRAME	FRAME	**1.** FRAME				Calcula a partir de un frame el frame inverso La concatenación de un frame con su frame inverso da siempre un frame nulo
MEAFRAME	FRAME	**1.** REAL[3,3]: Coordenadas de los puntos espaciales medidos	**2.** REAL[3,3]: Coordenadas de los puntos de consigna	**3.** VAR REAL: Variable con la información sobre calidad del cálculo del FRAME que se devuelve		Cálculo del frame a partir de 3 puntos medidos en el espacio

Funciones geométricas					
Identificador	**Valor de retorno**	**Parámetro**			**Explicación**
		1.	**2.**	**3.**	
CALCDAT	BOOL: Estado de error	VAR REAL [n, 2]: Tabla (abscisa, ordenada) de los puntos 1 a n	INT: Número de puntos	VAR REAL [3]: Resultado: Abscisa, ordenada y radio del centro de arco calculado	Calcula las coordenadas del centro y el radio de la circunferencia a partir de 3 o 4 puntos. Los puntos deben ser distintos.
INTERSEC	BOOL: Estado de error	VAR REAL [11]: Primer elemento de contorno	VAR REAL [11]: Segundo elemento de contorno	VAR REAL [2]: Vector de resultado para la coordenada de punto de intersección: abscisa y ordenada	Calcula las coordenadas de punto de intersección entre dos elementos de contorno. El estado del error indica si se ha encontrado un punto de intersección.

Funciones de tablas de levas								
Identificador	**Valor de retorno**	**Parámetro**						**Explicación**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTAB	REAL: Posición del eje esclavo	REAL: Posición del eje maestro	INT: Número de tabla	VAR REAL[ ]: Resultado de paso	AXIS: Eje esclavo para escalado	AXIS: Eje maestro para escalado		Determina la posición del eje esclavo respecto a la posición indicada del eje maestro a partir de la tabla de levas. Si los parámetros 4/5 no están programados, se calcula con el escalado estándar.
CTABINV	REAL: Posición del eje maestro	REAL: Posición del eje esclavo	REAL: Posición maestra	INT: Número de tabla	VAR REAL[ ]: Resultado de paso	AXIS: Eje esclavo para escalado	AXIS: Eje maestro para escalado	Determina la posición del eje maestro respecto a la posición indicada del eje esclavo a partir de la tabla de levas. Si los parámetros 5/6 no están programados, se calcula con el escalado estándar.
CTABID	INT: Número de tabla de levas	INT: Número de entrada en la memoria	STRING: Ubicación: "SRAM", "DRAM"					Determina el número de tabla de levas que se ha introducido bajo el número indicado en la memoria

Funciones de tablas de levas								
Identificador	**Valor de retorno**	**Parámetro**						**Explicación**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABISLOCK	INT: Estado de bloqueo	INT: Número de tabla						Determina el estado de bloqueo de la tabla de levas: > 0: La tabla está bloqueada 1: CTABLOCK 2: Acoplamiento activo 3: CTABLOCK y acoplamiento activo 0: La tabla no está bloqueada -1: La tabla no existe
CTABEXISTS	INT: Existencia	INT: Número de tabla						Determina la existencia de la tabla de levas en la memoria CN estática o dinámica: 0: FALSE 1: TRUE
CTABMEMTYP	INT: Ubicación	INT: Número de tabla						Determina la ubicación de la tabla de levas: 1: DRAM 0: SRAM -1: La tabla no existe
CTABPERIOD	INT: Periodicidad	INT: Número de tabla						Determina la periodicidad de la tabla de levas: 0: No periódica 1: Periódica en eje maestro 2: Periódica en eje maestro y esclavo -1: La tabla no existe
CTABNO	INT: Número de tablas de levas							Determina el número de las tablas de levas definidas (en la memoria CN estática y dinámica)
CTABNOMEM	INT: Número de tablas de levas	STRING: Ubicación: "SRAM", "DRAM"						Determina el número de tablas de levas definidas en la memoria indicada

Funciones de tablas de levas								
Identificador	**Valor de retorno**	**Parámetro**						**Explicación**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABFNO	INT: Número de tablas	STRING: Ubicación: "SRAM", "DRAM"						Determina el número de tablas de levas aún posibles en la memoria indicada
CTABSEG	INT: Número de segmentos de curvas	STRING: Ubicación: "SRAM", "DRAM"	STRING: Tipo de segmento: "L": Lineal "P": Polinomio					Determina el número de segmentos de curvas utilizados del tipo de segmento indicado en la memoria indicada >=0: Número -1: Tipo de memoria no válido Si no se programa el parámetro 2, se emitirá la suma de los segmentos lineales y de polinomio.
CTABFSEG	INT: Número de segmentos de curvas	STRING: Ubicación: "SRAM", "DRAM"	STRING: Tipo de segmento: "L": Lineal "P": Polinomio					Determina el número de segmentos de curva aún posibles del tipo de segmento indicado en la memoria especificada >=0: Número -1: Tipo de memoria no válido
CTABSEGID	INT: Número de segmentos de curvas	INT: Número de tabla	STRING: Tipo de segmento: "L": Lineal "P": Polinomio					Determina el número de segmentos de curvas del tipo de segmento indicado utilizados por la tabla de levas >=0: Número -1: La tabla no existe
CTABMSEG	INT: Número de segmentos de curvas	STRING: Ubicación: "SRAM", "DRAM"	STRING: Tipo de segmento: "L": Lineal "P": Polinomio					Determina el número máximo de segmentos de curva posibles del tipo de segmento indicado en la memoria especificada >=0: Número -1: La tabla no existe

Funciones de tablas de levas								
Identificador	Valor de retorno	Parámetro						Explicación
		1.	2.	3.	4.	5.	6.	
CTABPOL	INT: Número de polinomios de leva	STRING: Ubicación: "SRAM", "DRAM"						Determina el número de polinomios de leva utilizados en la memoria indicada >=0: Número -1: La tabla no existe
CTABPOLID	INT: Número de polinomios de leva	INT: Número de tabla						Determina el número de polinomios de leva utilizados por la tabla de levas >=0: Número -1: La tabla no existe
CTABFPOL	INT: Número de polinomios de leva	STRING: Ubicación: "SRAM", "DRAM						Determina el número máximo posible de polinomios de leva en la memoria indicada >=0: Número -1: La tabla no existe
CTABMPOL	INT: Número de polinomios de leva	STRING: Ubicación: "SRAM", "DRAM						Determina el número máximo posible de polinomios de leva en la memoria indicada >=0: Número -1: La tabla no existe
CTABSSV	REAL: Posición del eje esclavo	REAL: Posición del eje maestro	INT: Número de tabla	VAR REAL[ ]: Resultado de paso	AXIS: Eje esclavo para escalado	AXIS: Eje maestro para escalado		Determina la posición del eje esclavo al principio del segmento de curva correspondiente al valor de eje maestro indicado.
CTABSEV	REAL: Posición del eje esclavo	REAL: Posición del eje maestro	INT: Número de tabla	VAR REAL[ ]: Resultado de paso	AXIS: Eje esclavo para escalado	AXIS: Eje maestro para escalado		Determina la posición del eje esclavo al final del segmento de curva correspondiente al valor de eje maestro indicado.
CTABTSV	REAL: Posición del eje esclavo	INT: Número de tabla	VAR REAL[ ]: Resultado de paso inicio de la tabla	AXIS: Eje esclavo				Determina la posición del eje esclavo al inicio de la tabla de levas.

Funciones de tablas de levas								
Identificador	**Valor de retorno**	**Parámetro**						**Explicación**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
CTABTEV	REAL: Posición del eje esclavo	INT: Número de tabla	VAR REAL[ ]: Resultado de paso final de la tabla	AXIS: Eje esclavo				Determina la posición del eje esclavo al final de la tabla de levas.
CTABTSP	REAL: Posición del eje maestro	INT: Número de tabla	VAR REAL[ ]: Resultado de paso inicio de la tabla	AXIS: Eje maestro				Determina la posición del eje maestro al inicio de la tabla de levas.
CTABTEP	REAL: Posición del eje maestro	INT: Número de tabla	VAR REAL[ ]: Resultado de paso final de la tabla	AXIS: Eje maestro				Determina la posición del eje maestro al final de la tabla de levas.
CTABTMIN	REAL: Valor mínimo	INT: Número de tabla	REAL: Límite inferior del intervalo de eje maestro	REAL: Límite superior del intervalo de eje maestro	AXIS: Eje esclavo	AXIS: Eje maestro		Determina el valor mínimo del eje esclavo en toda la zona de definición de la tabla de levas o en un intervalo definido.
CTABTMAX	REAL: Valor máximo	INT: Número de tabla	REAL: Límite inferior del intervalo de eje maestro	REAL: Límite superior del intervalo de eje maestro	AXIS: Eje esclavo	AXIS: Eje maestro		Determina el valor máximo del eje esclavo en toda la zona de definición de la tabla de levas o en un intervalo definido.
Nota: Las funciones de tablas de levas se pueden programar también en acciones síncronas.								

Funciones de ejes						
Identificador	**Valor de retorno**	**Parámetro**				
		1.	**2.**	**3.**	**4.**	**Explicación**
AXNAME	AXIS: Identificador de eje	STRING [ ]: Cadena de caracteres de entrada				Conversión de la cadena de caracteres introducida en un identificador de eje
AXSTRING	STRING[ ]: Nombre del eje	AXIS: Identificador de eje				Convierte el identificador de eje en string

Funciones de ejes						
Identificador	**Valor de retorno**	**Parámetro**				
		1.	**2.**	**3.**	**4.**	**Explicación**
ISAXIS	BOOL: Eje disponible (TRUE) o no disponible (FALSE)	INT: Número del eje geométrico (1 a 3)				Comprueba si existe el eje geométrico indicado (1 a 3) atendiendo a la parametrización de los datos de máquina MD20050 $MC_AXCONF_GEOAX_ASSIGN_TAB.
SPI	AXIS: Identificador de eje	INT: Número de cabezal				Conversión del número de cabezal en un identificador de eje
AXTOSPI	INT: Número de cabezal	AXIS: Identificador de eje				Convierte el identificador de eje en un número de cabezal
MODAXVAL	REAL: Valor de módulo	AXIS: Identificador de eje	REAL: Posición de eje			Calcula el resto del módulo a partir de la posición indicada del eje Si el eje indicado no es un eje módulo, se devolverá la posición del eje inalterada.
POSRANGE	BOOL: Posición nominal dentro de la ventana de posición (TRUE) o no (FALSE)	AXIS: Identificador de eje	REAL: Posición de referencia en el sistema de coordenadas	REAL: Anchura de la ventana de posición	INT: Sistema de coordenadas	Calcula si la posición nominal de un eje se encuentra en una ventana en torno a una posición de referencia predefinida

Gestión de herramientas					
Identificador	**Valor de retorno**	**Parámetro**			**Explicación**
		1.	**2.**	**3.**	
CHKDM	INT: Estado: resultado de la comprobación	INT: Número de almacén	INT: Número D		Comprueba la unicidad del número D dentro de un almacén
CHKDNO	INT: Estado: resultado de la comprobación	INT: Número T de la 1.ª herramienta	INT: Número T de la 2.ª herramienta	INT: Número D	Comprueba la unicidad del número D
GETACTT	INT: Estado	INT: Número T	STRING [32]: Nombre de herramienta		Determina la herramienta activa a partir de un grupo de herramientas con el mismo nombre
GETACTTD	INT: Estado: resultado de la comprobación	VAR INT: N.º T encontrado (valor de retorno)	INT: Número D		Determina el número T correspondiente para el número D absoluto

Gestión de herramientas					
Identificador	**Valor de retorno**	**Parámetro**			**Explicación**
		1.	**2.**	**3.**	
GETDNO	INT: Número D	INT: Número T	INT: Número de filo		Determina el número D del filo de la herramienta T
GETT	INT: Número T	STRING [32]: Nombre de herramienta	INT: Número Duplo		Determina el número T del nombre de herramienta
NEWT	INT: Número T	STRING [32]: Nombre de herramienta	INT: Número Duplo		Crea una nueva herramienta (preparar datos de herramienta) El número duplo puede omitirse.
TOOLENV	INT: Estado	STRING: Nombre			Guarda el entorno de herramienta con el nombre indicado en la memoria CN estática
DELTOOLENV	INT: Estado	STRING: Nombre			Borra el entorno de herramienta con el nombre indicado en la memoria CN estática Borra todos los entornos de herramienta si no se indica ningún nombre.
GETTENV	INT: Estado	STRING: Nombre	VAR INT: Número T [0] Número D [1] Número DL [2]		Determina el número T, el número D y el número DL de un entorno de herramienta con un nombre indicado

Aritmética					
Identificador	**Valor de retorno**	**Parámetro**			**Explicación**
		1.	**2.**	**3.**	
SIN	REAL	REAL			Seno
ASIN	REAL	REAL			Arcoseno
COS	REAL	REAL			Coseno
ACOS	REAL	REAL			Arcocoseno
TAN	REAL	REAL			Tangente
ATAN2	REAL	REAL	REAL		Arcotangente 2
SQRT	REAL	REAL			Raíz cuadrada
POT	REAL	REAL			Cuadrado
TRUNC	REAL	REAL			Parte entera
ROUND	REAL	REAL			Redondear a la baja
ROUNDUP	REAL	REAL			Redondear al alza
ABS	REAL	REAL			Valor absoluto
LN	REAL	REAL			Logaritmo neperiano (logaritmo natural)
EXP	REAL	REAL			Función exponencial e^x
MINVAL	REAL	REAL	REAL		Determina el valor menor de los dos parámetros

Aritmética					
Identificador	Valor de retorno	Parámetro			Explicación
		1.	2.	3.	
MAXVAL	REAL	REAL	REAL		Determina el valor mayor de los dos parámetros
BOUND	REAL: Estado de comprobación	REAL: Límite inferior	REAL: Límite superior	REAL: Valor de referencia	Determina si el valor de referencia se encuentra dentro de los límites.
Nota: Las funciones aritméticas se pueden programar también en acciones síncronas. El cálculo y la evaluación de estas funciones aritméticas se realiza entonces en la marcha principal. Para cálculos y como memoria intermedia también se puede utilizar el parámetro de acciones síncronas $AC_PARAM [<n>].					

Funciones String					
Identificador	Valor de retorno	Parámetro			Explicación
		1.	2.	3.	
ISNUMBER	BOOL	STRING: Cadena de caracteres de entrada			Comprueba si la cadena de caracteres de entrada puede convertirse en un número
NUMBER	REAL	STRING: Cadena de caracteres de entrada			Convierte la cadena de caracteres introducida en un número
TOUPPER	STRING	STRING: Cadena de caracteres de entrada			Convierte la cadena de caracteres de entrada a letras mayúsculas
TOLOWER	STRING	STRING: Cadena de caracteres de entrada			Convierte la cadena de caracteres de entrada a letras minúsculas
STRLEN	INT	STRING: Cadena de caracteres de entrada			Determina la longitud de la cadena de caracteres de entrada hasta el final de la cadena (/0)
INDEX	INT	STRING: Cadena de caracteres de entrada	CHAR: Carácter de búsqueda		Determina la posición del carácter en la cadena de caracteres de entrada de izquierda a derecha. El 1.er carácter de la cadena por la izquierda tiene el índice 0.
RINDEX	INT	STRING: Cadena de caracteres de entrada	CHAR: Carácter de búsqueda		Determina la posición del carácter en la cadena de caracteres de entrada de derecha a izquierda. El 1.er carácter de la cadena por la derecha tiene el índice 0.

Funciones String					
Identificador	**Valor de retorno**	**Parámetro**			**Explicación**
		1.	**2.**	**3.**	
MINDEX	INT	STRING: Cadena de caracteres de entrada	STRING: Carácter de búsqueda		Determina la posición de uno de los caracteres indicados en el 2.º parámetro en la cadena de caracteres de entrada de izquierda a derecha. El 1.er carácter de la cadena de caracteres de entrada por la izquierda tiene el índice 0.
SUBSTR	STRING	STRING: Cadena de caracteres de entrada	INT	INT	Determina la parte de la cadena de caracteres de entrada descrita mediante el carácter de inicio (2.º parámetro) y el número de caracteres (3.er parámetro).
SPRINT	STRING	STRING: Cadena de caracteres de entrada			Determina la cadena de caracteres de entrada formateada

Funciones de los ciclos de medida								
Identificador	Valor de retorno	Parámetro						Explicación
		1.	2.	3.	4.	5.	6.	
CALCPOSI	INT: Estado	REAL[3]: Posición de salida en el WKS	REAL[3]: Especificación de recorrido incremental con respecto a la posición de salida	REAL[5]: Distancias mínimas con los límites de vigilancia	REAL[3]: Valor de retorno para el posible recorrido incr.	BOOL: Conversión del sistema de medidas sí/no	INT: Tipo de vigilancia de límites	Comprueba si, a partir de un punto inicial establecido, los ejes geométricos pueden desplazarse por un trayecto especificado sin vulnerar sus límites. En caso de que el recorrido determinado no se pueda ejecutar sin vulneraciones, se devuelve el valor máximo admisible.
GETTCOR	INT: Estado	REAL [11]:	STRING: Componentes longitudinales de herramienta: Sistema de coordenadas	STRING: Nombre del entorno de herr.	INT: N.º T interno de la herramienta	INT: Número de filo (n.º D) de la herramienta	INT: Número de la corrección dependiente de la posición (n.º DL de la herram.)	Determina las longitudes de herramienta y los componentes longitudinales de herramienta del entorno de herramienta o del entorno actual.
LENTOAX	INT: Estado	INT[3]: Asignación de ejes geométricos	REAL[3]: Matriz para la representación de las longitudes de herramienta en el sistema de coordenadas	STRING: Sistema de coordenadas para la asignación				Proporciona información sobre la asignación de las longitudes de herramienta L1, L2, L3 de la herramienta actual con abscisa, ordenada y aplicada. La asignación a los ejes geométricos se modifica a través de frames y del plano activo (G17-G19).

SETTCOR	INT: Estado	1.	2.	3.	4.	5.	6.	7.	8.	9.	
		REAL [3]: Vector de corrección en el espacio	STR.: Identificador de componente	INT: Componente(s) para corregir 0 - 11	INT: Tipo de operaciones de escritura 0 - 3	INT: Índice del eje geométrico	STRING: Nombre del entorno de herramienta	INT: N.º T int. de la herram.	INT: N.º D de la herram.	INT: N.º DL de la herram.	Modifica componentes de herramienta teniendo en cuenta todas las condiciones que afectan a la evaluación de los diferentes componentes.

Otras funciones								
Identificador	Valor de retorno	Parámetro						Explicación
		1.	2.	3.	4.	5.	6.	
STRINGIS	INT: Información sobre el string	STRING: Nombre del elemento para comprobar						Comprueba si el string indicado como elemento del lenguaje de programación CN está disponible en el repertorio actual del lenguaje
ISVAR	BOOL: Variable conocida sí/no	STRING: Nombre de las variables						Comprueba si el parámetro de transferencia contiene una variable conocida en el CN (dato de máquina, dato de operador, variable de sistema, variables generales como GUD)
GETVARTYP	INT: Tipo de datos	STRING: Nombre de las variables						Determina el tipo de datos de una variable de sistema o usuario.
GETVARPHU	INT: Valor numérico de la unidad física	STRING: Nombre de las variables						Determina la unidad física de una variable de sistema o usuario.
GETVARAP	INT: Nivel de protección para el acceso	STRING: Nombre de las variables	STRING: Tipo de acceso					Determina el derecho de acceso a una variable de sistema o usuario.

Otras funciones								
Identificador	Valor de retorno	Parámetro						Explicación
		1.	2.	3.	4.	5.	6.	
GETVARLIM	INT: Estado	STRING: Nombre de las variables	CHAR: Indica qué valor límite debe leerse	VAR REAL: Devuelve el valor límite				Determina el valor límite inferior/superior de una variable de sistema o usuario.
GETVARDFT	INT: Estado	STRING: Nombre de las variables	VAR REAL/ STRING/ FRAME: Devuelve el valor estándar	INT: Índice en la primera dimensión (opcional)	INT: Índice en la segunda dimensión (opcional)	INT: Índice en la tercera dimensión (opcional)		Determina el valor estándar de una variable de sistema o usuario.
COLLPAIR	INT: Resultado de comprobación	STRING: Nombre de la 1.ª zona protegida	STRING: Nombre de la 2.ª zona protegida	BOOL: Supresión de alarmas (opcional)				Comprueba la pertenencia a un par de colisión.
PROTD	REAL: Distancia de las dos zonas protegidas	STRING: Nombre de la 1.ª zona protegida	STRING: Nombre de la 2.ª zona protegida	VAR REAL: Valor de retorno: Vector tridimensional de distancia	BOOL: Sistema de medida para la distancia y el vector de distancia (opcional)			Determina la distancia de las dos zonas protegidas indicadas.
DELOBJ	INT: Número de error	STRING: Tipo de componente que se desea eliminar	INT: Índice de inicio de los componentes que se desean eliminar (opcional)	INT: Índice de fin de los componentes que se desean eliminar (opcional)	BOOL: Supresión de alarmas (opcional)			Borra elementos de cadenas cinemáticas, zonas protegidas, elementos de zonas protegidas, pares de colisión y datos de transformadas.
NAMETOINT	INT: Índice de variable de sistema	STRING: Nombre del elemento de matriz de sistema	STRING: Cadena de caracteres/ Nombre	BOOL: Supresión de alarmas (opcional)				Determina el índice de variable de sistema correspondiente a partir de la cadena de caracteres

Otras funciones								
Identificador	**Valor de retorno**	**Parámetro**						**Explicación**
		1.	**2.**	**3.**	**4.**	**5.**	**6.**	
ORISOLH	INT: Número de error	INT: Controla el comportamiento de la función	REAL: Primer ángulo	REAL: Segundo ángulo				Ayuda al usuario a ajustar las posiciones de los ejes giratorios de una máquina de modo que una herramienta de tornear se lleve a una posición definida relativa a la herramienta e independiente de la cinemática. Requisitos: Está activa una transformada de 6 ejes parametrizada con cadenas cinemáticas.
CORRTRAFO	INT: Número de error	REAL: Vector de corrección	INT: Elemento que se va a modificar	INT: Modo de corrección	BOOL: Supresión de alarmas (opcional)			Modificar los vectores de offset o los vectores de dirección de los ejes de orientación en el modelo cinemático de la máquina.
CORRTC	INT: Número de error	REAL: Vector de corrección	INT: Elemento que se va a modificar	INT: Modo de corrección	BOOL: Supresión de alarmas (opcional)			Modificar los vectores de offset o los vectores de dirección de los portaherramientas orientables después de la medición de la máquina.

4.7 Idioma actual del HMI

La siguiente tabla contiene todos los idiomas disponibles en la interfaz de usuario.

El idioma ajustado actualmente se puede consultar en el programa de pieza y en acciones síncronas mediante la siguiente variable del sistema:

$AN_LANGUAGE_ON_HMI = <Valor>

<Valor>	Idioma	Código de idioma
1	Alemán (Alemania)	DEU
2	Francés	FRA
3	Inglés (Reino Unido)	ENG
4	Español	ESP
6	Italiano	ITA
7	Holandés	NLD
8	Chino (simplificado)	CHS
9	Sueco	SVE
18	Húngaro	HUN
19	Finés	FIN
28	Checo	CSY
50	Portugués (Brasil)	PTB
53	Polaco	PLK
55	Danés	DAN
57	Ruso	RUS
68	Eslovaco	SKY
72	Rumano	ROM
80	Chino (tradicional)	CHT
85	Coreano	KOR
87	Japonés	JPN
89	Turco	TRK

Nota

$AN_LANGUAGE_ON_HMI se actualiza:

- después del arranque del sistema;
- después de reset del CN o del PLC.
- después de conmutar a otro CN dentro de M2N.
- después del cambio de idioma a HMI.

Anexo: A

A.1 Lista de abreviaturas

A	
A	Salida
ADI4	Analog Drive Interface for 4 Axes
AC	Adaptive Control
ALM	Active Line Module
ARM	Motor asíncrono rotativo
AS	Sistema de automatización
ASCII	American Standard Code for Information Interchange: código estándar americano para el intercambio de la información
ASIC	Application Specific Integrated Circuit: circuito integrado del usuario
ASUP	Subprograma asíncrono
AUXFU	Auxiliary Function: función auxiliar
AWL	Lista de instrucciones
AWP	Programa de usuario

B	
BA	Modo de operación
BAG/GMO	Grupo de modos de operación
BCD	Binary Coded Decimals: decimales codificados en binario
BERO	Serie de detectores de proximidad
BI	Binector Input
BICO	Binector Connector
BIN	Binary Files: fichero binarios
BIOS	Basic Input Output System
BKS	Sistema de coordenadas básico
BO	Binector Output
BTSS	Interfaz de panel de operador

C	
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CC	Compile Cycle: ciclos compilados
CEC	Cross Error Compensation
CI	Connector Input
CNC	Computerized Numerical Control: Control numérico computerizado

C	
CO	Connector Output
CoL	Certificado de licencia
COM	Comunicación
CPA	Compiler Projecting Data: datos de configuración del compilador
CRT	Cathode Ray Tube: pantalla catódica
CSB	Central Service Board: tarjeta del PLC
CU	Control Unit
CP	Communication Processor
CPU	Central Processing Unit: unidad central de proceso
CR	Carriage Return
CST	Configured Stop: parada configurada
CTS	Clear To Send: código de "Preparado para enviar" en interfaces serie de datos
CUTCOM	Cutter Radius Compensation: corrección de radio de herramienta

D	
DAU	Convertidor digital-analógico
DB	Bloque de datos (PLC)
DBB	Byte de bloque de datos (PLC)
DBD	Palabra doble de bloque de datos (PLC)
DBW	Palabra de bloque de datos (PLC)
DBX	Bit de bloque de datos (PLC)
DDE	Dynamic Data Exchange
DDS	Drive Data Set: juego de datos de accionamiento
DIN	Deutsche Industrie Norm: Norma Industrial Alemana
DIO	Data Input/Output: señalización en la pantalla para la transmisión de datos
DIR	Directory: directorio
DLL	Dynamic Link Library
DO	Drive Object
DPM	Dual Port Memory
DPR	Dual Port RAM
DRAM	Memoria dinámica (sin respaldo)
DRF	Differential Resolver Function: función resolver diferencial (volante)
DRIVE-CLiQ	Drive Component Link with IQ
DRY	Dry Run: avance de recorrido de prueba
DSB	Decoding Single Block: decodificación secuencia a secuencia
DSC	Dynamic Servo Control/Dynamic Stiffness Control
DW	Palabra de datos
DWORD	Palabra doble (actualmente 32 bits)

E	
E	Entrada
EES	Execution from External Storage
E/S	Entrada/Salida
ENC	Encoder: captador de posición real
EFP	Módulo de periferia sencillo (módulo de E/S PLC)
ESD (ESDS)	Componentes/tarjetas sensibles a descargas electrostáticas
CEM	Compatibilidad electromagnética
EN	Norma europea
ENC	Encoder: captador de posición real
EnDat	Interfaz de encóder
EPROM	Erasable Programmable Read Only Memory: memoria de solo lectura borrable y programable eléctricamente
ePS Network Services	Servicios de mantenimiento remoto de máquinas por Internet
EQN	Denominación de tipo para un encóder absoluto con 2048 señales senoidales por vuelta
ES	Engineering System
ESR	Parada y retirada ampliada
ETC	Tecla ETC ">"; ampliación del menú de pulsadores en el mismo menú

F	
FB	Bloque de función (PLC)
FC	Function Call: bloque de función (PLC)
FEPROM	Flash-EPROM: memoria de lectura y escritura
FIFO	First In First Out: Tipo de almacenamiento en memoria sin direccionamiento, en el cual los datos son leídos en el mismo orden en el que fueron almacenados
FIPO	Interpolador fino
FPU	Floating Point Unit: unidad de coma flotante
FRK	Corrección del radio de la fresa
FST	Feed Stop: Parada de avance
FUP	Esquema de funciones (método de programación para PLC)
FW	Firmware

G	
GC	Global Control (PROFIBUS: telegrama Broadcast)
GDIR	Memoria global de programas de pieza
GEO	Geometría, p. ej., eje geométrico
GIA	Gear Interpolation Data: datos de interpolación del reductor
GND	Signal Ground
GP	Programa básico (PLC)
GS	Escalón de reducción
GSD	Fichero de datos del equipo que describe un esclavo PROFIBUS

G	
GSDML	Generic Station Description Markup Language: lenguaje descriptivo basado en XML para crear un fichero GSD
GUD	Global User Data: datos globales del usuario

H	
HEX	Abreviatura para número hexadecimal
HiFu	Función auxiliar
HLA	Accionamiento hidráulico lineal
HMI	Human Machine Interface: interfaz de usuario SINUMERIK
HSA	Accionamiento de cabezal
HW	Hardware

I	
PeM	Puesta en marcha
IKA	Compensación interpolatoria
IM	Interface-Modul: módulo de conexión
IMR	Interface-Modul Receive: módulo de conexión para el servicio de recepción
IMS	Interface-Modul Send: módulo de conexión para el servicio de transmisión
INC	Increment: acotado incremental
INI	Initializing Data: datos de inicialización
IPO	Interpolador
ISA	International Standard Architecture
ISO	International Standard Organization

J	
JOG	Jogging: manejo convencional para preparación (manual)

K	
K_V	Ganancia del lazo de regulación
K_P	Ganancia proporcional
$K_Ü$	Relación de transmisión
KOP	Esquema de contactos (método de programación para PLC)

L	
LAI	Logic Machine Axis Image: imagen de proceso lógica de los ejes de la máquina
LAN	Local Area Network
LCD	Liquid-Crystal Display: pantalla de cristal líquido
LED	Light Emitting Diode: diodo luminiscente
LF	Line Feed

L	
LMS	Sistema de medida de la posición
LR	Regulador de posición
LSB	Least Significant Bit: bit menos significativo
LUD	Local User Data: datos del usuario (locales)

M	
MAC	Media Access Control
MAIN	Main program: programa principal (OB1, PLC)
MB	Megabyte
MCI	Motion Control Interface
MCIS	Motion Control Information System
MCP	Machine Control Panel: panel de mando de máquina
MD	Dato o datos de máquina
MDA	Manual Data Automatic: introducción de programa manual
MDS	Motor Data Set: juego de datos de motor
MELDW	Palabra de señalización
MKS	Sistema de coordenadas de máquina
MM	Motor Module
MPF	Main Program File: programa principal (CN)
MSTT	Panel de mando de máquina

N	
CN	Numerical Control: control numérico (CN) con procesamiento de secuencias, zona de desplazamiento, etc.
NCU	Numerical Control Unit: unidad de hardware del CN
NRK	Denominación del sistema operativo del CN
NST	Señal de interfaz
NURBS	Non-Uniform Rational B-Spline
NV	Decalaje de origen
NX	Numerical Extension: módulo de ampliación de eje

O	
OB	Bloque de organización en el PLC
OEM	Original Equipment Manufacturer
OP	Operation Panel: dispositivo de operación
OPI	Operation Panel Interface: conexión para el panel de operador
OPT	Options: opciones
OLP	Optical Link Plug: conector de bus para fibra óptica
OSI	Open Systems Interconnection: Normalización para la comunicación con ordenadores

P	
PAA	Imagen de proceso de las salidas
PAE	Imagen de proceso de las entradas
PC	Ordenador personal
PCIN	Nombre del software para el intercambio de datos con el control
PCMCIA	Personal Computer Memory Card International Association: normalización para tarjetas de memoria
PCU	PC Unit: PC-Box (caja central)
PG	Programadora
PKE	Identificador del parámetro: es parte del PKW
PKW	Identificador del parámetro: valor (parte de parametrización de un PPO)
PLC	Programmable Logic Control: mando de interconexión
PN	PROFINET
PNO	PROFIBUS User Organisation (organización de usuarios de PROFIBUS)
PO	POWER ON
POE	Unidad de organización de programa
POS	Posición/Posicionar
POSMO A	Positioning Motor Actuator: motor de posicionamiento
POSMO CA	Positioning Motor Compact AC: unidad de accionamiento completa con módulo de potencia y regulación integrado, unidad de posicionamiento y memoria de programa; alimentación de corriente alterna
POSMO CD	Positioning Motor Compact DC: como CA, pero alimentación de corriente continua
POSMO SI	Positioning Motor Servo Integrated: motor de posicionamiento; alimentación de corriente continua
PPO	Parámetro datos de proceso de objeto; telegrama cíclico de datos durante la transmisión a través del PROFIBUS-DP con el perfil "Accionamientos de velocidad variable"
PPU	Panel Processing Unit (hardware central de un control CNC basado en panel, p. ej., SINUMERIK 828D)
PROFIBUS	Process Field Bus: bus de datos serie
PRT	Test del programa
PSW	Palabra de mando de programa
PTP	Point to Point: punto a punto
PUD	Program Global User Data: Variables de usuario globales de programa
PZD	Datos de proceso: parte de datos del proceso de un PPO

Q	
QFK	Compensación de errores de cuadrante

R	
RAM	Random Access Memory: memoria de escritura y lectura
REF	Función "Aproximación al punto de referencia"
REPOS	Función "Reposicionamiento"

R	
RISC	Reduced Instruction Set Computer: tipo de procesador con juego de instrucciones reducido y tiempos de elaboración muy cortos
ROV	Rapid Override: corrección de entrada
RP	Parámetros R, parámetros de cálculo, variable de usuario predefinida
RPA	R-Parameter Active: Zona de memoria en el CN para los números de los parámetros R (parámetros de cálculo)
RPY	Roll Pitch Yaw: modo de giro de un sistema de coordenadas
RTLI	Rapid Traverse Linear Interpolation: interpolación lineal con desplazamiento en rápido
RTS	Request To Send: conexión de la unidad de transmisión, señal de control de interfaces serie de datos
RTCP	Real Time Control Protocol

S	
SA	Acción síncrona
SBC	Safe Brake Control: mando de freno seguro
SBL	Single Block: secuencia a secuencia
SBR	Subroutine: subprograma (PLC)
SD	Dato o datos de operador
SDB	Bloque de sistema
SEA	Setting Data Active: identificación (tipo de fichero) para datos de operador
SERUPRO	Search-Run by Program Test: Búsqueda de secuencia vía test del programa
SFB	Bloque de función del sistema
SFC	System Function Call
SH	Parada segura
SIM	Single in Line Module
SK	Pulsador de menú
SKP	Skip: función para ocultar una secuencia de programa de pieza
SLM	Motor síncrono lineal
SM	Motor paso a paso
SMC	Sensor Module Cabinet Mounted
SME	Sensor Module Externally Mounted
SMI	Sensor Module Integrated
SPF	Sub Program File: subprograma (CN)
PLC	Autómata programable = PLC
SRAM	Memoria estática (con respaldo)
SRK	Corrección del radio del filo
SRM	Motor síncrono rotativo
SSFK	Corrección del error del paso de husillo
SSI	Serial Synchron Interface: interfaz serie síncrona
SSL	Búsqueda de número de secuencia
STW	Palabra de mando
SUG	Velocidad periférica de muela
SW	Software

S	
SYF	System Files: ficheros de sistema
SYNACT	Synchronized Action: acción síncrona

T	
TB	Terminal Board (SINAMICS)
TCP	Tool Center Point: punta de la herramienta
TCP/IP	Transport Control Protocol/Internet Protocol
TCU	Thin Client Unit
TEA	Testing Data Active: identificación de datos de máquina
TIA	Totally Integrated Automation
TM	Terminal Module (SINAMICS)
TO	Tool Offset: corrección de herramienta
TOA	Tool Offset Active: identificación de correcciones de herramienta (tipo de fichero)
TRANSMIT	Transform Milling into Turning: transformación de coordenadas para fresados en un torno
TTL	Lógica transistor–transistor (tipo interfaces)
TZ	Ciclo tecnológico

U	
UFR	User Frame: decalaje de origen
UP	Subprograma
USB	Universal Serial Bus
SAI	Sistema de alimentación ininterrumpida

V	
VDI	Interfaz de comunicación interna entre CN y PLC
VDI	Verein Deutscher Ingenieure
VDE	Verband Deutscher Elektrotechniker
VI	Voltage Input
VO	Voltage Output
VSA	Accionamiento de avance

W	
WAB	Función de aproximación y retirada suaves
WKS	Sistema de coordenadas de pieza
WKZ	Herramienta
WLK	Corrección longitudinal de herramienta
WOP	Programación orientada al taller
WPD	Work Piece Directory: directorio de piezas
WRK	Corrección de radio de herramienta

W	
WZ	Herramienta
WZK	Corrección de herramienta
WZV	Gestión de herramientas
WZW	Cambio de herramienta

X	
XML	Extensible Markup Language

Z	
ZOA	Zero Offset Active: identificación de decalajes de origen
ZSW	Palabra de estado (del accionamiento)

A.2 Vista general de la documentación

Una amplia documentación sobre las funciones de SINUMERIK 840D sl a partir de la versión 4.8 SP4 se encuentra en Vista general de la documentación 840D sl (https://support.industry.siemens.com/cs/ww/en/view/109766213).

Tiene la posibilidad de ver los documentos o descargarlos en formato PDF y HTML5.

La documentación se divide en las siguientes categorías:

- Usuario: Manejo
- Usuario: Programar
- Fabricante/servicio: Configuración
- Fabricante/servicio: Puesta en marcha
- Fabricante/servicio: Funciones
- Safety Integrated
- SINUMERIK Integrate / MindApp
- Información y formación

Índice alfabético

$

$A_PROBE, 631, 638
$A_PROBE_LIMITED, 639
$AA_ACC, 131
$AA_ATOL, 896
$AA_COUP_ACT
  con acoplamiento axial de valores maestros, 946
  en el arrastre de ejes, 924
$AA_ESR_ENABLE, 1010
$AA_FGREF, 115
$AA_FGROUP, 115
$AA_G0MODE, 187
$AA_LEAD_SP, 946
$AA_LEAD_SV, 946
$AA_MM, 632
$AA_MM1...4, 639
$AA_MW, 632
$AA_MW1...4, 639
$AC_ACT_PROG_NET_TIME, 1018
$AC_ACTUAL_PARTS, 1020
$AC_AXCTSWA, 912
$AC_AXCTSWE, 913
$AC_CTOL, 896
$AC_CTOL_G0_ABS, 190
$AC_CUT_INV, 831
$AC_CUTMOD, 830
$AC_CUTMOD_ANG, 830
$AC_CUTMODK, 830
$AC_CUTTING_TIME, 1017
$AC_CYCLE_TIME, 1017
$AC_DELAYFST, 880
$AC_ESR_TRIGGER, 1010
$AC_F_TYPE, 146
$AC_FGROUP_MASK, 115
$AC_FZ, 146
$AC_MEA, 631, 638
$AC_OLD_PROG_NET_TIME, 1018
$AC_OLD_PROG_NET_TIME_COUNT, 1018
$AC_OPERATING_TIME, 1017
$AC_OTOL, 896
$AC_OTOL_ G0_ABS, 191
$AC_PROG_NET_TIME_TRIGGER, 1018
$AC_REPOS_PATH_MODE, 888
$AC_REQUIRED_PARTS, 1020
$AC_S_TYPE, 100
$AC_SMAXVELO, 892
$AC_SMAXVELO_INFO, 892
$AC_SPECIAL_PARTS, 1020
$AC_STOLF, 190
$AC_SVC, 100
$AC_TOFF, 89
$AC_TOFFCR, 89
$AC_TOFFL, 89
$AC_TOFFR, 89
$AC_TOTAL_PARTS, 1020
$AC_TRAFO_CORR_ELEM_P, 758
$AC_TRAFO_CORR_ELEM_T, 758
$AC_TRAFO_ORIAX_LOC, 759
$AN_AXCTAS, 913
$AN_AXCTSWA, 912
$AN_ESR_TRIGGER, 1010
$AN_LANGUAGE_ON_HMI, 1355
$AN_POWER ON_TIME, 1017
$AN_SETUP_TIME, 1017
$NT_CLOSE_CHAIN_T, 759
$NT_CNTRL, 758
$NT_CORR_ELEM_P, 758
$NT_CORR_ELEM_T, 758
$NT_NAME, 751
$NT_ROT_AX_NAME, 826
$NT_TRAFO_INDEX, 751
$P_ACTBFRAME, 669
$P_AD, 830
$P_AEP, 288
$P_APDV, 288
$P_APR, 288
$P_BFRAME, 669
$P_CHBFRAME, 669
$P_CHBFRMASK, 669
$P_CTOL, 897
$P_CTOL_ G0_ABS, 191
$P_CUT_INV, 831
$P_CUTMOD, 830
$P_CUTMOD_ANG, 830
$P_CUTMOD_ERR, 831
$P_CUTMODK, 830
$P_DELAYFST, 880
$P_F_TYPE, 146
$P_FGROUP_MASK, 116
$P_FZ, 146
$P_GWPS, 107
$P_IFRAME, 670
$P_IS_EES_PATH, 578
$P_NCBFRAME, 668
$P_NCBFRMASK, 669
$P_ORI_DIFF, 821

$P_ORI_POS, 821
$P_ORI_SOL, 822
$P_ORI_STAT, 825
$P_OTOL, 897
$P_OTOL_ G0_ABS, 191
$P_PATH, 577
$P_PFRAME, 670
$P_PROG, 577
$P_PROGPATH, 577
$P_S_TYPE, 100
$P_SIM, 641
$P_STACK, 577
$P_STOLF, 190
$P_SUBPAR, 510
$P_SVC, 100
$P_TOFF, 89
$P_TOFFCR, 89
$P_TOFFL, 89
$P_TOFFR, 89
$P_TOOLENV, 840
$P_TOOLENVN, 840
$P_WORKAREA_CS_COORD_SYSTEM, 365
$P_WORKAREA_CS_LIMIT_MINUS, 366
$P_WORKAREA_CS_LIMIT_PLUS, 366
$P_WORKAREA_CS_MINUS_ENABLE, 365
$P_WORKAREA_CS_PLUS_ENABLE, 365
$PA_ATOL, 897
$PA_FGREF, 115
$PA_FGROUP, 116
$SA_LEAD_TYPE, 946
$SC_CONTPREC, 874
$SC_MINFEED, 874
$SC_PA_ACTIV_IMMED, 598
$SN_PA_ACTIV_IMMED, 598
$TC_CARR_CORR_ELEM, 813
$TC_CARR1...14, 802
$TC_CARR18...23, 802
$TC_CARR18[m], 806
$TC_DP1 ... 25, 760
$TC_ECPxy, 764
$TC_SCPxy, 764
$TC_TP_MAX_VELO, 96

*****

* (Función de cálculo), 454

/

/ (Función de cálculo), 454

+

+ (Función de cálculo), 454

<

< (Operador de comparación), 456
<< (operador de concatenación), 463
<= (Operador de comparación), 456
<> (Operador de comparación), 456

=

== (Operador de comparación), 456

>

> (Operador de comparación), 456
>= (Operador de comparación), 456

A

ABS, 454
AC, 153
ACC, 130
ACCLIMA, 870
ACN, 160
Acoplamiento
  genérico, 965
acoplamiento de valores de consigna, 957
Acoplamiento de valores maestros
  Acoplamiento de valores reales y de valores de consigna, 945
  Sincronización eje maestro y esclavo, 944
Acoplamiento de valores reales, 957
Acoplamiento de velocidad, 957
Acoplamiento entre ejes de valores maestros, 942
ACOS, 454
Acotado
  en el diámetro, 166
  en el radio, 166
  para ejes giratorios y cabezales, 160
  posibilidades, 152
Acotado absoluto, 30
Acotado incremental, 32, 155
ACP, 160
ACTBLOCNO, 521
ACTFRAME, 648

Adaptar a la carga: CYCLE782
programación externa, 1129
ADIS, 304
ADISPOS, 304
ADISPOSA, 643
Agujero rasgado, LONGHOLE
programación externa, 1076
Alarmas
activar en el programa CN, 1035
ALF
para retirada rápida del contorno, 563
para retirada rápida durante tallado de roscas, 236
Alisado
del desarrollo de la orientación, 719
AMIRROR, 337
AND, 456
ANG, 216
ANG1, 216
ANG2, 216
Ángulo euleriano, 690
Ángulo polar, 30
Ángulo RPY, 690
AP, 179
APR, 424
APRB, 424
APRP, 424
APW, 424
APWB, 424
APWP, 424
AR
programación de círculos, 199
Archivo
información, 589
AROT, 325
AROTS, 331
Arrastre de ejes, 921
AS, 502
ASCALE, 334
Asignación del valor, 48
ASIN, 454
ASPLINE, 610
ATAN2, 454
ATOL, 893
ATRANS, 319
Atributos de posición
programar indirectamente, 450
AV, 954
Avance
con corrección por volante, 131
corrección, 129, 133
inverso al tiempo, 111
para ejes de contorneado, 110
para ejes de posicionado, 126
para ejes síncronos, 112
regulación, 108
sistema de unidades, 113
velocidad, 191
Avisos, 358
AX, 905
AXCTSWE, 911
AXCTSWEC, 911
AXCTSWED, 911
AXIS, 407
AXNAME, 462
AXSTRING, 905
AXTOCHAN, 903
AXTOSPI, 905

B

B_AND, 456
B_NOT, 456
B_OR, 456
B_XOR, 456
BAUTO, 610
BFRAME, 648
BKS, 38
BLOCK, 546
BLSYNC, 557
BNAT, 610
BNS, 39
BOOL, 407
BOUND, 437
BRISK, 868
BRISKA, 868
Broca, 74
BSPLINE, 610
BTAN, 610
Bucle
Bucle contador, 493
Bucle final, 492
Bucle IF, 490
Bucle REPEAT, 495
Bucle WHILE, 494
Bucle contador, 493
Bucle sin fin, 492
Búsqueda de punto de referencia, 367

C

Cabezal
Funciones M, 356

intercambio, 899
limitación de la velocidad de giro, 107
Posicionar, 121
principal, 382
Sentido de giro, 91
Velocidad de giro, 91
Cabezal maestro, 382
Cabezal síncrono
acoplamiento, 953
definición de un par, 960
CAC, 609
CACN, 609
CACP, 609
CADAPTOF, 915
CADAPTON, 915
Cadena de herramientas, 757
Cadena de piezas, 757
Cadena de roscas, CYCLE98
programación externa, 1118
Caja circular, POCKET4
programación externa, 1068
Caja rectangular, POCKET3
programación externa, 1066
Cálculo del recorrido, 389
CALL, 545
CALLPATH, 549
Campo
definición, 430
elemento, 430
Canal
ejes, 383
Carácter 0, 461
Caracteres especiales, 52, 53
CASE, 481
case-insensitive, 570
CDC, 609
CDOF, 292
CDOF2, 292
CDON, 292
CFC, 135
CFIN, 135
CFINE, 657
CFTCP, 135
Chaflán, 251
CHAN, 407
CHANDATA, 578
CHAR, 407
CHF, 251
CHKDNO, 798
CHR, 251
CIC, 609
Cinemática
Simple, 806
CIP, 203
CLEARM, 496
CLRINT, 559
COARSE, 954
COARSEA, 643
Código G
programar indirectamente, 448
Coeficiente de polinomio, 623
COLLPAIR, 746
Comando, 44
Ejes, 385
Comandos G, 1293
Comentarios, 48
COMPCAD, 621
COMPCURV, 621
COMPLETE, 578
COMPOF, 621
COMPON, 621
Componente frame
FI, 653
MI, 653
RT, 653
SC, 653
TR, 653
COMPSURF, 621
Concatenación
de cadenas, 463
Conjunto de ejes, 921
Conjunto spline, 620
Constante, 394
Constante decimal, 394
Constante INTEGER, 394
Constante REAL, 394
Constantes binarias, 395
Constantes hexadecimales, 394
CONTDCON, 1050
Contorno
aproximación/retirada, 266
calculadora, 217
codificación, 1050
elemento, 174
preparación, 1044
reposicionamiento, 882
tabla, 1044
CONTPRON, 1044
Control
estructuras, 489
Coordenadas
Cartesianas, 28

cilíndricas, 180
polares, 29
Coordenadas cartesianas, 28
Coordenadas cilíndricas, 180
Coordenadas polares, 29
Corrección
longitudes de herramienta, 68
plano, 296
radio de la herramienta, 69
Corrección de herramienta
Memoria de corrección, 760
offset, 85
Sistema de coordenadas para valores de desgaste, 768
Corrección de radio de herramienta
CUT2DF, 296
Deceleración en vértices, 642
en esquinas exteriores, 273
Corrección longitudinal de herramienta online, 814
Correcciones de herramientas
aditivo, 763
CORROF, 345
CORRTC, 810
CORRTRAFO, 752
COS, 454
Cota incremental, 32
COUPDEF, 954
COUPDEL, 954
COUPOF, 954
COUPOFS, 954
COUPON, 954
COUPONC, 954
COUPRES, 954
CP, 728
CPBC, 966
CPDEF, 965
CPDEL, 965
CPFMOF, 969
CPFMON, 969
CPFMSON, 968
CPFPOS + CPOF, 969
CPFPOS + CPON, 966
CPFRS, 966
CPLA, 965
CPLCTID, 966
CPLDEF, 965
CPLDEL, 965
CPLDEN, 966
CPLINSC, 971
CPLINTR, 971
CPLNUM, 966
CPLOF, 966
CPLON, 966
CPLOUTSC, 971
CPLOUTTR, 971
CPLPOS, 967
CPLSETVAL, 966
CPMALARM, 972
CPMBRAKE, 972
CPMPRT, 971
CPMRESET, 970
CPMSTART, 971
CPMVDI, 972
CPOF, 966
CPON, 966
CPRECOF, 874
CPRECON, 874
CPROT, 595
CPROTDEF, 592
CPSETTYPE, 972
CPSYNCOP, 971
CPSYNCOP2, 972
CPSYNCOV, 972
CPSYNFIP, 972
CPSYNFIP2, 972
CPSYNFIV, 972
CR, 197
Criterio de fin del movimiento programable, 643
CROTS, 331
CSPLINE, 610
CT, 205
CTAB, 936
CTABDEF, 926
CTABDEL, 932
CTABEND, 926
CTABEXISTS, 932
CTABFNO, 940
CTABFPOL, 940
CTABFSEG, 940
CTABID, 934
CTABINV, 936
CTABISLOCK, 934
CTABLOCK, 933
CTABMEMTYP, 934
CTABMPOL, 940
CTABMSEG, 940
CTABNO, 940
CTABNOMEM, 940
CTABPERIOD, 934
CTABPOL, 940
CTABPOLID, 940
CTABSEG, 940
CTABSEGID, 940

CTABSEV, 936
CTABSSV, 936
CTABTEP, 936
CTABTEV, 936
CTABTMAX, 936
CTABTMIN, 936
CTABTSP, 936
CTABTSV, 936
CTABUNLOCK, 933
CTOL, 893
CTOLG0, 187
CTRANS, 657
CUT2D, 294
CUT2DD, 294
CUT2DF, 294
CUT2DFD, 294
CUT3DC, 776
CUT3DCC, 787
CUT3DCCD, 787
CUT3DCD, 776
CUT3DF, 781
CUT3DFD, 781
CUT3DFF, 781
CUT3DFS, 781
CUTCONOF, 297
CUTCONON, 297
CUTMOD, 827
CUTMODK, 827
CYCLE4071
programación externa, 1169
CYCLE4072
programación externa, 1171
CYCLE4073
programación externa, 1175
CYCLE4074
programación externa, 1176
CYCLE4075
programación externa, 1179
CYCLE4077
programación externa, 1182
CYCLE4078
programación externa, 1186
CYCLE4079
programación externa, 1188
CYCLE435 - Definir el sistema de coordenadas del diamante
programación externa, 1127
CYCLE495, perfilar
programación externa, 1127
CYCLE60, grabado
programación externa, 1078
CYCLE61, planear
programación externa, 1081
CYCLE62, llamada de contorno
programación externa, 1083
CYCLE63, fresar caja de contorno/material sobrante caja de contorno/fresar saliente de contorno/material sobrante saliente de contorno
programación externa, 1084
CYCLE64, pretaladrar caja de contorno
programación externa, 1086
CYCLE70, fresado de roscas
programación externa, 1088
CYCLE72, fresado en contorneado
programación externa, 1090
CYCLE76, saliente rectangular
programación externa, 1094
CYCLE77, saliente circular
programación externa, 1096
CYCLE78, fresado de roscas
programación externa, 1098
CYCLE782: Adaptar a la carga
programación externa, 1129
CYCLE79, poliedro
programación externa, 1100
CYCLE800, orientar plano/orientar herramienta/ alinear herramienta
programación externa, 1132
CYCLE801, patrón de posiciones rejilla o marco
programación externa, 1136
CYCLE802, posiciones libres
programación externa, 1137
CYCLE81, puntear
programación externa, 1102
CYCLE82, taladrar
programación externa, 1103
CYCLE83, taladrado profundo 1
programación externa, 1106
CYCLE830, taladrado profundo 2
programación externa, 1140
CYCLE832, High Speed Settings
programación externa, 1146
CYCLE84, Roscado sin mandril de compensación
programación externa, 1109
CYCLE840, Roscado con macho con mandril de compensación
programación externa, 1150
CYCLE85, escariado
programación externa, 1112
CYCLE86, mandrinar
programación externa, 1113
CYCLE899, ranura abierta
programación externa, 1152

CYCLE92, tronzar
programación externa, 1115
CYCLE930, ranurado
programación externa, 1155
CYCLE940, garganta forma E y F/garganta rosca
programación externa, 1158
CYCLE95, mecanizado de contorno
programación externa, 1116
CYCLE951, desbaste
programación externa, 1160
CYCLE952, desbastar/desbastar resto/ranurar/ranurar resto/ranurar derecha-izquierda/ranurar derecha-izquierda resto
programación externa, 1163
CYCLE98, cadena de roscas
programación externa, 1118
CYCLE99, tallado de roscas
programación externa, 1122

D

D..., 81
D0, 81
DAC, 168
Datos de circunferencia
no Siemens, 1057
DC, 160
Decalaje básico, 40
Decalaje de origen
ajustable, 147
externo, 658
programable, 319
Decalaje de origen externo, 658
Decalaje del punto inicial
en el tallado de roscas, 227
Deceleración en lvértices en esquinas interiores, 642
Deceleración en vértices en todas las esquinas, 642
DEF, 407
DEFAULT, 481
DEFINE ... AS, 502
Definición de pieza en bruto, 1037
Definir el sistema de coordenadas del diamante - CYCLE435
programación externa, 1127
DELAYFSTOF, 878
DELAYFSTON, 878
DELDL, 764
DELETE, 585
DELOBJ, 741
DELTOOLENV, 838
Denominador polinómico, 626
Desbastar/desbastar resto/ranurar/ranurar resto/ranurar derecha-izquierda/ranurar derecha-izquierda resto, CYCLE952
programación externa, 1163
Desbaste
funciones de ayuda, 1044
Desbaste, CYCLE951
programación externa, 1160
Desplazamiento PTP cartesiano, 727
Desplazamiento punto a punto, 727
DIACYCOFA, 168
DIAM90, 166
DIAM90A, 168
DIAMCHAN, 168
DIAMCHANA, 168
DIAMCYCOF, 166
DIAMOF, 166
DIAMOFA, 168
DIAMON, 166
DIAMONA, 168
DIC, 168
DILF, 236
DIN 66217, 36
Dirección
Asignación del valor, 48
Direccionamiento, 571
Direcciones, 390
Direcciones OEM, 641
DISABLE, 559
DISC, 273
DISCL, 276
DISPLOF, 521
DISPLON, 521
DISPR, 882
DISR, 276
DISRP, 276
DITE, 234
DITS, 234
DIV, 454
División automática de la trayectoria, 1002
DL, 763
DM10651, 241
DO, 983
DO42465, 311
DO42900, 767
DO42910, 767
DO42920, 768
DO42930, 768
DO42935, 770
DO42940, 771
DRFOF, 348
DRIVE, 868

DRIVEA, 868
DV, 954
DYNFINISH, 871
DYNNORM, 871
DYNPOS, 871
DYNPREC, 871
DYNROUGH, 871
DYNSEMIFIN, 871

E

Easy XML, 1027
EAUTO, 610
EES, 569
EG
  Reductor electrónico, 947
EGDEF, 947
EGDEL, 953
EGOFC, 952
EGOFS, 952
EGON, 948
EGONSYN, 948
EGONSYNE, 948
Eje
  Contenedor, 386
  intercambio, 899
  tipos, 380
Eje arrastrado o esclavo
  con acoplamiento axial de valores maestros, 942
Eje de refrentado, 173
Eje geométrico
  Conmutar, 906
Eje maestro
  con acoplamiento axial de valores maestros, 942
Ejes
  adicionales, 382
  arrastrado, 923
  canal, 383
  Comando, 385
  contorneado, 383
  eje lincado guía, 387
  geométricos, 380
  lincados, 385
  máquina, 382
  PLC, 385
  posicionado, 383
  principales, 380
  síncronos, 384
Ejes adicionales, 382
Ejes arrastrados, 923
Ejes de contorneado, 383
Ejes de orientación, 696
Ejes de posicionado, 383
Ejes geométricos conmutables, 906
Ejes giratorios
  ángulo de torsión, 802
  vectores de dirección, 802
  vectores de distancia, 802
Elementos de contorno
  retirada, 1056
ELSE, 490, 983
Emisión de funciones auxiliares
  en modo de contorneado, 354
  rápida, 353
ENABLE, 559
ENAT, 610
ENDFOR, 493
ENDIF, 490
ENDLABEL, 483
ENDLOOP, 492
ENDWHILE, 494
ENS, 40
Entrada principal, 171
Escariado, CYCLE85
  programación externa, 1112
Escritura de dirección extendida, 391
Esquina de contorno
  achaflanar, 251
  redondear, 251
ESR, 1009
ESRR, 1015
ESRS, 1015
Estado de acoplamiento
  con acoplamiento axial de valores maestros, 946
  en el arrastre de ejes, 924
Estado inicial de la orientación de herramienta
ORIRESET, 686
ETAN, 610
EVERY, 983
Evoluta, 211
EXECSTRING, 452
EXECTAB, 1056
EXECUTE, 1058
EXP, 454
EXTCALL
  para SINUMERIK 840D sl, 550
EXTCLOSE, 1031
Extensión numérica, 391
EXTERN, 539
EXTOPEN, 1031

F

F...
con interpolación lineal, 191
en avance, 108
para tallado de roscas G34 G35, 233
FA, 126
Factor de
acoplamiento, 921
Factor de escala, 334
FAD, 276
FALSE, 407
FB, 140
FCTDEF, 772
FCUB, 863
FD, 131
FDA, 131
FENDNORM, 642
FFWOF, 873
FFWON, 873
FGREF, 108
FGROUP, 108
FIFOCTRL, 875
FILEDATE, 589
FILEINFO, 589
FILESIZE, 589
FILESTAT, 589
FILETIME, 589
Filos
centro, 70
Número, 82
número de herramientas de contornos, 295
posición, 70
posición relevante, 299
punto de referencia, 299
radio, 70
Fin adición aprox. - GROUP_ADDEND
programación externa, 1192
Fin bloque de programa - GROUP_END
programación externa, 1191
FINE, 954
FINEA, 643
FL, 108
FLIN, 863
FMA, 137
FNORM, 863
FOR, 493
Formato de cinta perforada, 44
FP, 368
FPO, 863
FPR, 126
FPRAOF, 126
FPRAON, 126
Frame
cancelar, 345
concatenación, 672
escala, 334
instrucciones, 316
llamar, 654
simetría especular, 337
FRAME, 407
Frame cero, 147
Frames, 314
Asignar, 655
Cadenas de frames, 656
específicos de canal, 667
globales, 666
requisitos, 668
Frames de sistema, 668
FRC, 251
FRCM, 251
Fresado de roscas, CYCLE70
programación externa, 1088
Fresado de roscas, CYCLE78
programación externa, 1098
Fresado en contorneado, CYCLE72
programación externa, 1090
Fresado frontal, 693
Fresar caja de contorno/material sobrante caja de contorno/fresar saliente de contorno/material sobrante saliente de contorno, CYCLE63
programación externa, 1084
Fresas, 72
FROM, 983
FTOCOF, 776
FTOCON, 776
Función
predefinida, 1340
Funciones M, 354
Funciones OEM, 641

G

G1, 191
G110, 177
G111, 177
G112, 177
G140, 276
G141, 276
G142, 276
G143, 276
G147, 276
G148, 276

G153
cancelar para frame, 345
con decalaje de origen, 147
G17, 149
G18, 149
G19, 149
G2, 194
G247, 276
G248, 276
G25
limitación de velocidad del cabezal, 107
limitación del campo de trabajo, 361
G26
limitación de velocidad del cabezal, 107
limitación del campo de trabajo, 361
G290, 1041
G291, 1041
G3, 194
G33, 226
G335, 240
G336, 240
G34, 233
G340, 276
G341, 276
G347, 276
G348, 276
G35, 233
G4, 377
G40, 257
G41, 257
G42, 257
G450, 273
G451, 273
G460, 288
G461, 288
G462, 288
G5, 724
G500
con decalaje de origen, 147
G505 ... G599, 147
G53
cancelar para frame, 345
con decalaje de origen, 147
G54 ... G57, 147
G58, 323
G59, 323
G60, 302
G601, 302
G602, 302
G603, 302
G62, 642
G621, 642
G64, 304
G641, 304
G642, 304
G643, 304
G644, 304
G645, 304
G7, 724
G70, 162
G700, 162
G71, 162
G710, 162
G74, 367
G75, 368
G810 ... G819, 641
G820 ... G829, 641
G9, 302
G90, 153
G91, 155
G93, 108
G94, 108
G95, 108
G96, 101
G961, 101
G962, 101
G97, 101
G971, 101
G972, 101
G973, 101
Garganta forma E y F/garganta rosca, CYCLE940
programación externa, 1158
GEOAX, 906
Geometría
ejes, 380
GET, 899
GETACTTD, 800
GETD, 899
GETDNO, 799
GETTCOR, 840
GETTENV, 839
GETVARAP, 441
GETVARDFT, 443
GETVARDIM, 442
GETVARLIM, 442
GETVARPHU, 440
GETVARTYP, 444
GFRAME0 ... GFRAME100, 350
GOTO, 478
GOTOB, 478
GOTOC, 478
GOTOF, 478
GOTOS, 477
GP, 450

Grabado, CYCLE60
programación externa, 1078
GROUP_ADDEND - Fin adición aprox.
programación externa, 1192
GROUP_BEGIN - Inicio bloque de programa
programación externa, 1191
GROUP_END - Fin bloque de programa
programación externa, 1191
Grupo G
Tecnología, 871
GUD, 408
GWPSOF, 106
GWPSON, 106

H

Herramienta
cambio con M6, 66
cambio con número T, 65
corrección del radio, 69
corrección longitudinal, 68
filo, 81
grupo, 71
memoria de corrección, 70
número de tipo, 71
orientación, 792
orientación al producirse un cambio de frame, 809
parámetros, 760
punta, 70
punto de cambio, 35
tipo, 71
Herramientas de tornear, 77
Herramientas especiales, 79
High Speed Settings, CYCLE832
programación externa, 1146
HOLES1, patrón de posiciones línea
programación externa, 1063
HOLES2, patrón de posiciones arco circular o arco
programación externa, 1064

I

I...
con interpolación circular, 194
para tallado de roscas G33, 226
para tallado de roscas G34 G35, 233
IC, 155
ID, 983
identificación
para cadena de caracteres, 53
para valores numéricos especiales, 53
para variables del sistema, 53
Identificador, 43
IDS, 983
IF, 490
IFRAME, 648
INDEX, 466
Índice de matriz, 432
INICF, 407
Inicialización
de matrices, 430
Inicio bloque de programa - GROUP_BEGIN, 1191
INIPO, 407
INIRE, 407
INIT, 496
INITIAL, 578
INITIAL_INI, 578
Instrucción, 44
INT, 407
Interpolación circular
con introducción de punto intermedio y punto final, 203
Interpolación helicoidal, 209
Interpolación de polinomios, 622
Interpolación del vector de giro, 705
Interpolación helicoidal, 209
INTERSEC, 1054
INVCCW, 211
INVCW, 211
IPOBRKA, 643
IPOENDA, 643
IPOSTOP, 954
IPTRLOCK, 880
IPTRUNLOCK, 880
IR, 240
ISAXIS, 905
ISFILE, 588
ISNUMBER, 462
ISOCALL, 547
ISVAR, 438

J

J...
con interpolación circular, 194
para tallado de roscas G34 G35, 233
JERKLIM, 890
JERKLIMA, 870
JR, 240
Juego de caracteres, 52

K

K...
con interpolación circular, 194
para tallado de roscas G33, 226
para tallado de roscas G34 G35, 233
KONT, 266
KONTC, 266
KONTT, 266
KR, 240

L

L..., 537
Label, 483
LEAD, 687
LEADOF, 942
LEADON, 942
Lenguaje de alto nivel CN, 45
LENTOAX, 859
Letras para direcciones, 1280
LF, 47
LFOF, 236
LFON, 236
LFPOS, 236
LFTXT, 236
LFWP, 236
LIFTFAST, 560
Limitación del campo de trabajo
En el sistema de coordenadas básico, 361
Límites en transformadas, 739
LIMS, 101
Lincado
eje lincado guía, 387
ejes, 385
variables, 406
LINE FEED, 47
Lineal
interpolación, 191
Llamada de contorno, CYCLE62
programación externa, 1083
Llamada de subprograma con ruta de acceso y parámetros, 548
LLI, 420
LN, 454
LONGHOLE, agujero rasgado
programación externa, 1076
LookAhead, 309
LOOP, 492
LUD, 408

M

M..., 354
M0, 354
M1, 354
M17, 525
M19
funciones M, 354
Posicionar en cabezales, 121
M2, 354
M3, 91
M30, 525
M4, 91
M40, 354
M41, 354
M42, 354
M43, 354
M44, 354
M45, 354
M5, 91
M6, 66
M70, 121
Macro, 502
Mandrinar, CYCLE86
programación externa, 1113
Máquinas
ejes, 382
origen, 34
Marca de salto
para repeticiones de secciones del programa, 483
para saltos del programa, 479
Marcha síncrona
fina, 957
somero, 957
MASLDEF, 979
MASLDEL, 979
MASLOF, 979
MASLOFS, 979
MASLON, 979
MATCH, 466
Matriz, 430
MAXVAL, 437
MCALL, 543
MD10010, 496
MD10240, 164
MD10260, 162
MD10280, 496
MD10710, 236
MD15800, 405
MD18104, 837

MD18116, 838
MD18156, 405
MD20360, 845
MD24558, 847
MD24658, 847
MD9440, 83
MEAC, 632
MEAFRAME, 662
MEAS, 630
MEASA, 632
MEAW, 630
MEAWA, 632
Mecanizado de contorno, CYCLE95
  programación externa, 1116
Memoria
  de programa, 567
  de trabajo, 578
  pretratamiento, 875
Memoria de corrección, 760
Memoria de programas
  Directorios estándar, 567
  Tipos de fichero, 568
Memoria de trabajo, 578
Memoria global de programas de pieza (GDIR), 569
MINDEX, 466
MINVAL, 437
MIRROR, 337
MKS, 36
MMC, 1026
MOD, 454
Modalmente activos, 46
MODAXVAL, 905
Modo de aceleración, 868
Modo de acoplamiento, 957
Modo de contorneado, 304
Modo de lenguaje, 1041
Movimiento de vaivén
  de inversión derecho, 992
  Penetración en el punto de inversión, 994
  Zona de inversión, 992
MPF, 568
MSG, 358
Muelas rectificadoras, 75

N

NAMETOINT, 744
NCK, 407
NEWCONF, 1022
Niveles de omisión, 50
NOC, 954
Nombre de fichero, 574
Nombre de la unidad, 572
Nombre de subprograma DIN, 575
NORM, 266
NOT, 456
Notación EES, 571
Notación NCK, 571
NPROT, 595
NPROTDEF, 592
NUMBER, 462
Número D
  Asignación libre, 798
Número del filo, 798
Números D
  Cambiar nombre, 799
  comprobar, 798
NUT, 698

O

OEMIPO1/2, 641
OFFN, 257
Offset
  longitudes de herramienta, 85
  radio de la herramienta, 85
OMA1 ... OMA5, 641
Operadores de comparación, 456
Operadores lógicos, 456
OR, 456
Orden de desplazamiento, 174
ORIAXES, 696
ORIC, 792
ORICONCCW, 698
ORICONCW, 698
ORICONIO, 698
ORICONTO, 698
ORICURVE, 701
ORID, 792
Orientación de herramienta relativa a la trayectoria, 707
Orientación de la herramienta
  relativa a la trayectoria, 707
Orientar plano/orientar herramienta/alinear herramienta, CYCLE800
  programación externa, 1132
ORIEULER, 696
Orígenes
  en el torneado, 172
ORIMKS, 694
ORIPATH, 709
ORIPATHS, 709
ORIPLANE, 698
ORIRESET(A, B, C), 685

ORIROTA, 705
ORIROTC
para giro de la orientación de herramienta, 705
para interpolación del giro de herramienta, 710
ORIROTR, 705
ORIROTT, 705
ORIRPY, 696
ORIRPY2, 696
ORIS, 792
ORISOF, 719
ORISOLH, 816
ORISON, 719
ORIVECT, 696
ORIVIRT1, 696
ORIVIRT2, 696
ORIWKS, 694
OS, 984
OSB, 984
OSC, 792
OSCILL, 989
OSCTRL, 984
OSD, 792
OSE, 984
OSNSC, 984
OSOF, 792
OSP1, 984
OSP2, 984
OSS, 792
OSSE, 792
OST, 792
OST1, 984
OST2, 984
OTOL, 893
OTOLG0, 187
OVR, 129
OVRA, 129
OVRRAP, 129

P

P..., 541
P_ACTFRAME, 670
Par de apriete
tope fijo, 375
Parada
al final del ciclo, 356
controlada por CN, 1014
ejecutada en el accionamiento, 1015
opcional, 356
programada, 356
Parada de decodificación interna, 379
Parada de decodificación previa
Nivel de señal, 379
Parada opcional, 356
Parada precisa, 302
Parada programada, 356
Parámetro
actual, 509
formal, 508
Parámetros
Herramienta, 760
transferencia en llamada de subprograma, 509, 539
Parámetros de cálculo
específicos de canal, 402
global, 404
Parámetros de ciclo de medida
CYCLE961, 1215
CYCLE971, 1232
CYCLE973, 1195
CYCLE974, 1197
CYCLE976, 1203
CYCLE977, 1211
CYCLE978, 1205
CYCLE979, 1218
CYCLE982, 1229
CYCLE994, 1200
CYCLE995, 1223
CYCLE996, 1225
CYCLE9960, 1227
CYCLE997, 1221
CYCLE998, 1208
PAROT, 342
PAROTOF, 342
Patrón de posiciones arco circular o arco, HOLES2
programación externa, 1064
Patrón de posiciones línea, HOLES1
programación externa, 1063
Patrón de posiciones rejilla o marco, CYCLE801
programación externa, 1136
PCALL, 548
PDELAYOF, 997
PDELAYON, 997
Perfilar, CYCLE495
programación externa, 1127
PFRAME, 648
PHI
para orientación a lo largo de una superficie de cono, 698
polinomios de orientación, 704
PHU, 421
Pieza
contador, 1020

contorno, 175
directorio raíz, 568
directorios, 568
origen, 34
PL
para interpolación polinómica, 622
para interpolación spline, 610
Planeado, CYCLE61
programación externa, 1081
Plano de trabajo, 33
PLC
ejes, 385
PM, 276
PO, 622
PO[PHI]
para giro de la orientación de herramienta, 709
para orientación a lo largo de una superficie de cono, 698
polinomios de orientación, 704
PO[PSI]
para giro de la orientación de herramienta, 709
para orientación a lo largo de una superficie de cono, 698
polinomios de orientación, 704
PO[THT]
para giro de la orientación de herramienta, 709
polinomios de orientación, 704
PO[XH]
para especificar la orientación de dos puntos de contacto, 701
polinomios de orientación, 704
PO[YH]
para especificar la orientación de dos puntos de contacto, 701
polinomios de orientación, 704
PO[ZH]
para especificar la orientación de dos puntos de contacto, 701
polinomios de orientación, 704
POCKET3, caja rectangular
programación externa, 1066
POCKET4, caja circular
programación externa, 1068
POLF
para retirada controlada por CN, 1010
para retirada rápida durante tallado de roscas, 236
POLFA, 1010
POLFMASK
para retirada controlada por CN, 1010
para retirada rápida durante tallado de roscas, 236
POLFMLIN
para retirada controlada por CN, 1010
para retirada rápida durante tallado de roscas, 236
Poliedro, CYCLE79
programación externa, 1100
Polo, 177
POLY, 622
POLYPATH, 622
PON, 1006
PONS, 997
Portaherramientas
cinemática, 802
orientable, 807
punto de referencia, 35
Portaherramientas orientables, 802
POS, 116
POSA, 116
POSFS, 954
Posiciones libres, CYCLE802
programación externa, 1137
Posiciones singulares, 695
POSP, 116
POT, 454
PR, 276
Precisión del contorno
programable, 874
Preparación del contorno
Señalización de error, 1058
PREPRO, 524
PRESETON, 659
PRESETONS, 661
Pretaladrar caja de contorno, CYCLE64
programación externa, 1086
Pretratamiento
memoria, 875
PRIO, 557
PRLOC, 407
Procedimiento
predefinido, 1316
Process DataShare, 1031
Profundidad de imbricado
de estructuras de control, 489
Profundidad de penetración, 779
Programa
bifurcación, 481
cabecera, 53
de inicialización, 579
direccionamiento, 571
fin, 46
memoria, 568
Nombre, 43

repetición, 541
saltos, 478
tiempos de ejecución, 1017
Programa CN
crear, 51
Programa de inicialización, 579
Programación CN
Juego de caracteres, 52
Programación de círculos
con ángulo en el vértice y centro, 199
con coordenadas polares, 201
con introducción del centro y el punto final, 195
con introducción del radio y el punto final, 197
tipos de interpolación, 194
Programación de la orientación de la herramienta, 696
Programación de la sucesión de elementos de contorno, 216
programación externa, 1191
Programación indirecta
códigos G, 448
de atributos de posición, 450
de líneas de programa de pieza, 452
direcciones, 446
Programación por diámetro, 166
Programación por radio, 166
PROTA, 747
PROTD, 749
PROTS, 748
PSI
para orientación a lo largo de una superficie de cono, 698
polinomios de orientación, 704
PTP, 728
PTPG0, 728
PTPWOC, 728
PUD, 408
PUNCHACC, 997
Punta de la fresa, 780
Puntear, CYCLE81
programación externa, 1102
Puntero de interrupción automático, 882
Punto auxiliar de la fresa, 780
Punto de mecanizado de la fresa, 780
Punto de referencia, 35
Punto de tope, 34
Punto fijo
aproximación, 368
Punto final, 174
Punto inicial, 35
Punto inicial - punto final, 174
Punto inicial/ángulo inicial, 269
Puntos de referencia, 34
Punzonado, 997
PUTFTOC, 775
PUTFTOCF, 774
PW, 610

Q

QU, 353

R

RAC, 168
Radio
efectivo, 114
Radio de referencia, 114
Radio polar, 29
Ranura abierta, CYCLE899
programación externa, 1152
Ranura circular, SLOT2
programación externa, 1074
Ranura longitudinal, SLOT1
programación externa, 1071
Ranurado, CYCLE930
programación externa, 1155
READ, 586
REAL, 407
Rectificado oblicuo de ranuras, 724
REDEF, 413
redondear, 460
Redondeo, 251
Reductor electrónico, 947
Registrar y buscar áreas no aptas para la búsqueda, 881
Regla de los tres dedos, 36
RELEASE, 899
REP, 430
REPEAT, 483
REPEATB, 483
Repetición de partes del programa
con programación indirecta CALL, 546
REPOSA, 882
REPOSH, 882
REPOSHA, 882
REPOSL, 882
REPOSQ, 882
REPOSQA, 882
RET, 526
RET (parametrizable), 527
RETB (parametrizable), 533

Retirada
controlada por CN, 1010
dirección en el tallado de roscas, 237
ejecutada en el accionamiento, 1015
Retirada rápida
Tallado de roscas, 236
Retirada rápida del contorno, 560
RG, 404
RIC, 168
RINDEX, 466
RMBBL, 882
RMEBL, 882
RMIBL, 882
RMNBL, 882
RND, 251
RNDM, 251
Rosca
cadena de, 227
de varios hilos, 227
paso, 233
sentido de giro, 228
tallado G33, 226
tallado G34 G35, 233
Rosca a derechas, 228
Rosca a izquierdas, 228
Rosca cilíndrica, 231
Rosca cónica, 232
Rosca transversal, 231
Roscado con macho con mandril de compensación, CYCLE840
programación externa, 1150
Roscado con macho sin mandril de compensación, CYCLE84
programación externa, 1109
Roscas convexas, 240
ROT, 325
Rotación
del vector de orientación, 705
programable, 325
ROTS, 331
ROUND, 454
ROUNDUP, 460
RP, 179
RPL, 325
Run MyScreens, 1027
Ruta de acceso, 572
Ruta de búsqueda
en llamada a subprograma, 576
Ruta de búsqueda programable, 549
Ruta de directorio, 573
Rutina de interrupción
Del, 559
Desconexión/conexión, 559
Movimiento de retroceso, 562
Reasignar, 558
Retirada rápida del contorno, 560
sentido de desplazamiento programable, 563

S

S, 91
Salida
a equipo/fichero externo, 1031
Salida de funciones auxiliares
Propiedades de las funciones auxiliares, 351
Saliente circular, CYCLE77
programación externa, 1096
Saliente rectangular, CYCLE76
programación externa, 1094
Salto
a marcas (etiquetas) de salto, 478
al inicio del programa, 477
SAVE, 515
SBLOF, 516
SBLON, 516
SCALE, 334
SCC, 101
SCPARA, 914
SD, 610
SD41610, 759
SD41611, 759
SD42010, 235
SD42440, 156
SD42442, 156
SD42466, 311
SD42475, 714, 717
SD42476, 714, 717
SD42477, 715, 717
SD42984, 829
SD43240, 122
SD43250, 122
Sección de programa
repetición, 483
Secuencia, 44
fin, 47
final LF, 53
longitud, 47
número, 47
Ocultar, 49
Orden de las instrucciones, 47
Secuencia a secuencia
supresión, 516
Secuencia de parada, 881
Secuencialmente activos, 46

Secuencias opcionales, 49
SET, 430
SETAL, 1035
SETDNO, 799
SETINT, 557
SETM, 496
SETMS, 91
SETTCOR, 847
SF, 226
Sierra para ranurar, 80
Simulación de valores maestros, 946
SIN, 454
Sincronismo de la velocidad, 954
Sincronismo de posición, 954
Sincronismo de posición con decalaje angular adicional, 954
Síncronos
  ejes, 384
Sistema de coordenadas
  Básico, 38
  vista general, 36
Sistema de coordenadas básico, 38
Sistema de coordenadas de máquina, 36
Sistema de coordenadas de pieza, 41
Sistema de medida, 162
Sistema de origen
  ajustable, 40
  Básico, 39
Sistema de origen básico, 39
SLOT1, ranura longitudinal
  programación externa, 1071
SLOT2, ranura circular
  programación externa, 1074
SOFT, 868
SOFTA, 868
SON, 997
SONS, 997
SPCOF, 120
SPCON, 120
SPF, 568
SPI, 905
SPIF1, 997
SPIF2, 997
Spline
  interpolación, 610
  tipos, 616
Spline A, 616
Spline B, 617
Spline C, 618
SPLINEPATH, 620
SPN, 1002
SPOF, 997
SPOS, 121
SPOSA, 121
SPP, 1002
SPRINT, 469
SQRT, 454
SR, 137
SRA, 137
ST, 137
STA, 137
START, 496
STARTFIFO, 875
STAT, 729
STOLF, 187
STOPFIFO, 875
STOPRE, 875
String
  concatenación, 463
  formatear, 469
  longitud, 465
  operaciones, 461
STRING, 407
STRINGIS, 1023
STRLEN, 465
Subprograma
  aplicación, 505
  llamada con transferencia de parámetros, 539
  llamada sin transferencia de parámetros, 537
  llamada, indirecta, 545
  llamada, modal, 543
  Nombre, 506
  repetición, 541
  Ruta de búsqueda programable, 549
  salto atrás parametrizable (RET...), 527
  salto atrás parametrizable (RETB...), 533
SUBSTR, 467
SUG, 106
SUPA
  cancelar para frame, 345
  con decalaje de origen, 147
SVC, 94
SYNR, 407
SYNRW, 407
SYNW, 407

T

T0, 65
Taladrado profundo 1, CYCLE83
  programación externa, 1106
Taladrado profundo 2 CYCLE830
  programación externa, 1140

Taladrar, CYCLE82
programación externa, 1103
Tallado de roscas, CYCLE99
programación externa, 1122
TAN, 454
TANG, 973
TANGDEL, 978
TANGOF, 977
TANGON, 976
TCARR, 807
TCOABS, 807
TCOFR, 807
TCOFRX, 807
TCOFRY, 807
TCOFRZ, 807
THETA
para giro de la orientación de herramienta, 705
para interpolación del giro de herramienta, 710
Tiempo de ejecución
comportamiento de estructuras de control, 489
Tiempo de espera, 377
Tiempo de mecanizado, 1017
Tiempo residual
de una pieza, 1019
TILT, 687
Tipo de cinemática, 806
Tipos de transformada
Función general, 673
Tirón
corrección, 890
limitación, 868
TLIFT, 975
TMOF, 1008
TMON, 1008
TOFF, 85
TOFFL, 85
TOFFLR, 85
TOFFOF, 814
TOFFON, 814
TOFFR, 85
TOFRAME, 342
TOFRAMEX, 342
TOFRAMEY, 342
TOFRAMEZ, 342
Tolerancias G0, 187
TOLOWER, 464
TOOLENV, 835
Tope fijo, 372
TOROT, 342
TOROTOF, 342
TOROTX, 342
TOROTY, 342
TOROTZ, 342
TOUPPER, 464
TOWBCS, 769
TOWKCS, 769
TOWMCS, 769
TOWSTD, 769
TOWTCS, 769
TOWWCS, 769
TRAANG
con ángulo programable, 723
TRACON, 726
TRACYL, 720
TRAFOOF, 740
TRAFOON, 751
TRAILOF, 921
TRAILON, 921
TRANS, 319
Transformaciones de coordenadas (frames), 41
Transformada con eje lineal basculable, 683
Transformada de ángulo inclinado (TRAANG)
con ángulo programable, 723
Transformada de envolvente de cilindro, 677
Transformada de orientación TRAORI
Cinemática de máquina, 676
Movimientos de orientación, 675
Programación de la orientación, 685
Transformada genérica de 5/6 ejes, 676
Variantes de la programación de la orientación, 685
Transformada polar, 677
Transformadas
concatenadas, 726
Estado inicial de la orientación de herramienta independiente de la cinemática, 674
Transformada de orientación, 673
Transformadas cinemáticas, 674
Transformadas concatenadas, 675
transformadas de tres, cuatro y cinco ejes, 684
TRANSMIT, 720
TRAORI, 684
Tronzar, CYCLE92
programación externa, 1115
Troquelado, 997
TRUE, 407
TRUNC, 454
TU, 733
TURN, 209

U

ULI, 420
UNTIL, 495

V

Vaivén
asíncrono, 984
controlar a través de acciones síncronas, 989
Penetración parcial, 992
Vaivén asíncrono, 984
Vaivén síncrono, 989
Vaivén asíncrono, 984
Vaivén síncrono
Acciones síncronas, 993
Asignación del eje de vaivén y del eje de penetración, 992
Definir penetraciones, 992
evaluación ciclo IPO, 995
Penetración a profundidad de trabajo, 993
Penetración en la zona de inversión, 993
Siguiente penetración parcial, 995
Valor de ajuste, 764
Valor de desgaste, 764
Valor S
interpretación, 93
Variable
Conversión de tipos, 445
Variables
Conversión de tipos, 461
definidas por el usuario, 407
Variables frame
Asignar valores, 651
Llamada a transformadas de coordenadas, 646
Variable frame predefinida, 648
Vector de dirección, 690
Vector de orientación THETA, 705
Velocidad de corte, 94
Velocidad de corte (constante), 101
Velocidad de giro de la herramienta
máximo, 96
VELOLIM, 891
VELOLIMA, 870
Vigilancia de colisión, 292
Visualización de secuencia
suprimir, 521
Volante
corrección, 131

W

WAB, 276
WAITC, 954
WAITE, 496
WAITENC, 913
WAITM, 496
WAITMC, 496
WAITP, 116
WAITS, 121
WALCS<n>, 364
WALCS0, 364
WALIMOF, 361
WALIMON, 361
WHEN, 983
WHEN-DO, 993
WHENEVER, 983
WHENEVER-DO, 993
WHILE, 494
WKS, 41
orientar según la pieza, 342
WORKPIECE, 1037
WRITE, 582
WRTPR, 359

X

X..., 176
XML de usuario, 1027
XOR, 456

Y

Y..., 176

Z

Z..., 176
Zonas protegidas, 592