

SINUMERIK

SINUMERIK 840D sl / 828D Arbetsförberedelse

Programmeringshandbok

Gäller för

Styrning
SINUMERIK 840D sl / 840DE sl / 828D

CNC-Software version 4.8 SP3

08/2018
6FC5398-2BP40-6FA2

Förord	
Grundläggande säkerhetsanvisningar	1
Flexibel NC-programmering	2
Fil- och programförvaltning	3
Skyddsområden	4
Speciella vägkommandon	5
Koordinattransformationer (frames)	6
Transformationer	7
Kinematiska följder	8
Undvikande av kollision med kinematiska följder	9
Transformationer med kinematiska följder	10
Verktygskompenseringar	11
Banbeteende	12
Axelkopplingar	13
Synkronaktioner	14
Pendling	15
Stansning och nibbling	16
Slipning	17
Ytterligare funktioner	18
Egna avspåningsprogram	19
Programmera cykler externt	20
Tabeller	21
Bilaga	A

Juridisk information

Varningskoncept

Denna handbok innehåller anvisningar, som du måste iakttaga för din personliga säkerhet och för att undvika materielskador. Anvisningarna för din personliga säkerhet framhävs av en varningstriangel, anvisningar för enbart materielskador står utan varningstriangel. Allt efter farlighetsgrad skildras varningsanvisningarna i avtagande ordningsföljd i följande beskrivning.

FARA

betyder att dödsfall eller svåra personskador **kommer att** inträda, om inte lämpliga försiktighetsåtgärder vidtages.

VARNING

betyder att dödsfall eller svåra personskador **kan** inträda, om inte lämpliga försiktighetsåtgärder vidtages.

SE UPP

betyder att lätta personskador kan inträda, om inte lämpliga försiktighetsåtgärder vidtages.

OBSERVERA

betyder att materielskador kan inträda, om inte lämpliga försiktighetsåtgärder vidtages.

Vid uppträdande av flera farlighetsgrader används alltid varningsanvisningen för den högsta graden. När det i en varningsanvisning varnas med en varningstriangel för personskador, då kan i samma varningsanvisning dessutom finnas en varning för materielskador bifogad.

Kvalificerad personal

Produkten eller systemet som tillhör denna dokumentation får endast hanteras av **kvalificerad personal** för vardera arbetsområde under beaktande av de för arbetsområdet gällande dokumentationerna, speciellt i dessa förekommande säkerhets- och varningsanvisningar. Kvalificerad personal kan på grund av sin utbildning och erfarenhet identifiera risker vid hanteringen av produkten/systemet och undvika möjliga faror.

Avsedd användning av produkter från Siemens

Var vänlig och iakttag följande:

VARNING

Siemensprodukter får endast användas för de ändamål som anges i katalogen och i den tillhörande tekniska dokumentationen. Om främmande produkter och komponenter används måste dessa vara rekommenderade resp. godkända av Siemens. Felfri och säker produktfunktion förutsätter korrekt transport samt korrekt förvaring, uppställning, montering, installering, driftstart, manövrering och underhåll. Föreskrivna omgivningsvillkor måste följas. Anvisningar i den tillhörande dokumentationen måste beaktas.

Märken

Alla med skyddsmärket ® markerade beteckningar är av Siemens AG registrerade varumärken. De övriga beteckningarna i detta dokument kan vara märken, vars användning av tredje man för eget ändamål kan skada innehavarens rättigheter.

Ansvarsbefrielse

Vi har kontrollerat innehållet i den tryckta skriften med avseende på överensstämmelse med den beskrivna hård- och mjukvaran. Trots detta kan avvikelser inte uteslutas så att vi inte kan garantera en fullständig överensstämmelse. Uppgifterna i denna skrift kontrolleras regelbundet, nödvändiga ändringar ingår i de följande upplagorna.

Förord

SINUMERIK- dokumentation

SINUMERIK-dokumentationen är uppdelad i följande kategorier:

- Allmän dokumentation/kataloger
- Användardokumentation
- Tillverkar- /Servicedokumentation

Ytterligare informationer

Under följande adress (<https://support.industry.siemens.com/cs/de/en/view/108464614>) finns informationer till följande teman:

- Beställa dokumentation/översikt tryckta skrifter
- Länkar som leder vidare för nerladdning av dokument
- Använda dokumentation online (hitta och söka igenom handböcker/informationer)

Vid frågor till den tekniska dokumentationen (t.ex. förslag, korrigeringar) sänder du e-post till följande adress (<mailto:docu.motioncontrol@siemens.com>).

mySupport/Dokumentation

Under följande Adress (<https://support.industry.siemens.com/My/ww/en/documentation>) finner du informationer, hur du individuellt sammanställer din dokumentation med Siemensinnehåll som bas och anpassar den till den egna maskindokumentationen.

Training

Under följande adress (<http://www.siemens.com/sitrain>) hittar du informationer om SITRAIN – Siemens utbildning för produkter, system och lösningar för servo- och automationsteknik.

FAQs

Frequently Asked Questions finns på Service&Support-sidorna under Produkt Support (<https://support.industry.siemens.com/cs/de/en/ps/faq>).

SINUMERIK

Informationer till SINUMERIK finns under följande adress (<http://www.siemens.com/sinumerik>).

Adressat

Den föreliggande skriften vänder sig till:

- Programmerare
- Projekterare

Användning

Programmeringshandboken kvalificerar målgruppen att projektera, att skriva, att testa och att åtgärda fel i program och softwareytor.

Standardomfattning

I den föreliggande programmeringshandboken beskrivs funktionaliteten för standardomfattningen. Kompletteringar eller ändringar som görs av maskintillverkaren, dokumenteras av maskintillverkaren.

I styrningen kan finnas ytterligare i denna dokumentation ej förklarade funktioner som kan köras. Det består dock inget anspråk på dessa funktioner vid ny leverans eller vid service.

Likaså innehåller denna dokumentation av överskådlighetsskäl inte samtliga detaljinformationer över alla typer av produkter och kan inte heller ta hänsyn till alla tänkbara fall av uppställning, drift och underhåll.

Teknisk support

Landesspecifika telefonnummer för teknisk rådgivning finns i internet under följande adress (<https://support.industry.siemens.com/sc/ww/en/sc/2090>) i området "Kontakt".

Informationer över struktur och innehåll

Programmeringshandbok Grunder / Arbetsförberedelse

Beskrivningarna för NC-programmeringen är uppdelad på två handböcker:

1. Grunder

Programmeringshandboken "Grunder" är till för yrkesmaskinister och förutsätter motsvarande kunskaper i borr-, fräs- och svarvbearbetning. I enkla programmeringsexempel förklaras de också enligt DIN 66025 kända kommandona och anvisningarna.

2. Arbetsförberedelse

Programmeringshandboken "Arbetsförberedelse" är till för teknologer med kännedom om alla programmeringsmöjligheterna. SINUMERIK-styrningen möjliggör med ett speciellt programmeringsspråk programmeringen av ett komplext arbetsstycksprogram (t.ex. ytor med fri form, kanalkoordinering, ...) och underlättar en krävande programmering för teknologerna.

Disponibilitet av de beskrivna NC-språkelementen

Alla i den föreliggande handboken beskrivna NC-språkelementen står för SINUMERIK 840D sl till förfogande. Disponibiliteten beträffande SINUMERIK 828D framgår av tabellen "Anvisningar: Disponibilitet vid SINUMERIK 828D (Sida 854)".

Innehållsförteckning

	Förord.....	3
1	Grundläggande säkerhetsanvisningar.....	17
1.1	Allmänna säkerhetsanvisningar.....	17
1.2	Garanti och ansvar för applikationsexempel.....	18
1.3	Industrial Security.....	19
2	Flexibel NC-programmering.....	21
2.1	Variabler.....	21
2.1.1	Systemdata.....	21
2.1.2	Fördefinierade användarvariabler: Räkneparameter.....	23
2.1.2.1	Kanalspecifika räkneparametrar (R).....	23
2.1.2.2	Globala räkneparametrar (RG).....	25
2.1.3	Fördefinierade användarvariabler: Link-variabler.....	26
2.1.4	Definition av användarvariabler (DEF).....	28
2.1.5	Redefinition av systemdata, användardata och NC-språkkommandon (REDEF).....	34
2.1.6	Attribut: Initieringsvärde.....	37
2.1.7	Attribut: Gränsvärden (LLI, ULI).....	40
2.1.8	Attribut: Fysikalisk enhet (PHU).....	41
2.1.9	Attribut: Åtkomsträtter (APR, APW, APRP, APWP, APRB, APWB).....	44
2.1.10	Attribut: Dataklass (DCM, DCI, DCU) - endast SINUMERIK 828D.....	49
2.1.11	Översikt över definierbara och redefinierbara attribut.....	50
2.1.12	Definition och initiering av fältvariabler (DEF, SET, REP).....	51
2.1.13	Definition och initiering av fältvariabler (DEF, SET, REP): Ytterligare informationer.....	55
2.1.14	Datatyper.....	56
2.1.15	Kontrollera om en variabel finns (ISVAR).....	57
2.1.16	Läsa attributvärden/datatyp (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP).....	59
2.2	Indirekt programmering.....	65
2.2.1	Indirekt programmering av adresser.....	65
2.2.2	Indirekt programmering av G-kommandon.....	67
2.2.3	Indirekt programmering av positionsattribut (GP).....	68
2.2.4	Indirekt programmering av detaljprogramrader (EXECSTRING).....	71
2.3	Räknefunktioner.....	72
2.4	Jämförelse- och logiska operationer.....	75
2.5	Noggrannhetskorrigerig vid jämförelsefel (TRUNC).....	77
2.6	Minimum, maximum och område med variabler (MINVAL, MAXVAL, BOUND).....	79
2.7	Prioritet för operationerna.....	81
2.8	Möjliga typkonverteringar.....	82
2.9	Stringoperationer.....	83
2.9.1	Typkonvertering enligt STRING (AXSTRING)	83

2.9.2	Typkonvertering av STRING (NUMBER, ISNUMBER, AXNAME)	84
2.9.3	Länkning av strings (<<)	85
2.9.4	Omvandling i små/stora bokstäver (TOLOWER, TOUPPER)	86
2.9.5	Bestämma längd för en string (STRLEN)	87
2.9.6	Söka tecken/string i string (INDEX, RINDEX, MINDEX, MATCH)	88
2.9.7	Val av en delstring (SUBSTR)	89
2.9.8	Läsning och skrivning av enskilda tecken.....	89
2.9.9	Formatera string (SPRINT).....	91
2.10	Programhopp och -förgreningar.....	100
2.10.1	Återhopp till programbörjan (GOTOS).....	100
2.10.2	Programhopp till hoppmärken (GOTOB, GOTO, GOTO, GOTOC).....	101
2.10.3	Programförgrening (CASE ... OF ... DEFAULT ...)	104
2.11	Upprepning av programdel (REPEAT, REPEATB, ENDLABEL, P).....	106
2.12	Kontrollstrukturer.....	112
2.12.1	Villkorlig anvisning och förgrening (IF, ELSE, ENDIF).....	113
2.12.2	Ändlös programslinga (LOOP, ENDLOOP).....	115
2.12.3	Räkneslinga (FOR ... TO ..., ENDFOR).....	115
2.12.4	Programslinga med villkor vid slingans början (WHILE, ENDWHILE).....	117
2.12.5	Programslinga med villkor vid slingans slut (REPEAT, UNTIL).....	117
2.12.6	Programexempel med kapslade kontrollstrukturer.....	118
2.13	Koordineringskommandon (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)....	119
2.14	Interruptrutin (ASUP).....	124
2.14.1	Funktion hos en interruptrutin.....	124
2.14.2	Upprätta interruptrutin.....	125
2.14.3	Tillordna och starta interruptrutin (SETINT, PRIO, BLSYNC).....	126
2.14.4	Inaktivera/aktivera tillordning för en interruptrutin (DISABLE, ENABLE).....	127
2.14.5	Radera tillordning av en interruptrutin (CLRINT).....	128
2.14.6	Snabblyftning från konturen (SETINT LIFTFAST, ALF).....	129
2.14.7	Förflytningsriktning vid snabblyftning från konturen	131
2.14.8	Rörelseförlopp vid interruptrutiner.....	134
2.15	Axelbyte, spindelbyte (RELEASE, GET, GETD).....	135
2.16	Överlämna axel till en annan kanal (AXTOCHAN).....	140
2.17	Ställa in maskindata verksamt (NEWCONF).....	142
2.18	Skriva fil (WRITE).....	143
2.19	Radera fil (DELETE).....	147
2.20	Läsa rader i fil (READ).....	148
2.21	Kontrollera om en fil finns (ISFILE).....	150
2.22	Läsa ut fil-informationer (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO).....	151
2.23	Avrundning uppåt (ROUNDUP).....	153
2.24	Underprogramteknik.....	154
2.24.1	Allmänt.....	154
2.24.1.1	Underprogram.....	154
2.24.1.2	Underprogramnamn.....	155
2.24.1.3	Förgrening av underprogram.....	156
2.24.1.4	Sökväg.....	157

2.24.1.5	Formal- och aktualparametrar.....	157
2.24.1.6	Parameteröverföring.....	158
2.24.2	Definition av ett underprogram.....	160
2.24.2.1	Underprogram utan parameteröverföring.....	160
2.24.2.2	Underprogram med parameteröverföring Call-by-Value (PROC).....	160
2.24.2.3	Underprogram med parameteröverföring Call-by-Reference (PROC, VAR).....	162
2.24.2.4	Säkra modala G-funktioner (SAVE).....	164
2.24.2.5	Undertrycka blockvis bearbetning (SBLOF, SBLON).....	165
2.24.2.6	Undertrycka aktuell blockindikering (DISPLOF, DISPLON, ACTBLOCNO).....	171
2.24.2.7	Känneteckna underprogram med förberedelse (PREPRO).....	174
2.24.2.8	Underprogramåterhopp M17.....	174
2.24.2.9	Underprogramåterhopp RET.....	175
2.24.2.10	Parametrerbart underprogramåterhopp (RET ...).....	176
2.24.2.11	Parametrerbart underprogramåterhopp (RETB ...).....	182
2.24.3	Anrop av ett underprogram.....	186
2.24.3.1	Underprogramanrop utan parameteröverföring.....	186
2.24.3.2	Underprogramanrop med parameteröverföring (EXTERN).....	188
2.24.3.3	Antal programupprepningar (P).....	190
2.24.3.4	Modalt underprogramanrop (MCALL).....	191
2.24.3.5	Indirekt underprogramanrop (CALL).....	193
2.24.3.6	Indirekt underprogramanrop med uppgift över den programdel som ska utföras (CALL BLOCK ... TO ...).....	194
2.24.3.7	Indirekt anrop av ett i ISO-språk programmerat program (ISOCALL).....	195
2.24.3.8	Anropa underprogram med sökvägsuppgift och parametrar (PCALL).....	196
2.24.3.9	Utvidga sökvägen vid underprogramanrop (CALLPATH).....	197
2.24.3.10	Genomarbета externt underprogram (840D sl) (EXTCALL).....	198
2.24.3.11	Genomarbета externt underprogram (828D) (EXTCALL).....	201
2.25	Makroteknik (DEFINE ... AS).....	206
3	Fil- och programförvaltning.....	209
3.1	Programminne.....	209
3.1.1	Programminne i NC.....	209
3.1.2	Extern programminne.....	211
3.1.3	Adressering av filer i programminnet.....	213
3.1.4	Sökväg vid underprogramanrop.....	217
3.1.5	Avfrågning av sökväg och filnamn.....	218
3.2	Arbetsminne (CHANDATA, COMPLETE, INITIAL).....	220
4	Skyddsområden.....	223
4.1	Definiera skyddsområden (CPROTDEF, NPROTDEF).....	223
4.2	Aktivera/inaktivera skyddsområden (CPROT, NPROT).....	227
4.3	Kontroll av kränkning av skyddsområden, arbetsfältsbegränsning och mjukvarugränsbrytare (CALCPOSI).....	231
5	Speciella vägkommandon.....	241
5.1	Uppsöka kodade positioner (CAC, CIC, CDC, CACP, CACN).....	241
5.2	Spline-interpolering (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL).....	242
5.3	Spline-förband (SPLINEPATH).....	253

5.4	Koppla till/från NC-block-kompression (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF).....	255
5.5	Polynom-interpolering (POLY, POLYPATH, PO, PL).....	256
5.6	Inställbar banreferens (SPATH, UPATH).....	262
5.7	Mäta med kopplande mätprobe (MEAS, MEAW).....	264
5.8	Axelspecifik mätning (MEASA, MEAWA, MEAC) (option).....	267
5.9	Speciella funktioner för OEM-användaren (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829).....	277
5.10	Matningsreducering med hörnfördröjning (FENDNORM, G62, G621)	278
5.11	Programmerbart rörelseslutkriterium (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA).....	279
6	Koordinattransformationer (frames).....	283
6.1	Koordinattransformation via framevariabel.....	283
6.1.1	Fördefinierad framevariabel (\$P_CHBFRAME, \$P_IFRAME, \$P_PFRAME, \$P_ACTFRAME).....	285
6.2	Värdetillordningar till frames.....	289
6.2.1	Tillordna direkta värden (axelvärde, vinkel, skala).....	289
6.2.2	Läsa och förändra framekomponenter (TR, FI, RT, SC, MI).....	291
6.2.3	Räkna med frames.....	292
6.2.4	Definition av framevariabler (DEF FRAME).....	294
6.3	Grov- och finförflyttning (CTRANS, CFINE).....	295
6.4	Extern nollpunktsförflyttning (\$AA_ETRANS).....	297
6.5	Ärvärdesinställning med förlust av referensstatus (PRESETON).....	299
6.6	Ärvärdesinställning utan förlust av referensstatus (PRESETON).....	301
6.7	Frame-beräkning ur 3 mätpunkter i rymden (MEAFRAME).....	303
6.8	NCU-globala frames.....	307
6.8.1	Kanalspecifika frames (\$P_CHBFR, \$P_UBFR).....	308
6.8.2	I kanalen verksamma frames.....	308
7	Transformationer.....	313
7.1	Allmän programmering av transformationstyperna.....	313
7.1.1	Orienteringsrörelser vid transformationerna.....	315
7.1.2	Översikt av orienteringstransformationen TRAORI.....	319
7.2	Tre-, fyr- och fem-axel-transformation (TRAORI).....	321
7.2.1	Allmänna sammanhang kardanskt verktygshuvud.....	321
7.2.2	Tre-, fyr- och fem-axel-transformation (TRAORI).....	324
7.2.3	Varianter för orienteringsprogrammering och grundläge (ORIRESET).....	325
7.2.4	Programmering av verktygsorienteringen (A..., B..., C..., LEAD, TILT).....	327
7.2.5	Ändplansfräsning (A4, B4, C4, A5, B5, C5).....	333
7.2.6	Referens för orienteringsaxlarna (ORIWKS, ORIMKS).....	334
7.2.7	Programmering av orienteringsaxlarna (ORIAxes, ORIVect, ORIEuler, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2).....	336

7.2.8	Orienteringsprogrammering längs en konisk mantelyta (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO).....	338
7.2.9	Orienteringsuppgift med två kontaktpunkter (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=)....	341
7.3	Orienteringspolynom (PO[Winkel], PO[Koordinate]).....	343
7.4	Vridningar av verktygsorienteringen (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA).....	345
7.5	Banrelativa orienteringar.....	348
7.5.1	Orienteringstyper relativt till banan.....	348
7.5.2	Banrelativ vridning av verktygsorienteringen (ORIPATH, ORIPATHS, vridvinkel).....	349
7.5.3	Banrelativ interpolering för verktygsvridningen (ORIROTC, THETA).....	350
7.5.4	Jämning av orienteringsförloppet (ORIPATHS A8=, B8=, C8=).....	352
7.6	Komprimering av orienteringen (COMPON, COMPCURV, COMPCAD, COMPSURF).	354
7.7	Koppla till/från jämning av orienteringsförloppet (ORISON, ORISOF).....	357
7.8	Kinematisk transformation.....	359
7.8.1	Koppla till frontsidtransformation (TRANSMIT).....	359
7.8.2	Koppla till cylindermanteltransformation (TRACYL).....	359
7.8.3	Koppla till snedvinkeltransformation med programmerbar vinkel (TRAANG).....	362
7.8.4	Sned instickning på slipmaskiner (G5, G7).....	363
7.9	Koppla till sammanlänkad transformation (TRACON).....	365
7.10	Kartesisk PTP-körning.....	367
7.10.1	Koppla till/från kartesisk PTP-körning (PTP, PTPG0, PTPWOC, CP).....	367
7.10.2	Ge länkarnas position (STAT).....	368
7.10.3	Ange förtecken för axelvinkel (TU).....	372
7.10.4	Exempel 1: PTP-körning av en 6-axelrobot med ROBX-transformation.....	375
7.10.5	Exempel 2: PTP-körning vid generisk 5-axel transformation.....	376
7.10.6	Exempel 3: PTPG0 och TRANSMIT.....	376
7.11	Randvillkor vid valet av en transformation.....	378
7.12	Välja bort transformation (TRAFOOF).....	379
8	Kinematiska följder.....	381
8.1	Radering av komponenter (DELOBJ).....	381
8.2	Indexbestämning med namn (NAMETOINT).....	384
9	Undvikande av kollision med kinematiska följder.....	385
9.1	Kontroll av kollisionsspar (COLLPAIR).....	386
9.2	Begära nyberäkning av maskinmodellen för undvikande av kollision (PROTA).....	387
9.3	Sätta skyddsområdestillstånd (PROTS).....	388
9.4	Avståndsbestämning för två skyddsområden (PROTD).....	389
10	Transformationer med kinematiska följder.....	391
10.1	Koppla till transformation (TRAFOON).....	391
10.2	Modifiera orienteringstransformation enligt uppmätning av maskinen (CORRTRAFO).....	392
11	Verktygskompenseringar.....	401
11.1	Kompenseringsminne.....	401

11.2	Additiva kompenseringar.....	404
11.2.1	Välja additiva kompenseringar (DL).....	404
11.2.2	Fastlägga förslitnings- och inställningsvärden (\$TC_SCPxy[t,d], \$TC_ECPxy[t,d]).....	405
11.2.3	Radera additiva kompenseringar (DELDL).....	406
11.3	Verktygskompensering-specialbehandling.....	408
11.3.1	Spegla verktygslängder.....	409
11.3.2	Förteckenvärdering förslitning.....	410
11.3.3	Koordinatsystem i den aktiva bearbetningen (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS).....	411
11.3.4	Verktygslängd och planbyte.....	413
11.4	Online-verktygskompensering.....	415
11.4.1	Definiera polynomfunktion (FCTDEF).....	415
11.4.2	Skriva online-verktygskompensering, kontinuerligt (PUTFTOCF).....	416
11.4.3	Skriva online-verktygskompensering, diskret (PUTFTOC).....	417
11.4.4	Koppla till/från online-verktygskompenseringen (FTOCON/FTOCOF).....	418
11.5	3D-verktygssradiekompensering.....	419
11.5.1	Välja 3D-verktygssradiekompensering för 3D-periferifräsningen (CUT3DC, CUT3DCD, ISD).....	419
11.5.2	Välja 3D-verktygssradiekompensering för 3D-ändplansfräsning (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD).....	423
11.5.3	3D-periferifräsning under hänsynstagande till en begränsningsyta (CUT3DCC, CUT3DCCD).....	428
11.6	Verktygssorientering (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST).....	433
11.7	Fri D-nummerfördelning, skärnummer.....	439
11.7.1	Fri D-nummerfördelning, skärnummer (adress CE).....	439
11.7.2	Fri D-nummerfördelning: Kontrollera D-nummer (CHKDNO).....	439
11.7.3	Fri D-nummerfördelning: Byta namn på D-numren (GETDNO, SETDNO).....	440
11.7.4	Fri D-nummerfördelning: Fastställa T-nummer till föreskrivet D-nummer (GETACTTD).	441
11.7.5	Fri D-nummerfördelning: Sätta D-nummer ogiltiga (DZERO).....	441
11.8	Verktygssbärarkinematik.....	442
11.9	Verktygslängdkompenseringar för orienterbara verktygsbärare (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ).....	447
11.10	Online-verktygslängdkompensering (TOFFON, TOFFOF).....	450
11.11	Modifikation av kompenseringsdata för vridbara verktyg.....	453
11.11.1	Beräkna orienteringar (ORISOLH).....	453
11.11.2	Aktivera modifikation av kompenseringsdata för vridbara verktyg (CUTMOD, CUTMODK).....	462
11.12	Arbeta med verktygsomgivningar.....	469
11.12.1	Spara verktygsomgivning (TOOLENV).....	469
11.12.2	Radera verktygsomgivning (DELTOOLENV).....	472
11.12.3	Läsa T-, D- och DL-nummer (GETTENV).....	473
11.12.4	Läsa informationer över sparade verktygsomgivningar (\$P_TOOLENVN, (\$P_TOOLENV).....	474
11.12.5	Läsa verktygslängder resp. -längdkomponenter (GETTCOR).....	474
11.12.6	Ända verktygskomponenter (SETTCOR).....	480
11.13	Läsa tillordningen av verktygslängderna L1, L2, L3 till koordinataxlarna (LENTOAX).....	493

12	Banbeteende.....	497
12.1	Tangentialstyrning.....	497
12.1.1	Definiera koppling (TANG).....	497
12.1.2	Koppla till mellanblocksgenerering (TLIFT).....	498
12.1.3	Koppla till koppling (TANGON).....	499
12.1.4	Koppla från koppling (TANGOF).....	501
12.1.5	Radera koppling (TANGDEL).....	501
12.2	Matningsförlopp (FNORM, FLIN, FCUB, FPO).....	503
12.3	Accelerationsbeteende.....	508
12.3.1	Accelerationsmode (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA).....	508
12.3.2	Påverkan av accelerationen vid följdxlar (VELOLIMA, ACCLIMA, JERKLIMA).....	510
12.3.3	Aktivering av teknologispecifika dynamikvärden (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH).....	512
12.4	Köra med förstyrning (FFWON, FFWOF).....	514
12.5	Programmerbar konturprecision (CPRECON, CPRECOF).....	515
12.6	Programförlopp med förberäkningsminne (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE)	517
12.7	Definiera Stop-Delay-områden (DELAYFSTON, DELAYFSTOF).....	520
12.8	Förhindra programställe för SERUPRO (IPTRLOCK, IPTRUNLOCK).....	523
12.9	Återstart på kontur (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)	525
12.10	Påverkan av rörelsestyrningen.....	534
12.10.1	Procentuell jerkkompensering (JERKLIM).....	534
12.10.2	Procentuell hastighetskompensering (VELOLIM).....	535
12.10.3	Programexempel för JERKLIM och VELOLIM.....	537
12.11	Programmera kontur-/orienteringtolerans (CTOL, OTOL, ATOL).....	538
12.12	Blockbytesbeteende vid aktiv koppling (CPBC).....	542
13	Axelkopplingar.....	543
13.1	Medsläpning (TRAILON, TRAILOF).....	543
13.2	Kurvtabeller (CTAB).....	547
13.2.1	Definiera kurvtabeller (CTABDEF, CATBEND).....	547
13.2.2	Kontrollera om en kurvtabell finns (CTABEXISTS).....	553
13.2.3	Radera kurvtabell (CTABDEL).....	554
13.2.4	Spärra kurvtabeller mot radering och överskrivning (CTABLOCK, CTABUNLOCK).....	555
13.2.5	Kurvtabeller: Fastställa tabellegenskaper (CTABID, CTABISLOCK, CTABMENTYP, CTABPERIOD).....	556
13.2.6	Läsa kurvtabellvärden (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX).....	557
13.2.7	Kurvtabeller: Kontrollera resursanvändning (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL).....	562
13.3	Axial styrvärdeskoppling (LEADON, LEADOF).....	564
13.4	Elektronisk växel (EG).....	569

13.4.1	Definiera elektronisk växel (EGDEF).....	569
13.4.2	Koppla till elektronisk växel (EGON, EGONSYN, EGONSYNE).....	570
13.4.3	Koppla från elektronisk växel (EGOFS, EGOFC).....	573
13.4.4	Radera definition för en elektronisk växel (EGDEL).....	574
13.4.5	Varvmatning (G95) / elektronisk växel (FPR).....	574
13.5	Synkronspindel.....	575
13.5.1	Synkronspindel: Programmering (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC).....	575
13.6	Generisk koppling (CP...).....	586
13.7	Master-/Slave-koppling (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS).....	593
14	Synkronaktioner.....	597
14.1	Definition av en synkronaktion.....	597
15	Pendling.....	599
15.1	Asynkron pendling (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB).	599
15.2	Via synkronaktioner styrd pendling (OSCILL).....	604
16	Stansning och nibbling.....	611
16.1	Aktivering/inaktivering.....	611
16.1.1	Koppla till/från stansning och nibbling (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC).....	611
16.2	Automatisk väguppdelning.....	616
16.2.1	Väguppdelning vid banaxlar.....	618
16.2.2	Väguppdelning vid singelaxlar.....	620
17	Slipning.....	623
17.1	Koppla till/från slipspecifik verktygsövervakning (TMON, TMOF).....	623
18	Ytterligare funktioner.....	625
18.1	Axelfunktioner (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL).....	625
18.2	Omkopplingsbara geometriaxlar (GEOAX).....	628
18.3	Axelcontainer (AXCTSWE, AXCTSWED, AXCTSWEC).....	633
18.4	Vänta på giltig axelposition (WAITENC).....	635
18.5	Programmerbar parameterblockomkoppling (SCPARA).....	636
18.6	Kontrollera förefintlig NC-språkomfattning (STRINGIS).....	638
18.7	Anropa fönster från detaljprogrammet interaktivt (MMC).....	642
18.8	Programkörningstid / arbetsstycksräknare.....	647
18.8.1	Programkörningstid.....	647
18.8.2	Arbetsstycksräknare.....	650
18.9	Process DataShare - utgåva på en extern apparat/fil (EXTOPEN, WRITE, EXTCLOSE)....	652
18.10	Larm (SETAL).....	657
18.11	Utvidgat stoppande och återdragande (ESR).....	658
18.11.1	NC-styrd ESR.....	659
18.11.1.1	NC-styrd återdragning (POLF, POLFA, POLFMASK, POLFMLIN).....	659

18.11.1.2	NC-styrt stoppande.....	662
18.11.2	Drivningsautarkt ESR.....	663
18.11.2.1	Projektera drivningsautarkt stoppande (ESRS).....	663
18.11.2.2	Projektera drivningsautarkt återdragande (ESRR).....	664
18.12	Definiera råämne (WORKPIECE).....	666
18.13	Koppla om språkmode (G290, G291).....	670
19	Egna avspåningsprogram.....	673
19.1	Stödjande funktioner för avspåningen.....	673
19.2	Upprätta konturtabell (CONTPRON).....	674
19.3	Upprätta kodad konturtabell (CONTDCON).....	680
19.4	Fastställa skärningspunkt mellan två konturelement (INTERSEC).....	684
19.5	Blockvis bortkörning av konturelement från en tabell (EXECTAB).....	686
19.6	Beräkna cirkeldata (CALCDAT).....	687
19.7	Koppla från konturförberedelse (EXECUTE).....	689
20	Programmera cykler externt.....	691
20.1	Teknologiska cykler.....	691
20.1.1	Inledning.....	691
20.1.2	Teknologis specifik översikt.....	692
20.1.3	HOLES1 - hålrad.....	694
20.1.4	HOLES2 - hålrad.....	694
20.1.5	POCKET3 - fräsa fyrkantficka.....	696
20.1.6	POCKET4 - fräsa cirkelsficka.....	699
20.1.7	SLOT1 - längsspår.....	701
20.1.8	SLOT2 - cirkelspår.....	704
20.1.9	LONGHOLE - långhål.....	706
20.1.10	CYCLE60 - gravrycykel.....	708
20.1.11	CYCLE61- planfräsa.....	711
20.1.12	CYCLE62 - konturanrop.....	713
20.1.13	CYCLE63 - fräsa konturficka.....	714
20.1.14	CYCLE64 - förborra konturficka.....	716
20.1.15	CYCLE70 - gängfräsning.....	717
20.1.16	CYCLE72 - banfräsa.....	719
20.1.17	CYCLE76 - fräsa fyrkanttapp.....	723
20.1.18	CYCLE77 - fräsa cirkeltapp.....	725
20.1.19	CYCLE78 - fräsa hålgänga.....	727
20.1.20	CYCLE79 - flerkant.....	729
20.1.21	CYCLE81 - borra, centrera.....	731
20.1.22	CYCLE82 - borra, plansänka.....	732
20.1.23	CYCLE83 - djuphålsborra.....	735
20.1.24	CYCLE84 - Gängtappning utan flytande gänghållare.....	738
20.1.25	CYCLE85 - brotscha.....	741
20.1.26	CYCLE86 - ursvarvning.....	742
20.1.27	CYCLE92 - avstick.....	743
20.1.28	CYCLE95 - konturavspåna.....	745
20.1.29	CYCLE98 - kopplade gängor.....	747
20.1.30	CYCLE99 - gängsvarvning.....	751

20.1.31	CYCLE435 - ställa in koordinatsystem för skärpverktyg.....	756
20.1.32	CYCLE495 - profilera.....	756
20.1.33	CYCLE800 - vridning.....	758
20.1.34	CYCLE801 - gitter eller ram.....	761
20.1.35	CYCLE802 - valfria positioner.....	763
20.1.36	CYCLE830 - djuphålsborra 2.....	765
20.1.37	CYCLE832 - High Speed Settings.....	771
20.1.38	CYCLE840 - gängtappning med flytande gänghållare.....	774
20.1.39	CYCLE899 - fräsa öppet spår.....	777
20.1.40	CYCLE930 - Instick.....	780
20.1.41	CYCLE940 - fristick former.....	782
20.1.42	CYCLE951- avspåna.....	785
20.1.43	CYCLE952 - kontursticka.....	788
20.1.44	CYCLE4071 - längsslipning med ansättning i vändpunkten.....	794
20.1.45	CYCLE4072 - längsslipning mit ansättning i vändpunkten.....	795
20.1.46	CYCLE4073 - längsslipning med kontinuerlig ansättning.....	799
20.1.47	CYCLE4074 - Längsslipning med kontinuerlig ansättning och avbrottssignal.....	800
20.1.48	CYCLE4075 - flatslipning med ansättning i vändpunkten.....	803
20.1.49	CYCLE4077 - flatslipning med ansättning i vändpunkten och avbrottssignal.....	806
20.1.50	CYCLE4078 - flatslipning med kontinuerlig ansättning.....	809
20.1.51	CYCLE4079 - flatslipning med intermittent ansättning.....	811
20.1.52	GROUP_BEGIN - början programblock.....	813
20.1.53	GROUP_END - slut programblock.....	814
20.1.54	GROUP_ADDEND - slut inkörningstillägg.....	814
20.1.55	Randvillkor.....	814
20.1.55.1	Teknologiskalning inom cykelmaskerna.....	814
20.2	Mätcykler.....	816
21	Tabeller.....	817
21.1	Anvisningar.....	817
21.2	Anvisningar: Disponibilitet vid SINUMERIK 828D	854
21.2.1	Styrningsvarianter fräsa / svarva.....	854
21.2.2	Styrningsvarianter slipa.....	881
21.3	Aktuellt språk i HMI.....	908
A	Bilaga.....	909
A.1	Förteckning över förkortningar.....	909
A.2	Dokumentationsöversikt.....	918
	Ordlista.....	919
	Index.....	941

Grundläggande säkerhetsanvisningar

1.1 Allmänna säkerhetsanvisningar

VARNING

Livsfara vid ej respekterande av säkerhetsanvisningar och resterande risker

Vid ej respekterande av säkerhetsanvisningar och resterande risker i den tillhörande hårdvarudokumentationen kan olyckor med svåra personskador eller dödsfall uppträda.

- Respektera säkerhetsanvisningarna i hårdvarudokumentationen.
- Ta hänsyn till de resterande riskerna vid riskbedömningen.

VARNING

Felfunktioner hos maskinen till följd av felaktiga eller förändrade parametrar

Genom felaktiga eller förändrade parametrar kan felfunktioner uppträda hos maskinen, som kan leda till kroppsskador eller dödsfall.

- Skydda parametrarna mot obehörig åtkomst.
- Behärska möjliga felfunktioner med lämpliga åtgärder, t. ex. NÖDSTOPP.

1.2 Garanti och ansvar för applikationsexempel

Användningsexempel är oförbindliga och gör inte anspråk på fullständighet beträffande konfiguration och utrustning samt alla eventualiteter. Användningsexempel utgör inga kundspecifika lösningar utan ska bara erbjuda hjälp för typiska uppgifter.

Som användare är du själv ansvarig för en sakkunnig drift av de beskrivna produkterna. Användningsexempel befriar dig inte från plikten av en säker hantering vid användning, installation, drift och underhåll.

1.3 Industrial Security

Märk

Industrial Security

Siemens erbjuder produkter och lösningar med Industrial Security-funktioner, som stöder en säker drift av anläggningar, system, maskiner och nätverk.

För att skydda anläggningar, system, maskiner och nätverk mot cyberhot är det nödvändigt att tillämpa (och kontinuerligt upprätthålla) ett enhetligt Industrial Security-koncept, som motsvarar aktuell teknisk standard. Produkter och lösningar från Siemens utgör endast en del av ett dylikt koncept.

Kunden ansvarar för att förhindra obehörig åtkomst till sina anläggningar, system, maskiner och nätverk. System, maskiner och komponenter bör endast kopplas till verksamhetens nätverk eller till internet om och i den utsträckning detta är nödvändigt, och lämpliga skyddsåtgärder (t. ex. användning av firewalls och nätverkssegmentering) vidtas.

Dessutom bör Siemens rekommendationer om motsvarande skyddsåtgärder observeras. Ytterligare information om Industrial Security finns på:

Industrial Security (<http://www.siemens.com/industrialsecurity>)

Produkter och lösningar från Siemens vidareutvecklas ständigt för att göra dem ännu säkrare. Siemens rekommenderar uttryckligen att uppdateringar genomförs så snart motsvarande uppdateringar står till förfogande och att alltid enbart aktuella produktversioner används. Användning av gamla versioner eller versioner som inte längre stöds kan höja risker för cyberhot.

För att alltid informeras om uppdateringar av produkter kan du beställa Siemens Industrial Security RSS Feed på:

Industrial Security (<http://www.siemens.com/industrialsecurity>)

Ytterligare information hittar du på internet:

Projekteringshandbok Industrial Security (<https://support.industry.siemens.com/cs/ww/en/view/108862708>)



VARNING

Osäkra driftstillstånd på grund av manipulering av mjukvaran

Manipuleringar av mjukvaran (t. ex. virus, trojaner, malware eller maskar) kan förorsaka osäkra driftstillstånd i anläggningen som kan leda till dödsfall, svåra personskador och materialskador.

- Håll mjukvaran aktuell.
- Integrera automatiserings- och driftskomponenterna i ett Industrial Security helhetskoncept för anläggningen eller maskinen enligt teknikens aktuella stånpunkt.
- Ta vid Industrial Security helhetskonceptet hänsyn till alla använda produkter.
- Skydda filerna i utbytbara minnesmedier för skadlig programvara genom lämpliga skyddsåtgärder, t. ex. virusskanner.
- Skydda drivenheten för obehöriga ändringar, genom att aktivera omformarfunktionen "Know-How-skydd".

Flexibel NC-programmering

2.1 Variabler

Genom användning av variabler från områdena systemdata och användardata speciellt i förbindelse med räknefunktioner och kontrollstrukturer kan NC-program och cykler gestaltas flexibelt.

- **Systemdata**
I systemdata ligger de i systemet fördefinierade variablerna. Dessa variabler har en definierad betydelse. De används i första hand av systemsoftware. Dessa variabler i NC-program och cykler kan läsas och skrivas av användaren. Exempel: Maskindata, settingdata, systemvariabler.
Fastän betydelsen för ett systemdatum är fast föreskriven, kan egenskaperna i viss omfattning ändras av användaren genom redefinition.
Se "Redefinition av systemdata, användardata och NC-språkkommandon (REDEF) (Sida 34)"
- **Användardata**
I användardata ligger de av användaren definierade variablerna, vars betydelse bestämts enbart av användaren. De utvärderas inte av systemet.
Användardata är uppdelade i:
 - **Fördefinierade användarvariabler**
Fördefinierade användarvariabler är variabler som redan är definierade i systemet och vars antal parametreras via maskindata. Egenskaperna för dessa variabler kan anpassas av användaren.
Se "Redefinition av systemdata, användardata och NC-språkkommandon (REDEF) (Sida 34)".
 - **Användardefinierade variabler**
Användardefinierade variabler är variabler som definieras av användaren och som läggs till av systemet vid körtiden. Dess antal, datatyp, synlighet och alla andra egenskaper fastläggs uteslutande av användaren.
Se "Definition av användarvariabler (DEF) (Sida 28)"

2.1.1 Systemdata

Systemdata innehåller de i systemet fördefinierade variabler som i NC-program och cykler möjliggör åtkomst till den aktuella parametreringen av styrningen, samt till maskin-, styrnings- och processtillstånd.

Förlöpsvariabler

Förlöpsvariabler är systemdata som läses och skrivs i samband med förlöpningen dvs. vid tidpunkten för tolkningen av det block i vilket systemvariabeln är programmerad. Förlöpsvariabel utlöser inget fördekoderingsstopp.

Huvudloppsvariabler

Huvudloppsvariabler är systemdata som läses och skrivs i samband med huvudloppet dvs. vid tidpunkten för utförandet av det block i vilket variabeln är programmerad.

Huvudloppsvariabler är:

- Variabel som kan programmeras i synkronaktioner (läsa/skriva)
- Variabel som kan programmeras i NC-programmet och utlöser fördekoderingsstopp (läsa/skriva)
- Variabel som kan programmeras i NC-programmet och vars värde fastställs under förkörningen, men skrivs först under huvudloppet (huvudlopp-synkront: endast skriva)

Prefix-systematik

För speciell märkning av systemdata placeras i normalfall före namnet ett prefix som är sammansatt av \$-tecknet följt av en eller två bokstäver och ett nedsänkt streck:

\$ + 1:a bokstav	Betydelse: Datatyp
Förberäkningsdata (Systemdata som läses / skrivs under förkörningen)	
\$M	Maskindata ¹⁾
\$S	Settingdata, skyddsområden ¹⁾
\$T	Verktygförvaltningsdata
\$P	Programmerade värden
\$C	Cyklervariabler till ISO-Hüllcykler
\$O	Optionsdata
R	R-parameter (räkneparameter) ²⁾
Huvudkörningsdata (Systemdata som läses / skrivs under huvudloppet)	
\$\$M	Maskindata ¹⁾
\$\$S	Settingdata ¹⁾
\$A	Aktuella huvudloppsdata
\$V	Lägesregulator-data
\$R	R-parameter (räkneparameter) ²⁾
¹⁾ Om maskin- och settingdata behandlas som för- eller huvudkörningsvariabler beror på om de skrivs med ett eller två \$-tecken. Skrivsättet kan väljas fritt användningsspecifikt. ²⁾ Vid användningen av en R-parameter i detaljprogram / cykel som förkörningsvariabel skrivs inget prefix, t.ex. R10. Vid användningen i en synkronaktion som huvudloppsvariabel skrivs \$-tecken som prefix, t.ex. \$R10.	

2:a bokstaven	Betydelse: Synlighet
N	NC-global variabel (NC)
C	Kanalspecifik variabel (Channel)
A	Axelspecifik variabel (Axis)

Randvillkor

Undantag i prefix-systematiken

Följande systemvariabler avviker från den ovan nämnda prefix-systematiken:

- \$TC_...: Den 2:a bokstaven C hänvisar här inte på kanalspecifika utan på verktygshållarspecifika systemvariabler (TC = Tool Carrier)
- \$P_ ...: Kanalspecifika systemvariabler

Användning av maskin- och settingdata i synkronaktioner

Vid användningen av maskin- och settingdata i synkronaktioner kan med prefixet bestämmas om maskin- eller settingdata läses/skrivs förlopps- eller huvudloppssynkront.

Förblir datumet oförändrat under bearbetningen kan läsas förloppssynkront. Prefixet för maskin- eller settingdatum skrivs för detta med ett \$-tecken:

```
ID=1 WHENEVER $AA_IM[z] < $SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Förändras datum under bearbetningen måste läsas / skrivas huvudloppssynkront. Prefixet för maskin- eller settingdatum skrivs för detta med två \$-tecken:

```
ID=1 WHENEVER $AA_IM[z] < $$SA_OSCILL_REVERSE_POS2[Z]-6 DO $AA_OVR[X]=0
```

Märk

Skrivning av maskin- och settingdata

Vid skrivning av ett maskin- eller settingdata ska ges akt på att det aktiva åtkomststeget vid utförandet av detaljprogrammet / cykeln tillåter skrivåtkomst och att verkan hos datum är "IMMEDIATE".

Litteratur

En fullständig översikt för alla systemvariabler finns i:

Handbok med listor Systemvariabler

Se även

Variabler (Sida 21)

2.1.2 Fördefinierade användarvariabler: Räkneparameter

2.1.2.1 Kanalspecifika räkneparametrar (R)

Kanalspecifika räkneparametrar eller R-parametrar är fördefinierade användarvariabler med beteckningen R, definierade som fält av datatyp REAL. Av historiska orsaker är för R-parametrar förutom skrivsättet med fältindex t.ex. R[10] också skrivsättet utan fältindex t.ex. R10 tillåtet.

Vid användning i synkronaktioner måste bokstaven \$ placeras före t.ex. \$R10.

Syntax

Vid användning som förlöpsvariabel:

R<n>

R[<Ausdruck>]

Vid användning som huvudloppsvariabel:

\$R<n>

\$R[<Ausdruck>]

Betydelse

R:	Beteckning vid användning som förkörningsvariabel t.ex. i detaljprogram	
\$R:	Beteckning vid användning som huvudkörningsvariabel t. ex. i synkronaktioner	
	Typ:	REAL
	Värdeområde:	Vid ej exponentiellt skrivsätt: $\pm (0.000\ 0001 \dots 9999\ 9999)$ Observera: Maximalt 8 decimaler
		Vid exponentiellt skrivsätt: $\pm (1*10^{-300} \dots 1*10^{+300})$ Observera: - Skrivsätt: <Mantissa>EX<exponent> t.ex. 8.2EX-3 - Maximalt 10 tecken inklusive förtecken och decimalpunkt är tillåtna.
<n>:	Nummer för R-parametern	
	Typ:	INT
	Värdeområde:	0 - MAX_INDEX Observera MAX_INDEX resulterar ur det parametrerade antalet R-parametrar: MAX_INDEX = (MD28050 \$MN_MM_NUM_R_PARAM) - 1
<Ausdruck>:	Fältindex Som fältindex kan ett valfritt uttryck anges så länge som resultatet av uttrycket kan omvandlas till datatypen INT (INT, REAL, BOOL, CHAR)	

Exempel

Tillordningar av R-parametrar och användning av R-parametrar i matematiska funktioner:

Programkod	Kommentar
R0=3.5678	; Tillordning i förlöpningen
R[1]=-37.3	; Tillordning i förlöpningen
R3=-7	; Tillordning i förlöpningen
\$R4=-0.1EX-5	; Tillordning i huvudloppet: R4 = -0.1 * 10 ⁻⁵

Programkod	Kommentar
\$R[6]=1.874EX8	; Tillordning i huvudloppet: R6 = 1.874 * 10 ⁸
R7=SIN(25.3)	; Tillordning i förlöpnigen
R[R2]=R10	; Indirekt adressering via R-parametrar
R[(R1+R2)*R3]=5	; Indirekt adressering via matematiska uttryck
X=(R1+R2)	; Förflyttar axel X till den position som resulterar ur summan av R1 och R2
Z=SQRT(R1*R1+R2*R2)	; Förflyttar axel Z till positionen kvadratroten ur (R1 ² + R2 ²)

Se även

Variabler (Sida 21)

2.1.2.2 Globala räkneparametrar (RG)

Funktion

Förutom de kanalspecifika R-parametrarna står också globala R-parametrar användaren till förfogande. De existerar en gång inom styrningen och kan läsas/skrivas från alla kanaler.

Globala R-parametrar används t.ex. får att få informationer från en kanal till den nästa. Ett annat exempel är globala inställningar, som ska utvärderas för alla kanaler, som t. ex. utkragningen av råämnet ur spindeln.

Läsning och skrivning av de globala R-parametrarna sker via användargränssnittet eller i NC-programmet under förkörningen. En användning i synkronaktioner eller teknologicykler är inte möjlig.

Märk

Vid läsning och skrivning av globala R-parametrar sker **ingen** synkronisering mellan kanalerna.

Eftersom läsningen och skrivningen sker under förkörningen, är den tidpunkt vid vilken ett skrivet värde blir verksamt från en kanal till en annan, inte definierad.

Exempel:

I kanal1 löper en slinga med en global R-prameter som slingräknare. Kanal 2 skriver ett värde i denna globala R-parameter, som leder till ett slingavbrott i kanal 1. Alla fram till denna tidpunkt i kanal 1 under förkörningen interpreterade slingor blir dock utförda. Wie viele Schleifen das sind ist nicht definiert und hängt unter anderem von der Auslastung des Kanals ab.

En synkronisering mellan kanalerna måste användaren praktiskt, t. ex. med WAIT-märken, själv realisera!

Syntax

Skriva i NC-programmet

RG[<n>]=<Wert>

RG[<Ausdruck>]=<Wert>

Läsa i NC-programmet

R...=RG[<n>]

R...=RG[<Ausdruck>]

Betydelse

RG:	Defaultnamn för NC-adressen för globala R-parametrar Observera: Namnet för NC-adressen är inställbar via MD15800 \$MN_R_PA-RAM_NCK_NAME	
<n>:	Nummer för den globala R-parametern	
	Typ:	INT
	Värdeområde:	0 ... MAX_INDEX Observera MAX_INDEX resulterar ur det parametrerade antalet globala R-parametrar: MAX_INDEX = (MD18156 \$MN_MM_NUM_R_PA-RAM_NCK) - 1
<Ausdruck>:	Som fältindex kan ett valfritt uttryck anges så länge som resultatet av uttrycket kan omvandlas till datatypen INT (INT, REAL, BOOL, CHAR)	
<Wert>:	Värde för den globala R-parametern	
	Typ:	REAL
	Värdeområde:	Vid ej exponentiellt skrivsätt: $\pm (0.000\ 0001 \dots 9999\ 9999)$ Observera: Maximalt 8 decimaler tillåtna
		Vid exponentiellt skrivsätt: $\pm (1*10^{-300} \dots 1*10^{+300})$ Observera: - Skrivsätt: <Mantissa>EX<exponent> t.ex. 8.2EX-3 - Maximalt 10 tecken inklusive förtecken och decimalpunkt är tillåtna.

2.1.3 Fördefinierade användarvariabler: Link-variabler

Via link-variabler kan inom ramen för funktionen "NCU-Link" data bytas ur cykliskt mellan NCUs, som är förbundna i ett nätverk med varandra. De möjliggör därvid en dataformat-specifik åtkomst till link-variabelminnet. Link-variabelminnet fastläggs anläggningsspecifikt av användaren / maskintillverkaren både vad gäller storlek och även datastrukturen.

Link-variabler är systemglobala användarvariabler som vid projekterad link-kommunikation kan läsas och skrivas i detaljprogram och cykler av alla NCUs i link-förbandet. I motsats till globala användarvariabler (GUD) kan link-variabler också användas i synkronaktioner.

Vid anläggningar utan aktiv NCU-link, kan link-variabler styrningslokalt vid sidan av de globala användarvariablerna (GUD), användas som extra globala användarvariabler.

Syntax

```
$A_DLB [<Index>]
$A_DLW [<Index>]
$A_DLD [<Index>]
$A_DLR [<Index>]
```

Betydelse

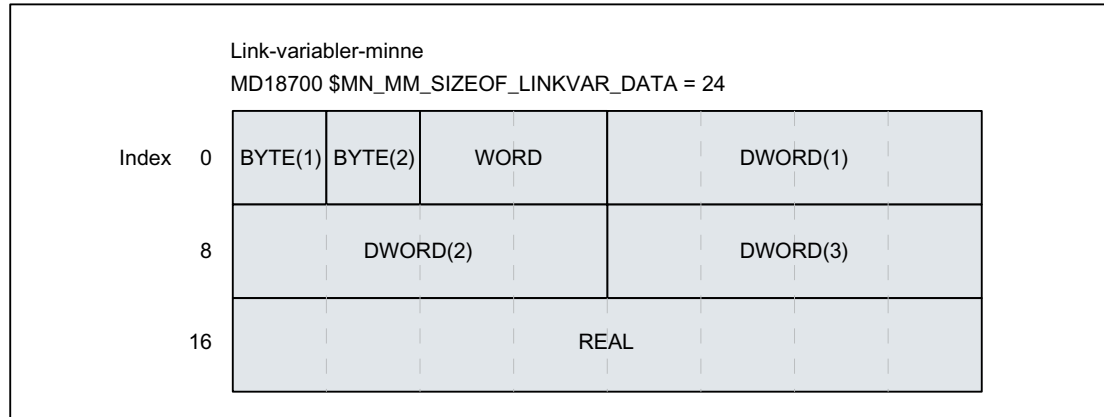
\$A_DLB:	Link-variabel för dataformatet BYTE (1 byte)	
	Datatyp:	UINT
	Värdeområde:	0 ... 255
\$A_DLW:	Link-variabel för dataformatet WORD (2 bytes)	
	Datatyp:	INT
	Värdeområde:	-32768 ... 32767
\$A_DLD:	Link-variabel för dataformatet DWORD (4 bytes)	
	Datatyp:	INT
	Värdeområde:	-2147483648 ... 2147483647
\$A_DLR:	Link-variabel för dataformatet REAL (8 bytes)	
	Datatyp:	REAL
	Värdeområde:	$\pm(2,2 \cdot 10^{-308} \dots 1,8 \cdot 10^{+308})$
<Index>:	Adressindex i byte, räknat från början av link-variabelminnet	
	Datatyp:	INT
	Värdeområde:	0 - MAX_INDEX Observera - MAX_INDEX resulterar ur den parameterade storleken på link-variabelminnet: $\text{MAX_INDEX} = (\text{MD18700 } \\$\text{MN_MM_SIZEOF_LINKVAR_DATA}) - 1$ - Det får endast programmeras index så att de i link-variabelminnet adresserade bytes ligger på en dataformatgräns $\Rightarrow$ Index = n * bytes, mit 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, ...

Exempel

I automatiseringsanläggningen finns 2 NCUs (NCU1 och NCU2). Till NCU1 är maskinaxel AX2 ansluten som förflyttas som link-axel av NCU2.

NCU1 skriver cykliskt strömvärdet (\$VA_CURR) för axel AX2 i link-variabelminnet. NCU2 läser cykliskt det per link-kommunikation överförda strömvärdet och visar vid överskridande av gränsvärdet larm 61000.

Datastrukturen i link-variabelminnet visas i följande bild. Strömvärdet överförs via REAL-värdet.



NCU1

NCU1 skriver i en statisk synkronaktion cykliskt i IPO-takt strömvärdet för axel AX2 via link-variabeln \$A_DLR[16] i link-variabelminnet.

Programkod

```
N111 IDS=1 WHENEVER TRUE DO $A_DLR[16]=$VA_CURR[AX2]
```

NCU2

NCU2 läser i en statisk synkronaktion cykliskt i IPO-takt strömvärdet för axel AX2 via link-variabeln \$A_DLR[16] från link-variabelminnet. Är strömvärdet större än 23.0 A, visas larm 61000.

Programkod

```
N222 IDS=1 WHEN $A_DLR[16] > 23.0 DO SETAL(61000)
```

Se även

Variabler (Sida 21)

2.1.4 Definition av användarvariabler (DEF)

Med kommandot `DEF` kan användarspecifika variabler eller användarvariabler (User Data), definieras och beläggas med värden.

I enlighet med gitighetsområdet dvs. det område i vilket variablen är synlig finns det följande kategorier av användarvariabler:

- **Lokala användarvariabler (LUD)**
Lokala användarvariabler (LUD) är variabler som är definierade i ett NC-program som vid tidpunkten för genomarbetningen inte är huvudprogram. De läggs till vid anrop av NC-programmet och raderas med programslut-reset resp. den nästa styrningsstarten. Till LUD finns åtkomst endast inom det NC-program i vilket de är definierade.
- **Programglobala användarvariabler (PUD)**
Programglobala användarvariabler (PUD) är variabler som är definierade i ett som huvudprogram använt NC-program. De läggs till vid anrop av NC-programmet och raderas med programslut-reset resp. den nästa styrningsstarten. Till PUD finns åtkomst i huvudprogrammet och i alla underprogram.

Märk

Disponibilitet för programglobala användarvariabler (PUD)

I huvudprogrammet definierade programglobala användarvariabler (PUD) är också disponibla i underprogrammen endast när följande maskindatum är inställt:

MD11120 \$MN_LUD_EXTENDED_SCOPE = 1

Med MD11120 = 0 är de i huvudprogrammet definierade programglobala användarvariablerna endast disponibla i huvudprogrammet.

- **Globala användarvariabler (GUD)**
Globala användarvariabler (GUD) är NC- resp. kanal-globala variabler som är definierade i en datakomponent (SGUD, MGUD, UGUD, GUD4 ... GUD9) och bibehålls också efter programslut-reset resp. den nästa styrningsstarten. Till GUD finns åtkomst i alla NC-program.

Användarvariabler måste vara definierade före sin användning (läsa / skriva). Följande regler ska därvid iakttas:

- GUD måste definieras i en definitionsfil, t. ex. _N_DEF_DIR/_N_UGUD_DEF.
- PUD och LUD måste definieras i definitionsfilen i ett NC-jprogram.
- Datadefinitionen måste göras i ett eget block.
- Per datadefinition får endast en datatyp användas.
- Per datadefinition kan flera variabler av samma datatyp definieras.

Syntax

LUD och PUD

```
DEF <Typ> <Phys_Einheit> <Grenzwerte> <Name>[<Wert_1>, <Wert_2>,
<Wert_3>]=<Init_Wert>
```

GUD

```
DEF <Bereich> <VL_Stop> <Zugriffsrechte> <Datenklasse> <Typ>
<Phys_Einheit> <Grenzwerte> <Name>[<Wert_1>, <Wert_2>,
<Wert_3>]=<Init_Wert>
```

Betydelse

DEF:	Kommando för definition av användarvariabler GUD, PUD, LUD	
<Bereich>:	Giltighetsområde endast relevant för GUD:	
	NC:	NC-global användarvariabel
	CHAN:	Kanal-global användarvariabel
<VL_Stop>:	Fördekoderingsstopp endast relevant för GUD (option)	
	SYNR:	Fördekoderingsstopp vid läsning
	SYNW:	Fördekoderingsstopp vid skrivning
	SYNRW:	Fördekoderingsstopp vid läsning/skrivning
<Zugriffsrechte>:	Skyddsnivå för läsning / skrivning av GUD via NCjprogram eller BTSS (option)	
	APRP <Schutzstufe>:	Läsa: NC-program
	APWP <Schutzstufe>:	Skriva: NC-program
	APRB <Schutzstufe>:	Läsa: BTSS
	APWP <Schutzstufe>:	Skriva: BTSS
	<Schutzstufe>:	Värdeområde: 0 ... 7
	Se "Attribut: Åtkomsträtter (APR, APW, APRP, APWP, APRB, APWB) (Sida 44)"	
<Datenklasse>:	Dataklasstillordning (endast SINUMERIK 828D!)	
	DCM:	Dataklass M (= Manufacturer)
	DCI:	Dataklass I (= Individual)
	DCU:	Dataklass U (= User)
	Se "Attribut: Dataklass (DCM, DCI, DCU) - endast SINUMERIK 828D (Sida 49)".	
<Typ>:	Datatyp:	
	INT:	Heltaligt värde med förtecken
	REAL:	Real-tal (LONG REAL efter IEEE)
	BOOL:	Sanningsvärde TRUE (1) / FALSE (0)
	CHAR:	ASCII-tecken
	STRING [<MaxLänge>]:	Teckenföljd definierad längd
	AXIS:	Axel/spindelbeteckning
	FRAME:	Geometrisk uppgifter för en statisk koordinatstransformation
	Se "Datatyper (Sida 56)"	
<Phys_Einheit>:	Fysikalisk enhet (option)	
	PHU <Einheit>:	Fysikalisk enhet
	Se "Attribut: Fysikalisk enhet (PHU) (Sida 41)"	
<Grenzwerte>:	Undre / övre gränsvärde (option)	
	LLI <Grenzwert>:	Undre gränsvärde (lower limit)
	ULI <Grenzwert>:	Övre gränsvärde (upper limit)
	Se "Attribut: Gränsvärden (LLI, ULI) (Sida 40)"	

<Namn>:	Namn för variabler Observera - Maximalt 31 tecken - De båda första tecknen måste vara en bokstav och/eller ett nedsänkt streck. "\$"-tecknet är reserverat för systemvariabler och får inte användas.
[<Wert_1>, <Wert_2>, <Wert_3>]:	Angivande av fältstorlekar för 1- till max. 3-dimensionala fältvariabler (option) För initiering av fältvariabler se "Definition och initiering av fältvariabler (DEF, SET, REP) (Sida 51)"
<Init_Wert>:	Initieringsvärde (option) se "Attribut: Initieringsvärde (Sida 37)" För initiering av fältvariabler se "Definition och initiering av fältvariabler (DEF, SET, REP) (Sida 51)"

Exempel

Exempel 1: Definitioner av användarvariabler i datakomponenten för maskintillverkaren

Programkod	Kommentar
<pre>%_N_MGUD_DEF \$PATH=/_N_DEF_DIR DEF CHAN REAL PHU 24 LLI 0 ULI 10 STROM_1, STROM_2 ; Beskrivning ; Definition av två GUD: STROM_1, STROM_2 ; Giltighetsområde: Kanalomfattande ; Datatyp: REAL ; VL-stopp: ej programmerat => Defaultwert = inget VL-stopp ; Fys. enhet: 24 = [A] ; Gränsvärden Low = 0.0, High = 10.0 ; Åtkomsträtter: ej programmerade => Defaultwert = 7 = nyckelbrytarläge 0 ; Initieringsvärde ej programmerat => Defaultwert = 0.0 DEF NCK REAL PHU 13 LLI 10 APWP 3 APRP 3 APWB 0 APRB 2 ZEIT_1=12, ZEIT_2=45 ; Beskrivning ; Definition av två GUD: ZEIT_1, ZEIT_2 ; Giltighetsområde: NC-täckande ; Datatyp: REAL ; VL-stopp: ej programmerat => Defaultwert = inget VL-stopp ; Fys. enhet: 13 = [s] ; Gränsvärden Low = 10.0, High = ej programmerat => övre definitionsområdesgräns ; Åtkomsträtter: ; NC-programm: skriva/läsa = 3 = slutanvändare ; BTSS: skriva = 0 = Siemens, läsa = 3 = slutanvändare ; Initieringsvärde ZEIT_1 = 12.0, ZEIT_2 = 45.0</pre>	<pre>; GUD-komponent: Maskintillverkare</pre>

2.1 Variabler

Programkod	Kommentar
DEF NCK APWP 3 APRP 3 APWB 0 APRB 3 STRING[5] GUD5_NAME = "COUNTER"	
; Beskrivning	
; Definition av en GUD: GUD5_NAME	
; Giltighetsområde: NC-täckande	
; Datatyp: STRING, max. 5 tecken	
; VL-stopp: ej programmerat => Defaultwert = inget VL-stopp	
; Fys. enhet: ej programmerad => Defaultwert = 0 = ingen fys. enhet	
; Gränsvärden ej programmerade => definitionsområdesgränser: Low = 0, High = 255	
; Åtkomsträtter:	
; NC-programm: skriva/läsa = 3 = slutanvändare	
; BTSS: skriva = 0 = Siemens, läsa = 3 = slutanvändare	
; Initieringsvärde "COUNTER"	
M30	

Exempel 2: Program-globala och -lokala användarvariabler (PUD / LUD)

Programkod	Kommentar
PROC MAIN	; Huvudprogram
DEF INT VAR1	; PUD-definition
...	
SUB2	; Underprogramanrop
...	
M30	

Programkod	Kommentar
PROC SUB2	; Underprogram SUB2
DEF INT VAR2	; LUD-DEFINITION
...	
IF (VAR1==1)	; Läsa PUD
VAR1=VAR1+1	; Läsa och skriva PUD
VAR2=1	; Skriva LUD
ENDIF	
SUB3	; Underprogramanrop
...	
M17	

Programkod	Kommentar
PROC SUB3	; Underprogram SUB3
...	
IF (VAR1==1)	; Läsa PUD
VAR1=VAR1+1	; Läsa och skriva PUD
VAR2=1	; Fel: LUD från SUB2 inte känd
ENDIF	

Programkod	Kommentar
...	
M17	

Exempel 3: Definition och användning av användarvariabler av datatyp AXIS

Programkod	Kommentar
DEF AXIS ABSZISSE	; 1:a geometriaxeln
DEF AXIS SPINDLE	; Spindel
...	
IF ISAXIS(1) == FALSE GOTO WEITER	
ABSZISSE = \$P_AXN1	
WEITER:	
...	
SPINDLE=(S1)	; 1:a spindeln
OVRA[SPINDLE]=80	; Spindeloverride = 80%
SPINDLE=(S3)	; 3:e spindeln

Randvillkor

Globala användarvariabler (GUD)

Inom ramen för definitionen av globala användarvariabler (GUD) ska det tas hänsyn till följande maskindata:

Nr	Beteckning: \$MN_	Betydelse
11140	GUD_AREA_SAVE_TAB	Extra säkring för GUD-komponent
18118 ¹⁾	MM_NUM_GUD_MODULES	Antal GUD-filer i det aktiva filesystemet
18120 ¹⁾	MM_NUM_GUD_NAMES_NCK	Antal globala GUD-namn
18130 ¹⁾	MM_NUM_GUD_NAMES_CHAN	Antal kanalspec. GUD-namn
18140 ¹⁾	MM_NUM_GUD_NAMES_AXIS	Antal axelspec. GUD-namn
18150 ¹⁾	MM_GUD_VALUES_MEM	Minnesplats för globala GUD-värden
18660 ¹⁾	MM_NUM_SYNACT_GUD_REAL	Antal projekterbara GUD datatyp REAL
18661 ¹⁾	MM_NUM_SYNACT_GUD_INT	Antal projekterbara GUD datatyp INT
18662 ¹⁾	MM_NUM_SYNACT_GUD_BOOL	Antal projekterbara GUD datatyp BOOL
18663 ¹⁾	MM_NUM_SYNACT_GUD_AXIS	Antal projekterbara GUD datatyp AXIS
18664 ¹⁾	MM_NUM_SYNACT_GUD_CHAR	Antal projekterbara GUD datatyp CHAR
18665 ¹⁾	MM_NUM_SYNACT_GUD_STRING	Antal projekterbara GUD datatyp STRING

¹⁾ MD vid SINUMERIK 828D endast läsbar!

Kanalövergripande användning av en NC-global användarvariabel av datatyp AXIS

En NC-global användarvariabel av datatyp `AXIS`, som vid definitionen i datakomponenten initierades med en axelbeteckning, kan användas i olika kanaler till NC endast när axeln har samma kanalaxelnummer i dessa kanaler.

Är detta inte fallet måste variabeln laddas vid NC-programmets början eller användas som i följande exempel funktionen AXNAME(...) (se "Axelfunktioner (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL) (Sida 625)").

Programkod	Kommentar
DEF NCK STRING[5] ACHSE="X"	; Definition i datakomponent
...	
N100 AX[AXNAME(ACHSE)]=111 G00	; Användning i NC-programm

2.1.5 Redefinition av systemdata, användardata och NC-språkkommandon (REDEF)

Med kommandot **REDEF** kan attributen till systemdata, användardata och NC-språkkommandon ändras. Grundförutvättning för en redefinition är att den tidsmässigt utförs efter den motsvarande definitionen.

Vid en redefinition kan inte flera attribut ändras samtidigt. För varje attribut som ska ändras måste en egen **REDEF**-anvisning programmeras.

Programmeras flera konkurrerande attributändringar blir alltid den sista ändringen aktiv.

Återställa attributvärden

De med **REDEF** ändrade attributen för åtkomsträtter och initieringstidpunkt kan återställas genom förnyad programmering av **REDEF**, följt av namnet på variabeln eller NC-språkkommandot till ditt defaultvärde:

- Åtkomsträtter: Skyddsnivå 7
- Initieringstidpunkt: ingen initiering resp. bibehållande av det aktuella värdet

Redefinierbara attribut

Se "Översikt över definierbara och redefinierbara attribut (Sida 50)".

Lokala användarvariabler (PUD / LUD)

För lokala användarvariabler (PUD / LUD) får inga redefinitioner göras.

Syntax

```
REDEF <Name> <VL_Stop>
REDEF <Name> <Phys_Einheit>
REDEF <Name> <Grenzwerte>
REDEF <Name> <Zugriffsrechte>
REDEF <Name> <Init_Zeitpunkt>
REDEF <Name> <Init_Zeitpunkt> <Init_Wert>
REDEF <Name> <Datenklasse>
REDEF <Name>
```

Betydelse

REDEF:	Kommando för redefinition av ett bestämt attribut resp. till återställande av attributen "Åtkomsträtter" och/eller "Initieringstidpunkt" till systemvariabler, användarvariabler och NC-språkkommandon	
<Namn>:	Namn på en redan definierad variabel eller ett NC-språkkommando	
<VL-Stopp>:	Fördekoderingsstopp	
	SYNR:	Fördekoderingsstopp vid läsning
	SYNW:	Fördekoderingsstopp vid skrivning
	SYNRW:	Fördekoderingsstopp vid läsning/skrivning
<Phys_Einheit>:	Fysikalisk enhet	
	PHU <Einheit>:	Fysikalisk enhet
	Se "Attribut: Fysikalisk enhet (PHU) (Sida 41)".	
	Observera Inte redefinierbar för: • Systemvariabler • Globala användardata (GUD) för datatyperna: BOOL, AXIS, STRING, FRAME	
<Grenzwerte>:	Undre / övre gränsvärde	
	LLI <Grenzwert>:	Undre gränsvärde (lower limit)
	ULI <Grenzwert>:	Övre gränsvärde (upper limit)
	Se "Attribut: Gränsvärden (LLI, ULI) (Sida 40)". Observera Inte redefinierbar för: • Systemvariabler • Globala användardata (GUD) för datatyperna: BOOL, AXIS, STRING, FRAME	
<Zugriffsrechte>:	Åtkomsträtter för läsning / skrivning via detaljprogram eller BTSS	
	APX <Schutzstufe>:	Utföra: NC-språkelement
	APRP <Schutzstufe>:	Läsa: Detaljprogram
	APWP <Schutzstufe>:	Skriva: Detaljprogram
	APRB <Schutzstufe>:	Läsa: BTSS
	APWP <Schutzstufe>:	Skriva: BTSS
	<Schutzstufe>:	Värdeområde: 0 ... 7
	Se "Attribut: Åtkomsträtter (APR, APW, APRP, APWP, APRB, APWB) (Sida 44)".	

<Init_Zeitpunkt>:	Tidpunkt till vilken variabeln reinitieras	
	INIPO:	Power On
	INIRE:	Huvudprogramslut, NC-Reset eller Power On
	INICF:	NEWCONF eller huvudprogramslut, NC-Reset eller Power On
	PRLOC:	Huvudprogramslut, NC-Reset efter lokal ändring eller Power On
Se "Attribut: Initieringsvärde (Sida 37)".		
<Init_Wert>:	Initieringsvärde	
	Vid redefinition av initieringsvärdet måste alltid också en initieringstidpunkt (se <Init_Zeitpunkt>) anges. Se "Attribut: Initieringsvärde (Sida 37)". För initiering av fältvariabler se "Definition och initiering av fältvariabler (DEF, SET, REP) (Sida 51)" Observera Inte redefinierbar för systemvariabler, förutom settingdata.	
<Datenklasse>:	Dataklassstillordning (endast SINUMERIK 828D!)	
	DCM:	Dataklass M (= Manufacturer)
	DCI:	Dataklass I (= Individual)
	DCU:	Dataklass U (= User)
	Se "Attribut: Dataklass (DCM, DCI, DCU) - endast SINUMERIK 828D (Sida 49)".	

Exempel

Redefinitionen av systemvariabeln \$TC_DPCx i datakomponenten för maskintillverkare

Programkod

```

%_N_MGUD_DEF                                ; GUD-komponent: Maskintillverkare
N100 REDEF $TC_DPC1 APWB 2 APWP 3
N200 REDEF $TC_DPC2 PHU 21
N300 REDEF $TC_DPC3 LLI 0 ULI 200
N400 REDEF $TC_DPC4 INIPO (100, 101, 102, 103)
N800 REDEF $TC_DPC1
N900 REDEF $TC_DPC4
M30

```

till N100: Åtkomsträtt skriva: BTSS = skyddsnivå 2, detaljprogram = skyddsnivå 3

till N200: Fysikalisk enhet [%]

till N300: undre gränsvärde = 0, övre gränsvärde = 200

till N400: Fältvariabeln initieras vid PowerOn med de fyra värdena.

till Återställande av attributvärdena "Åtkomsträtter" och/eller "Initieringstidpunkt"

N800 /

N900

Märk**Användning av ACCESS-filer**

Vid användning av ACCESS-filer måste redefinitionen av åtkomsträtterna till `_N_MGUD_DEF` förflyttas till `_N_MACCESS_DEF`.

Randvillkor**Granularitet**

En redefinition hänför sig alltid till hela med sitt namn entydigt betecknade variabeln. Det är inte möjligt att tillordna t. ex. vid fältvariabler olika attributvärden för enskilda fältelement.

2.1.6 Attribut: Initieringsvärde**Definition (DEF) av användarvariabler**

Vid definitionen kan ett initieringsvärde föreskrivas för följande användarvariabler:

- Globala användarvariabler (GUD)
- Programglobala användarvariabler (PUD)
- Lokala användarvariabler (LUD)

Redefinition (REDEF) av system- och användarvariabler

Vid redefinitionen kan ett initieringsvärde föreskrivas för följande variabler:

- Systemdata
 - Settingdata
- Användardata
 - R-parametrar
 - Synkronaktionsvariabel (\$AC_MARKER, \$AC_PARAM, \$AC_TIMER)
 - Synkronaktions-GUD (SYG_xy[], med x=R, I, B, A, C, S och y=S, M, U, 4, ..., 9)
 - EPS-parametrar
 - Verktygsdata-OEM
 - Magasindata-OEM
 - Globala användarvariabler (GUD)

Reinitieringstidpunkt

Vid redefinitionen kan tidpunkt anges vid vilken variabeln reinitieras dvs. åter ska ställas in på initieringsvärdet:

- **INIPO (Power On)**
Variabeln reinitieras vid Power On.
- **INIRE (Reset)**
Variabeln reinitieras vid NC-Reset, BAG-Reset, detaljprogramslut (M02 / M30) eller Power On.
- **INICF (NEWCONF)**
Variabeln reinitieras vid funktionen "Sätta maskindata verksamt" via HMI, detaljprogramkommando `NEWCONF` eller NC-Reset, BAG-Reset, detaljprogramslut (M02 / M30) eller Power On.
- **PRLOC (programlokal ändring)**
Variabeln reinitieras vid NC-Reset, BAG-Reset eller detaljprogramslut (M02 / M30) endast när den har förändrats inom ramen för det aktuella detaljprogrammet.
Attributet `PRLOC` får endast användas i sammanhang med programmerbara settingdata (se följande tabell).

Tabell 2-1 Programmerbara settingdata

Nummer	Beteckning	G-kommando ¹⁾
42000	<code>\$SC_THREAD_START_ANGLE</code>	SF
42010	<code>\$SC_THREAD_RAMP_DISP</code>	DITS / DITE
42400	<code>\$SA_PUNCH_DWELLTIME</code>	PDELAYON
42800	<code>\$SA_SPIND_ASSIGN_TAB</code>	SETMS
43210	<code>\$SA_SPIND_MIN_VELO_G25</code>	G25
43220	<code>\$SA_SPIND_MAX_VELO_G26</code>	G26
43230	<code>\$SA_SPIND_MAX_VELO_LIMS</code>	LIMS
43300	<code>\$SA_ASSIGN_FEED_PER_REV_SOURCE</code>	FPRAON
43420	<code>\$SA_WORKAREA_LIMIT_PLUS</code>	G26
43430	<code>\$SA_WORKAREA_LIMIT_MINUS</code>	G25
43510	<code>\$SA_FIXED_STOP_TORQUE</code>	FXST
43520	<code>\$SA_FIXED_STOP_WINDOW</code>	FXSW
43700	<code>\$SA_OSCILL_REVERSE_POS1</code>	OSP1
43710	<code>\$SA_OSCILL_REVERSE_POS2</code>	OSP2
43720	<code>\$SA_OSCILL_DWELL_TIME1</code>	OST1
43730	<code>\$SA_OSCILL_DWELL_TIME2</code>	OST2
43740	<code>\$SA_OSCILL_VELO</code>	FA
43750	<code>\$SA_OSCILL_NUM_SPARK_CYCLES</code>	OSNSC
43760	<code>\$SA_OSCILL_END_POS</code>	OSE
43770	<code>\$SA_OSCILL_CTRL_MASK</code>	OSCTRL
43780	<code>\$SA_OSCILL_IS_ACTIVE</code>	OS
43790	<code>\$SA_OSCILL_START_POS</code>	OSB
1) med detta G-kommando aktiveras settingdatum		

Randvillkor

Initieringsvärde: Globala användarvariabler (GUD)

- För global användarvariabel (GUD) med giltighetsområdet `NC`, kan som initieringstidpunkt endast `INIPO` (Power On) föreskrivas.
- För globala användarvariabler (GUD) med giltighetsområdet `CHAN` kan som initieringstidpunkt förutom `INIPO` (Power On) också `INIRE` (Reset) eller `INICF` (NEWCONF) föreskrivas.
- Vid globala användarvariabler (GUD) med giltighetsområdet `CHAN` och initieringstidpunkt `INIRE` (Reset) eller `INICF` (NEWCONF), initieras vid NC-Reset, BAG-Reset och "Sätta maskindata verksamt" variablerna på nytt endast i de kanaler i vilka de nämnda händelserna utlöstes.

Initieringsvärde: Datatyp FRAME

För variabler av datatyp `FRAME` får inget initieringsvärde anges. Variabler av datatyp `FRAME` initieras implicit alltid med defaultramen.

Initieringsvärde: Datatyp CHAR

För variabler av datatyp `CHAR` kan i stället för ASCII-Codes (0...255) också det motsvarande ASCII-tecknet inom anföringstecken programmeras, t.ex. "A"

Initieringsvärde: Datatyp STRING

Vid variabler av datatyp `STRING` måste teckenföljden sättas inom anföringstecken t.ex.: ...="MASCHINE_1"

Initieringsvärde: Datatyp AXIS

För variabler av datatyp `AXIS` måste vid utvidgat skrivsätt för adresser axelbeteckningarna sättas inom parentes, t.ex.: ...=(X3)

Initieringsvärde: Systemvariabel

För systemvariabel kan genom redefinition inga användarspecifika initieringsvärden föreskrivas. Initieringsvärdena för systemvariablerna är fast föreskrivna av systemet. Genom redefinition kan dock tidpunkten (`INIRE`, `INICF`) vid vilken systemvariabeln reinitieras ändras.

Implicit initieringsvärde: Datatyp AXIS

För variabler av datatyp `AXIS` används följande implicita initieringsvärde:

- Systemdata "första geometriaxel"
- Synkronaktions-GUD (beteckning: SYG_A*), PUD, LUD:
Axelbeteckning från maskindatum: MD20082 \$MC_AXCONF_CHANAX_DEFAULT_NAME

Implicit initieringsvärde: Verktys- och magasindata

För verktygs- och magasindata kan initieringsvärden föreskrivas via följande maskindatum:
MD17520 \$MN_TOOL_DEFAULT_DATA_MASK

Märk

Synkronisering

För synkroniseringen av händelser som utlöser en reinitiering av en global variabel med läsningen av denna variabel på annat ställe ligger ansvaret uteslutande hos användaren / maskintillverkaren.

Se även

Variabler (Sida 21)

2.1.7 Attribut: Gränsvärden (LLI, ULI)

Ett övre och undre gränsvärde för definitionsområdet kan föreskrivas endast för följande datatyper:

- INT
- REAL
- CHAR

Definition (DEF) av användarvariabler: Gränsvärden och implicita initieringsvärden

Definieras vid definitionen av en användarvariabel av en av de ovan nämnda datatyperna inget explicit initieringsvärde ställs variabeln på det implicita initieringsvärdet för datatypen:

- INT: 0
- REAL: 0.0
- CHAR: 0

Ligger det implicita initieringsvärdet utanför det genom de programmerade gränsvärdena fastlagda definitionsområdet initieras variabeln med det gränsvärde som ligger närmast det implicita initieringsvärdet:

- implicit initieringsvärde < undre gränsvärde (LLI) $\Rightarrow$
initieringsvärde = undre gränsvärde
- implicit initieringsvärde > övre gränsvärde (ULI) $\Rightarrow$
initieringsvärde = övre gränsvärde

Exempel:

Programkod	Kommentar
DEF REAL GUD1	; undre gränsvärde = definitionsområdesgräns ; övre gränsvärde = definitionsområdesgräns ; inget initieringsvärde programmerat ; => implicit initieringsvärde = 0.0
DEF REAL LLI 5.0 GUD2	; undre gränsvärde = 5.0 ; övre gränsvärde = definitionsområdesgräns ; => initieringsvärde = 5.0
DEF REAL ULI -5 GUD3	; undre gränsvärde = definitionsområdesgräns ; övre gränsvärde = -5.0 ; => initieringsvärde = -5.0

Redefinition (REDEF) av användarvariabler: Gränsvärden och aktuella ärvärden

Ändras vid redefinitionen av gränsvärdena en användarvariabel så att det aktuella ärvärdet ligger utanför det nya definitionsområdet kommer ett larm och gränsvärdena övertas inte.

Märk

Redefinition (REDEF) av användarvariabler

Vid redefinitionen av gränsvärdena för en användarvariabel ska ges akt på en konsistent ändring av följande värden:

- Gränsvärden
- Ärvärde
- Initieringsvärde vid redefinieringen och vid den automatiska reinitieringen på grund av INIPO, INIRE eller INICF

Se även

Variabler (Sida 21)

2.1.8 Attribut: Fysikalisk enhet (PHU)

En fysikalisk enhet kan endast föreskrivas för variabler av följande datatyper:

- INT
- REAL

Programmerbara fysikaliska enheter (PHU)

Uppgiften för den fysikaliska enheten görs som tal med fast komma: PHU <Einheit>

Följande fysikaliska enheter kan programmeras:

<Einheit>	Betydelse	Fysikalisk enhet
0	Ingen fysikalisk enhet	-
1	Linjär- eller vinkel-position ¹⁾²⁾	[mm], [inch], [grader]
2	Linjär-position ²⁾	[mm], [inch]
3	Vinkel-position	[grader]
4	Linjär- eller vinkel-hastighet ¹⁾²⁾	[mm/min], [inch/min], [varv/min]
5	Linjär-hastighet ²⁾	[mm/min]
6	Vinkel-hastighet	[varv/min]
7	Linjär- eller vinkel-acceleration ¹⁾²⁾	[m/s ²], [inch/s ²], [varv/s ²]
8	Linjär-acceleration ²⁾	[m/s ²], [inch/s ²]
9	Vinkel-acceleration	[varv/s ²]
10	Linjär- eller vinkel-jerk ¹⁾²⁾	[m/s ³], [inch/s ³], [varv/s ³]
11	Linjär-jerk ²⁾	[m/s ³], [inch/s ³]
12	Vinkel-jerk	[varv/s ³]
13	Tid	[s]
14	Lägesreglerare-förstärkning	[16.667/s]
15	Varvmätning ²⁾	[mm/varv], [inch/varv]
16	Temperaturkompensation ¹⁾²⁾	[mm], [inch]
18	Kraft	[N]
19	Massa	[kg]
20	Tröghetsmoment ³⁾	[kgm ²]
21	Procent	[%]
22	Frekvens	[Hz]
23	Spänning	[V]
24	Ström	[A]
25	Temperatur	[°C]
26	Vinkel	[grader]
27	KV	[1000/min]
28	Linjär- eller vinkel-position ³⁾	[mm], [inch], [grader]
29	Skärhastighet ²⁾	[m/min], [feet/min]
30	Periferihastighet ²⁾	[m/s], [feet/s]
31	Motstånd	[Ohm]
32	Induktivitet	[mH]
33	Vridmoment ³⁾	[Nm]
34	Vridmomentkonstant ³⁾	[Nm/A]
35	Strömregulatorförstärkning	[V/A]
36	Varvtalsregulatorförstärkning ³⁾	[Nm/(rad*s)]
37	Varvtal	[varv/min]
42	Effekt	[kW]
43	Ström, svag	[µA]
46	Vridmoment, litet ³⁾	[µNm]
48	Promille	-

<Einheit>	Betydelse	Fysikalisk enhet
49	-	[Hz/s]
65	Flöde	[l/min]
66	Tryck	[bar]
67	Volym ³⁾	[cm ³]
68	Sträckförstärkning ³⁾	[mm/(V*min)]
69	Sträckförstärkning kraftregulator	[N/V]
155	Gängstigning ³⁾	[mm/varv], [inch/varv]
156	Gängstigningsändring ³⁾	[mm/varv / varv], [inch/varv / varv]

1) Den fysikaliska enheten är beroende av axeltypen: Linjär eller roterande axel

2) Måttsystem-omkoppling

G70/G71(inch/metrisk)

Efter en omkoppling av grundsystemet (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC) med G70/G71 följer vid skriv/läsåtkomst på längdbehäftade system- och användarvariabler **ingen** omräkning av värdena (ärvärde, defaultvärde och gränsvärden)

G700/G710(inch/metrisk)

Efter en omkoppling av grundsystemet (MD10240 \$MN_SCALING_SYSTEM_IS_METRIC) med G700/G710 följer vid skriv/läsåtkomst på längdbehäftade system- och användarvariabler **en** omräkning av värdena (ärvärde, defaultvärde och gränsvärden)

3) Variabeln räknas **inte** automatiskt om till det aktuella måttsystemet för NC (inch/metrisk). Omräkningen är uteslutande användarens / maskintillverkarens ansvar.

Märk

Planspill genom formatomräkning

Det interna utmatningsformatet för alla användarvariabler (GUD / PUD / LUD) med längdbehäftade fysikaliska enheter är metriskt. En excessiv användning av sådana variabler i huvudloppet för NCK, t. ex. i synkronaktioner, kan vid en omkoppling av måttsystem leda till ett räknetidsspill i interpolatorplanet, larm 4240.

Märk

Kompatibilitet för enheter

Vid användning av variabler (tillordning, jämförelse, beräkning etc.) görs ingen kontroll av kompatibilitet för de deltagande enheterna. En eventuellt erforderlig omräkning är uteslutande användarens / maskintillverkarens ansvar.

Se även

Variabler (Sida 21)

2.1.9 Attribut: Åtkomsträtter (APR, APW, APRP, APWP, APRB, APWB)

Beteckning

Beteckningen för åtkomstattributet AP... är sammansatt av:

1. A: Access
2. P: Protection
3. R / W: Read / Write
4. P / B: Program / BTSS

Åtkomsträtter / skyddsnivåer

Åtkomsträtterna motsvarar följande skyddsnivåer som ska anges vid programmeringen:

Åtkomsträtt	Skyddsnivå
Lösenord system	0
Lösenord maskintillverkare	1
Lösenord service	2
Lösenord slutanvändare	3
Nyckelbrytare läge 3	4
Nyckelbrytare läge 2	5
Nyckelbrytare läge 1	6
Nyckelbrytare läge 0	7

Definition (DEF) av användardata

Åtkomsträtter (APR... / APW...) kan definieras för följande data:

- Globala användardata (GUD)

Redefinition (REDEF) av system- och användardata

Åtkomsträtter (APR... / APW...) kan redefinieras för följande data:

- Systemdata

- Maskindata

Märk

Redefinition av läsrättigheterna för maskindata

Skyddsnivån för läsning av maskindata kan endast sättas gemensamt med nyckelordet APR för detaljprogram och BTSS.

Nyckelorden APRP och APRB stöds inte vid redefinitionen av läsrättigheterna och leder till meddelande av larm 12490 "Åtkomsträtt APRP/APRB <Schutzstufe> ställdes inte in".

- Settingdata

- Systemvariabel

- Processdata

- Magasindata

- Verktygsdata

- Användardata

- R-parametrar

- Synkronaktionsvariabel (\$AC_MARKER, \$AC_PARAM, \$AC_TIMER)

- Synkronaktions-GUD (SYG_xy[], med x=R, I, B, A, C, S och y=S, M, U, 4, ..., 9)

- EPS-parametrar

- Verktygsdata-OEM

- Magasindata-OEM

- Globala användarvariabler (GUD)

Märk

Vid redefinitionen kan åtkomsträtten för en variabel frigivas mellan den lägsta skyddsnivån 7 och den egna skyddsnivån t. ex. 1 (maskintillverkare).

Redefinition (REDEF) av NC-språkkommandon

Åtkomst- resp. verkställanderätten (APX) kan redefinieras för följande NC-språkkommandon:

- G-kommandon / (vägvillkor)

Litteratur:

Programmeringshandbok Grunder; Kapitel: G-kommandon / (vägvillkor)

- Fördefinierade funktioner

Litteratur:

Programmeringshandbok Grunder; Kapitel: Fördefinierade funktioner

- Fördefinierade underprogramanrop
Litteratur:
Programmeringshandbok Grunder; Kapitel: Fördefinierade underprogramanrop
- Anvisning DO vid synkronaktioner
- Programbeteckning för cykler
Cykeln måste finnas lagrad i en cykelkatalog och innehålla en PROC-anvisning.

Åtkomsträtter beträffande NC-program och cykler (APRP, APWP)

De olika åtkomsträtterna har följande inverkan på åtkomsten från ett NC-program resp. cykel:

- APRP 0 / APWP 0
 - Vid genomarbetning av NC-programmet måste lösenordet för systemet vara inställt
 - Cykeln måste vara lagrad i katalogen _N_CST_DIR (system)
 - För katalogen _N_CST_DIR måste verkställanderätten för systemet vara inställd i MD11160 \$MN_ACCESS_EXEC_CST
- APRP 1 / APWP 1 resp. APRP 2 / APWP 2
 - Vid genomarbetning av NC-programmet måste lösenordet för maskintillverkaren resp. servicen vara inställt
 - Cykeln måste vara lagrad i katalogen _N_CMA_DIR (maskintillverkare) eller _N_CST_DIR
 - För katalogerna _N_CMA_DIR resp. _N_CST_DIR måste verkställanderätten vara inställd på minst maskintillverkaren i maskindata MD11161 \$MN_ACCESS_EXEC_CMA resp. MD11160 \$MN_ACCESS_EXEC_CST
- APRP 3 / APWP 3
 - Vid genomarbetning av NC-programmet måste lösenordet för slutanvändaren vara inställt
 - Cykeln måste vara lagrad i katalogen _N_CUS_DIR (användare), _N_CMA_DIR eller _N_CST_DIR
 - För katalogerna _N_CUS_DIR, _N_CMA_DIR resp. _N_CST_DIR måste verkställanderätten vara inställd på minst slutanvändaren i maskindata MD11162 \$MN_ACCESS_EXEC_CUS, MD11161 \$MN_ACCESS_EXEC_CMA resp. MD11160 \$MN_ACCESS_EXEC_CST
- APRP 4...7 / APWP 4...7
 - Vid genomarbetning av NC-programmet måste nyckelbrytarläge 3 ... 0 vara inställt
 - Cykeln måste vara lagrad i katalogen _N_CUS_DIR, _N_CMA_DIR eller _N_CST_DIR
 - För katalogerna _N_CUS_DIR, _N_CMA_DIR resp. _N_CST_DIR måste verkställanderätten vara inställd på minst det motsvarande nyckelbrytarläget i maskindata MD11162 \$MN_ACCESS_EXEC_CUS, MD11161 \$MN_ACCESS_EXEC_CMA resp. MD11160 \$MN_ACCESS_EXEC_CST

Åtkomsträtter beträffande BTSS (APRB, APWB)

Åtkomsträtterna (APRB, APWB) begränsar åtkomstern till system- och användarvariabler via BTSS för alla systemkomponenter (HMI, PLC, extern dator, EPS-tjänster, etc.) på samma sätt.

Märk

HMI-lokala åtkomsträtter

Vid ändringar av åtkomsträtter till systemdata måste ges akt på att dessa görs konsistent till de via HMI-mekanism fastlagda åtkomsträtterna.

Åtkomstattribut APR / APW

Av kompatibilitetsskäl avbildas attributen APR och APW implicit på attributen APRP / APRB och APWP / APWB:

- $APR \ x \Rightarrow APRP \ x \ APRB \ x$
- $APW \ y \Rightarrow APWP \ y \ APWB \ y$

Åtkomsträtter via ACCESS-filer

Vid användningen av ACCESS-filer för tilldelningen av åtkomsträtter får redefinitioner av åtkomsträtter för systemdata, användardata och NC-språkkommandon uteslutande göras i ACCESS-filer. Globala användardata (GUD) utgör ett undantag. För dessa måste en redefinition av åtkomsträtter fortfarande göras i de motsvarande definitionsfilerna *_DEF.

För ett genomgående åtkomstskydd måste maskindata anpassas konsistent för verkställanderätterna och åtkomstskyddet av de tillhörande katalogerna.

Därur resulterar följande principiella tillvägagångssätt:

1. Upprättande av de nödvändiga definitionsfilerna:
 - _N_DEF_DIR/_N_SACCESS_DEF
 - _N_DEF_DIR/_N_MACCESS_DEF
 - _N_DEF_DIR/_N_UACCESS_DEF
2. Parametrering av skrivrätten för definitionsfilerna på det för redefinitionen nödvändiga värdet:
 - MD11170 \$MN_ACCESS_WRITE_SACCESS = <Schutzstufe>
 - MD11171 \$MN_ACCESS_WRITE_MACCESS = <Schutzstufe>
 - MD11172 \$MN_ACCESS_WRITE_UACCESS = <Schutzstufe>

2.1 Variabler

3. För åtkomst till skyddade element från cyklerna måste verkställande- och skrivrätter anpassas till cykelkatalogerna _N_CST_DIR, _N_CMA_DIR och _N_CST_DIR:
Verkställanderätter

- MD11160 \$MN_ACCESS_EXEC_CST = <Schutzstufe>
- MD11161 \$MN_ACCESS_EXEC_CMA = <Schutzstufe>
- MD11162 \$MN_ACCESS_EXEC_CUS = <Schutzstufe>

Skriv rättigheter

- MD11165 \$MN_ACCESS_WRITE_CST = <Schutzstufe>
- MD11166 \$MN_ACCESS_WRITE_CMA = <Schutzstufe>
- MD11167 MN_ACCESS_WRITE_CUS = <Schutzstufe>

Verkställanderätten måste minst ställas in på samma skyddsnivå som den högsta skyddsnivån för det använda elementet.

Skrivrätten måste minst ställas in på samma skyddsnivå som verkställanderätten.

4. Skrivrätterna till de HMI-lokala cykelkatalogerna måste ställas in på samma skyddsnivå som den för de NC-lokala cykelkatalogerna.

Litteratur

Användarhandbok

Underprogramanrop i ACCESS-filer

För den ytterligare struktureringen av åtkomstskyddet kan också underprogram (identifikation SPF eller MPF) anropas i ACCESS-filerna. Underprogrammen ärver därvid verkställanderätterna från den ACCESS-fil som anropar.

Märk

I ACCESS-filerna kan endast åtkomsträtterna redefinieras. Alla andra attribut måste fortfarande programmeras resp. redefinieras i de motsvarande definitionsfilerna.

Se även

Variabler (Sida 21)

2.1.10 Attribut: Dataklass (DCM, DCI, DCU) - endast SINUMERIK 828D

För att förenkla datahanteringen inom ramen för idrifttagande, serieidrifttagande och upprustning av maskiner och maskinsatser är alla system- och användardata för NC uppdelade i dataklasser.

Dataklass	Data
S = System	Av Siemens iordningställda systemdata som maskin- och settingdata, standard- och mätcykler, definitioner (SGUD) och makros (SMAC) etc.
M = Manufacturer (maskintillverkare)	Maskinseriespecifika idrifttagandedata som tillverkarcyklar, definitioner (MGUD) och makros (MMAC) och maskindata, som bestämmer omfattningen av funktioner hos MASKINEN.
I = Individual (maskinspecifik)	Maskinspecifika idrifttagandedata som kompensationsdata och referenspunktsförflyttningar.
U = User (användare)	Maskinspecifika data, som skapas under drift av MASKINEN som verktygsdata, settingdata, detaljprogram, användarcyklar, definitioner (UGUD) och makros (UMAC).

Litteratur:

SINUMERIK 828D idrifttagningshandbok Svarva och fräsa; Kap.: "Inledning och användning av dataklasser"

Definition (DEF) av användardata:

Med dataklassen för den fil eller den pärm, i vilken ett användardatum definieras, är dataklassen för datumet implicit fastlagt. Dataklassen för datumet kan inte ändras.

Vid definitionen (DEF) av användardatumet kan dock för **datavärdet** en egen, av dataklassen för datumet avvikande dataklass, angivas.

För dataklassen för datavärdet måste gälla:

Prioriteten för dataklassen för datavärdet $\leq$ prioriteten för dataklassen för datumet

Exempel:

Definitionen av GUD, som beskriver en mätprobe, bör befinna sig i dataklass M (= Manufacturer), eftersom det är nödvändigt för förloppet av tillverkarcyklerna. Men värdet för datumet bör, eftersom typen av mätprobe kan vara olika från MASKIN till MASKIN, tillhöra dataklassen I (= Individual).

MGUD.DEF (dataklass M)

```
...
DEF CHAN DCI INT CALIPER
...
```

Redefinition (REDEF) av systemdata

Dataklassen för systemdata kan ändras med redefinition (REDEF). Redefinitionen måste göras i en definitionsfil med dataklass S eller M.

Vid användning av ACCESS-filer får redefinitioner endast göras inom ACCESS-filerna.

Respektive dataklass för maskin-, setting- och optionsdata samt systemvariabler finns i:

- Listhandbok Utförlig maskindatabeskrivning, parametrar: "Klass"
- Listhandbok Systemvariabler

2.1.11 Översikt över definierbara och redefinierbara attribut

De följande tabellerna visar för vilka datatyper vilka attribut kan definieras (DEF) och/eller redefinieras (REDEF).

Systemdata

Datatyp	Init.värde	Gränsvärden	Fys. enhet	Åtkomsträtter	Dataklass (endast 828D)
Maskindata	---	---	---	REDEF	REDEF
Settingdata	REDEF	---	---	REDEF	---
FRAME-data	---	---	---	REDEF	---
Processdata	---	---	---	REDEF	---
Stigning kulmutterskruv felkomp. (EEC)	---	---	---	REDEF	---
Överhängskompensation (CEC)	---	---	---	REDEF	---
Kvadrantfelkompensation (QEC)	---	---	---	REDEF	---
Magasindata	---	---	---	REDEF	---
Verktygsgdata	---	---	---	REDEF	---
Skyddsområden	---	---	---	REDEF	---
Orienterbara verktygsbärare	---	---	---	REDEF	---
Kinematiska följder	---	---	---	REDEF	---
3D-skyddsområden	---	---	---	REDEF	---
Arbetsfältsbegränsning	---	---	---	REDEF	---

Användardata

Datatyp	Init.värde	Gränsvärden	Fys. enhet	Åtkomsträtter	Dataklass
R-parametrar	REDEF	REDEF	REDEF	REDEF	---
Synkronaktionsvariabel (\$AC_...)	REDEF	REDEF	REDEF	REDEF	---
Synkronaktions-GUD (SYG_...)	REDEF	REDEF	REDEF	REDEF	---
EPS-parametrar	REDEF	REDEF	REDEF	REDEF	---
Verktygsgdata-OEM	REDEF	REDEF	REDEF	REDEF	---
Magasindata-OEM	REDEF	REDEF	REDEF	REDEF	---
Globala användarvariabler (GUD)	DEF / REDEF	DEF	DEF	DEF / REDEF	DEF / REDEF
Lokala användarvariabler (PUD / LUD)	DEF	DEF	DEF	---	---

Se även

Variabler (Sida 21)

2.1.12 Definition och initiering av fältvariabler (DEF, SET, REP)

En användarvariabel kan definieras som 1- till maximalt 3-dimensionellt fält (array):

- 1-dimensionellt: DEF <Datentyp> <Variablenname>[<n>]
- 2-dimensionellt: DEF <Datentyp> <Variablenname>[<n>,<m>]
- 3-dimensionellt: DEF <Datentyp> <Variablenname>[<n>,<m>,<o>]

Märk

Användarvariabel av datatyp STRING kan definieras maximalt som 2-dimensionellt fält.

Datatyper

Användarvariabel kan definieras som fält för följande datatyper: BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME

Värdetillordning till fältelement

Värdetillordning till fältelement kan göras vid följande tidpunkter:

- vid fältdefinitionen (initieringsvärden)
- under programförloppet

Värdetillordning därvid göras via:

- explicit angivande av ett fältelement
- explicit angivande av ett fältelement som startelement och angivande av en värdelista (SET)
- explicit angivande av ett fältelement som startelement och angivande av ett värde och frekvensen för dess upprepande (REP)

Märk

Användarvariabler av datatyp FRAME kan inte tillordnas några initieringsvärden.

Syntax (DEF)

```
DEF <Datentyp> <Variablenname>[<n>,<m>,<o>]
DEF  STRING[<Stringlängde>] <Variablenname>[<n>,<m>]
```

Syntax (DEF...=SET...)

Användning av en värdelista:

- vid definitionen:

```
DEF <Datentyp> <Variablenname> [<n>, <m>, <o>] =
SET(<Wert1>, <Wert2>, ...)
```

likabetydande med:

```
DEF <Datentyp> <Variablenname> [<n>, <m>, <o>] =
(<Wert1>, <Wert2>, ...)
```

Märk

Vid initieringen via en värdelista är angivande av SET optional.

- vid en värdetillordning:

```
<Variablenname> [<n>, <m>, <o>] = SET(<WERT1>, <Wert2>, ...)
```

Syntax (DEF...=REP...)

Användning av ett värde med upprepning

- vid definitionen:

```
DEF <Datentyp> <Variablenname> [<n>, <m>, <o>] = REP(<Wert>)
DEF <Datentyp> <Variablenname> [<n>, <m>, <o>] =
REP(<Wert>, <Anzahl_Feldelemente>)
```
- vid en värdetillordning:

```
<Variablenname> [<n>, <m>, <o>] = REP(<Wert>)
<Variablenname> [<n>, <m>, <o>] = REP(<Wert>, <Anzahl_Feldelemente>)
```

Betydelse

DEF:	Kommando för definition av variabler	
<Datentyp>:	Datatyp för variabler	
	Värdeområde:	
	• vid systemvariabler: BOOL, CHAR, INT, REAL, STRING, AXIS • vid GUD- eller LUD-variabler: BOOL, CHAR, INT, REAL, STRING, AXIS, FRAME	
<Stringlängde>:	Maximalt antal tecken vid datatyp STRING	
<Variablenname>:	Variabelnamn	
[<n>, <m>, <o>]:	Fältstorlekar resp. fältindex	
<n>:	Fältstorlek resp. fältindex för 1:a dimensionen	
	Typ:	INT (vid systemvariabler också AXIS)
	Värdeområde:	Max. fältstorlek 65535 Fältindex: $0 \leq n \leq 65534$
<m>:	Fältstorlek resp. fältindex för 2:a dimensionen	
	Typ:	INT (vid systemvariabler också AXIS)
	Värdeområde:	Max. fältstorlek 65535 Fältindex: $0 \leq m \leq 65534$

<o>:	Fältstorlek resp. fältindex för 3:e dimensionen		
	Typ:	INT (vid systemvariabler också AXIS)	
	Värdeområde:	Max. fältstorlek 65535 Fältindex: $0 \leq o \leq 65534$	
SET:	Värdetillordning via den angivna värdelistan		
(<Wert1>,<Wert2>,...):	Värdelista		
REP:	Värdetillordning via det angivna <Wert>		
<Wert>:	Värde vid vilket fältelementen ska beskrivas vid initieringen med REP.		
<Anzahl_Feldelemente>:	Antal fältelement som ska beskrivas med det angivna <Wert>. För de resterande fältelementen gäller oberoende av tidpunkten: - Initiering vid fältdefinitionen: → De resterande fältelementen beskrivs med noll - Tillordning under programkörningen: → De aktuella värdena för fältelementen förblir oförändrade. Är parametern inte programmerad beskriva alla fältelementen med <Wert>. Är parametern lika med noll gäller beroende av tidpunkten: - Initiering vid fältdefinitionen: → Alla element förbeläggs med noll - Tillordning under programkörningen: → De aktuella värdena för fältelementen förblir oförändrade.		

Fältindex

Den implicita ordningsföljden för fältelementen t.ex. vid en värdetillordning via SET eller REP sker genom upprepning av fältindex från höger till vänster.

Exempel: Initiering av ett 3-dimensionellt fält med 24 fältelement:

```

DEF INT FELD[2,3,4] = REP(1,24)
  FELD[0,0,0] = 1      1:a fältelement
  FELD[0,0,1] = 1      2:a fältelement
  FELD[0,0,2] = 1      3:e fältelement
  FELD[0,0,3] = 1      4:e fältelement
  ...
  FELD[0,1,0] = 1      5:e fältelement
  FELD[0,1,1] = 1      6:e fältelement
  ...
  FELD[0,2,3] = 1      12:e fältelement
  FELD[1,0,0] = 1      13:e fältelement
  FELD[1,0,1] = 1      14:e fältelement
  ...
  FELD[1,2,3] = 1      24:e fältelement

```

motsvarande:

```
FOR n=0 TO 1
  FOR m=0 TO 2
    FOR o=0 TO 3
      FELD[n,m,o] = 1
    ENDFOR
  ENDFOR
ENDFOR
```

Exempel: Initiering av kompletta variabelfält

Aktuell beläggning se bild.

Programkod

```
N10 DEF REAL FELD1[10,3]=SET(0,0,0,10,11,12,20,20,20,30,30,30,40,40,40,)
N20 FELD1[0,0]=REP(100)
N30 FELD1[5,0]=REP(-100)
N40 FELD1[0,0]=SET(0,1,2,-10,-11,-12,-20,-20,-20,-30, , , -40,-40,-50,-60,-70)
N50 FELD1[8,1]=SET(8.1,8.2,9.0,9.1,9.2)
```

Fältindex										2
[1,2]		N10: Initiering vid definition			N20/N30: Initiering med identiskt värde			N40/N50: Initiering med olika värden		
		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
		Fältelementen [5,0] till [9,2] initierades med defaultvärdet (0.0).						Fältelementen [3,1] till [4,0] initierades med defaultvärdet (0.0). Fältelementen [6,0] till [8,0] förändrades inte.		

Se även

Definition och initiering av fältvariabler (DEF, SET, REP): Ytterligare informationer (Sida 55)
 Variabler (Sida 21)

2.1.13 Definition och initiering av fältvariabler (DEF, SET, REP): Ytterligare informationer**Ytterligare informationer (SET)****Initiering vid definitionen**

- Det blir, med början med 1:a fältelementet, så många fältelement med värdena från värdelistan initierade som det finns element programmerade i värdelistan.
- Fältelement utan explicit angivna värden i värdelistan (luckor i värdelistan) beläggs med 0.
- Vid variabler av datatyp AXIS är luckor i värdelistan inte tillåtna.
- Innehåller värdelistan fler värden än det finns fältelement definierade indikeras ett larm.

Värdetillordning under programkörningen

Vid värdetillordningen under programkörningen gäller de ovan vid definitionen beskrivna reglerna. Dessutom finns det följande möjligheter:

- Som element i värdelistan är också uttryck tillåtna.
- Värdetillordningen börjar vid det programmerade fältindexet. Härigenom låter sig delfält målinriktat beläggas med värden.

Exempel:

Programkod	Kommentar
DEF INT FELD[5,5]	; Fältdefinition
FELD[0,0]=SET(1,2,3,4,5)	; Värdetillordning till de första 5 fältelementen [0,0] - [0,4]
FELD[0,0]=SET(1,2, , ,5)	; Värdetillordning med lucka till de första 5 fältelementen [0,0] - [0,4], fältelement [0,2] och [0,3] = 0
FELD[2,3]=SET(VARIABLE,4*5.6)	; Värdetillordning med variabel och uttryck från fältindex [2,3]: [2,3] = VARIABLE [2,4] = 4 * 5.6 = 22.4

Ytterligare informationer (REP)**Initiering vid definitionen**

- Alla eller det optional angivna antalet fältelement initieras med det angivna värdet (konstant).
- Variabler av datatyp FRAME kan inte initieras.

Exempel:

Programkod	Kommentar
DEF REAL varName[10]=REP(3.5,4)	; Initiera fältdefinition och fältelement [0] till [3] med värdet 3,5

Värdetillordning under programkörningen

Vid värdetillordningen under programkörningen gäller de ovan vid definitionen beskrivna reglerna. Dessutom finns det följande möjligheter:

- Som element i värdelistan är också uttryck tillåtna.
- Värdetillordningen börjar vid det programmerade fältindexet. Härigenom låter sig delfält målinriktat beläggas med värden.

Exempel:

Programkod	Kommentar
DEF REAL varName[10]	; Fältdefinition
varName[5]=REP(4.5,3)	; Fältelement [5] till [7] = 4,5
R10=REP(2.4,3)	; R-parametrar R10 till R12 = 2,4
DEF FRAME FRM[10]	; Fältdefinition
FRM[5]=REP(CTrans(X,5))	; Fältelement [5] till [9] = CTRANS(X,5)

Se även

Definition och initiering av fältvariabler (DEF, SET, REP) (Sida 51)

2.1.14 Datatyper

Följande datatyper står i NC till förfogande:

Datatype	Betydelse	Värdeområde
INT	Heltaligt värde med förtecken	-2147483648 ... +2147483647
REAL	Real-tal (LONG REAL efter IEEE)	$\pm(\sim 2,2 \cdot 10^{-308} \dots \sim 1,8 \cdot 10^{+308})$
BOOL	Sanningsvärde TRUE (1) och FALSE (0)	1, 0
CHAR	ASCII-tecken	ASCII-Code 0 ... 255
STRING	Teckenföljd definierad längd	Maximalt 200 tecken (inga specialtecken)
AXIS	Axel-/spindelbeteckning	Kanalaxelbeteckning
FRAME	Geometriska uppgifter för en statisk koordinattransformation (förflyttning, rotation, skaländring, spegling)	---

Implicita datatypomvandlingar

Följande datatypomvandlingar är möjliga och blir vid tillordningar och överlämnande av parametrar implicit gjorda:

från ↓ / till →	REAL	INT	BOOL
REAL	x	o	&
INT	x	x	&
BOOL	x	x	x
x: utan begränsningar möjligt o: Dataförlust genom överskridande av värdeområdet möjligt ⇒ larm; Avrundning: Värde efter komma ≥ 0,5 ⇒ avrunda uppåt, värde efter komma < 0,5 ⇒ avrunda &: Värde ≠ 0 ⇒ TRUE, värde == 0 ⇒ FALSE			

Se även

Variabler (Sida 21)

2.1.15 Kontrollera om en variabel finns (ISVAR)

Med den fördefinierade funktionen ISVAR kan kontrolleras, om en system-/användarvariabel (t. ex. maskindatum, settingdatum, systemvariabel, allmän variabel som GUD) är känd i NC.

Variabel

Den variabel som ska avföras uppbyggd på följande sätt:

Dimensionslös variabel: <Variable>
 Endimensionell variabel utan fältindex: <Variable>[]
 Endimensionell variabel med fältindex n: <Variable>[<n>]
 Tvådimensionell variabel utan fältindex: <Variable>[,]
 Tvådimensionell variabel med fältindex n och m: <Variable>[<n>,<m>]

Syntax

<Ergebnis>=ISVAR(<Variable>[<n>,<m>])

Betydelse

<Ergebnis>:	Returvärde		
	Datotyp:	BOOL	
	Värdeområde:	1	Variabel finns
		0	Variabel okänd
ISVAR:	Kontrollerar om den angivna system-/användarvariabeln är känd i NC.		
<Variable>:	Namn för system- /användarvariabler		
	Datotyp:	STRING	

2.1 Variabler

<n>:	Fältindex för den första dimensionen (option)	
	Datatyp:	INT
<m>:	Fältindex för den andra dimensionen (option)	
	Datatyp:	INT

I enlighet med överföringsparametern utförs följande kontroller:

- Är namnet känt
- Är variablen ett fält
- Är det ett en- eller tvådimensionellt fält
- Är respektive fältindex i det tillåtna området

När alla kontroller är positiva, returneras TRUE (1).

Är en kontroll negativ eller har ett syntaxfel uppstått, returneras FALSE (0).

Exempel

Programkod	Kommentar
DEF INT VAR1	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VAR1")	; IS_VAR är i detta fall TRUE.

Programkod	Kommentar
DEF REAL VARARRAY[10,10]	
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("VARARRAY[,]")	; IS_VAR är i detta fall TRUE, är ett tvådimensionellt fält.
N20 IS_VAR=ISVAR("VARARRAY")	; IS_VAR är TRUE, variabel existerar.
N30 IS_VAR=ISVAR("VARARRAY[8,11]")	; IS_VAR är FALSE, fältindex är inte tillåtet.
N40 IS_VAR=ISVAR("VARARRAY[8,8]")	; IS_VAR ist FALSE, "]" saknas (syntaxfel).
N50 IS_VAR=ISVAR("VARARRAY[,8]")	; IS_VAR är TRUE, fältindex är tillåtet.
N60 IS_VAR=ISVAR("VARARRAY[8,]")	; IS_VAR är TRUE, fältindex är tillåtet.

Programkod	Kommentar
DEF BOOL IS_VAR=FALSE	
N100 IS_VAR=ISVAR("\$MC_GCODE_RESET_VALUES[1]")	; Överföringsparameter är ett maskindatum, IS_VAR är TRUE.

Programkod	Kommentar
DEF BOOL IS_VAR=FALSE	
N10 IS_VAR=ISVAR("\$P_EP")	; IS_VAR är i detta fall TRUE.
N20 IS_VAR=ISVAR("\$P_EP[X]")	; IS_VAR är i detta fall TRUE.

2.1.16 Läs attributvärden/datatyp (GETVARPHU, GETVARAP, GETVARLIM, GETVARDIM, GETVARDFT, GETVARTYP)

Med de fördefinierade funktionerna GETVARPHU, GETVARAP, GETVARLIM och GETVARDFT kan attributvärdena för system-/användarvariabler läsas, med GETVARTYP datatypen för en system-/användarvariabel.

Läs fysikalisk enhet

Syntax:

<Ergebnis>=GETVARPHU (<Name>)

Betydelse:

<Ergebnis>:	Siffervärde för den fysikaliska enheten		
	Datatyp:	INT	
	Värdeområde:	se tabell i "Attribut: Fysikalisk enhet (PHU) (Sida 41)"	
		I händelse av fel:	
		- 2	Det angivna <Name> är inte tillordnat till någon system-parameter eller användarvariabel.
GETVARPHU:	Läsa den fysikaliska enheten för en system- /användarvariabel		
<Namn>:	Namn för system- /användarvariabel		
	Datatyp:	STRING	

Exempel:

NC innehåller följande GUD-variabel:

```
DEF CHAN REAL PHU 42 LLI 0 ULI 10000 electric
```

Programkod	Kommentar
DEF INT result=0	
result=GETVARPHU("elec- tric")	; Fastställer den fysikaliska enheten för GUD-variabeln.
IF (result < 0) GOTO error	

Som resultat returneras värdet 42. Detta motsvarar den fysikaliska enheten [kW].

Märk

Med GETVARPHU låter sig t. ex. kontrolleras om vid en variabeltillordning a = b båda variablerna har de väntade fysikaliska enheterna.

Läs åtkomsträtt

Syntax:

<Ergebnis>=GETVARAP (<Name>, <Zugriff>)

Betydelse:

<Ergebnis>:	Skyddsnivå för den angivna <Zugriff>		
	Datatyp:	INT	
	Värdeområde:	0 ... 7	Se "Attribut: Åtkomsträtter (APR, APW, APRP, APWP, APRB, APWB) (Sida 44)".
		I händelse av fel:	
		- 1	kan inte skrivas (endast vid <Zugriff> "WP" och "WB" relevant)
		- 2	Det angivna <Name> är inte tillordnat till någon systemparameter eller användarvariabel.
	- 3	felaktigt värde för <Zugriff>	
GETVARAP:	Läsa åtkomsträtten för en system- /användarvariabel		
<Namn>:	Namn för system- /användarvariabel		
	Datatyp:	STRING	
<Zugriff>:	Typ av åtkomst		
	Datatyp:	STRING	
	Värdeområde:	"RP"	Läsa via detaljprogram
		"WP"	Skriva via detaljprogram
		"RB"	Läsa via BTSS
		"WB"	Skriva via BTSS

Exempel:

Programkod	Kommentar
DEF INT result=0	
result=GETVA-	; Fastställer åtkomstskyddet för systemparametern "Ma-
RAP("\$TC_MAP8","WB")	gasinposition" beträffande skriva via BTSS.
IF (result < 0) GOTO error	

Som resultat returneras värdet 7. Detta motsvarar nyckelbrytarläge 0 (= inget åtkomstskydd).

Märk

Med GETVARAP kan t. ex. ett kontrollprogram realiseras som kontrollerar den av applikationen väntade åtkomsträtten.

Läsa gränsvärden**Syntax:**

```
<Status>=GETVARLIM(<Name>,<Grenzwert>,<Ergebnis>)
```


Betydelse:

<Status>:	Funktionsstatus		
	Datatyp:	INT	
	Värdeområde:	1	o. k.
		-1	inget gränsvärde definierat (vid variabler av typ AXIS, STRING, FRAME)
		-2	Det angivna <Name> är inte tillordnat till någon systemparameter eller användarvariabel.
-3		felaktigt värde för <Grenzwert>	
GETVARLIM:	Läsa det undre/övre gränsvärdet för en system- /användarvariabel		
<Namn>:	Namn för system- /användarvariabel		
	Datatyp:	STRING	
<Grenzwert>:	Anger vilket gränsvärde som ska läsas ut		
	Datatyp:	CHAR	
	Värdeområde:	"L"	= undre gränsvärde
		:	
		"U"	= övre gränsvärde
:			
<Ergebnis>:	Retur av gränsvärdet		
	Datatyp:	VAR REAL	

Exempel:

Programkod	Kommentar
DEF INT state=0	
DEF REAL result=0	
state=GETVAR-	; Fastställer det undre gränsvärdet för
LIM("\$MA_MAX_AX_VELO", "L", result)	MD32000 \$MA_MAX_AX_VELO.
IF (result < 0) GOTO error	

Läsa attribut/datatyp**Syntax:**

<Ergebnis>=GETVARDIM(<Name>, Index)

Betydelse:

<Ergebnis>:	Dimension/antal för fältet <Index>	
	Datatyp:	INT
GETVARDIM:	Läsa det undre/övre gränsvärdet för en system- /användarvariabel	
<Name>:	Läsa antalet element för fältet	
	Datatyp:	STRING
<Index>:	Nummer på fältet, max 3.	
	Datatyp:	INT

Exempel:

Programkod	Kommentar
N5 DEF REAL myReal[5,4]	
N10 R1 = GETVATDIM(„myReal“,1)	R1 = 5
N15 R2 = GETVATDIM(„myReal“,2)	R2 = 4

Läsa standardvärde**Syntax:**

```
<Status>=GETVARDFT(<Name>,<Ergebnis>[,<Index_1>,<Index_2>,<Index_3>]
)
```

Betydelse:

<Status>:	Funktionsstatus			
	Datatyp:	INT		
	Värdeområde:	1	o. k.	
		-1	inget standardvärde tillgängligt (t. ex. eftersom <Ergebnis> har den felaktiga typen för <Name>)	
		-2	Det angivna <Name> är inte tillordnat till någon systemparameter eller användarvariabel.	
		-3	felaktigt värde för <Index_1>, dimension mindre än ett (= ingen array = skalar)	
		-4	felaktigt värde för <Index_2>	
-5	felaktigt värde för <Index_3>			
GETVARDFT:	Läsa standardvärdet för en system- /användarvariabel			
<Namn>:	Namn för system- /användarvariabel			
	Datatyp:	STRING		
<Ergebnis>:	Retur av standardvärdet			
	Datatyp:	VAR REAL (vid läsning av standardvärdet för variabler av typerna INT, REAL, BOOL, AXIS)		
		VAR STRING (vid läsning av standardvärdet för variablerna av typerna STRING och CHAR)		
		VAR FRAME (vid läsning av standardvärdet för variabler av typen FRAME)		
<Index_1>:	Index på första dimension (option)			
	Datatyp:	INT		
	Ej programmerat betyder = 0			
<Index_2>:	Index på andra dimension (option)			
	Datatyp:	INT		
	Ej programmerat betyder = 0			

<Index_3>:	Index på tredje dimension (option)	
	Datatyp:	INT
	Ej programmerat betyder = 0	

Exempel:

Programkod	Kommentar
DEF INT state=0	
DEF REAL resultR=0	; Variabel för upptagande av standardvärdena av typerna INT, REAL, BOOL, AXIS.
DEF FRAME resultF=0	; Variabel för upptagande av standardvärdena av typen FRAME
IF (GETVARTYP("\$MA_MAX_AX_VELO") <> 4) GOTO error	
state=GETVARDF("MA_MAX_AX_VELO", resultR, AXTOINT(X))	; Fastställer standardvärdet för "X"-axeln.
IF (resultR < 0) GOTO error	
IF (GETVARTYP("\$TC_TP8") <> 3) GOTO error	
state=GETVARDF("\$TC_TP8", resultR)	
IF (GETVARTYP("\$P_UBFR") <> 7) GOTO error	
state=GETVARDF("\$P_UBFR", resultF)	

Läsa datatyp**Syntax:**

<Ergebnis>=GETVARTYP (<Name>)

Betydelse:

<Ergebnis>:	Datatyp för den angivna system- /användarvariabeln		
	Datatyp:	INT	
	Värdeområde:	1	= BOOL
		2	= CHAR
		3	= INT
		4	= REAL
		5	= STRING
		6	= AXIS
		7	= FRAME
		I händelse av fel:	
		< 0	Det angivna <Name> är inte tillordnat till någon systemparameter eller användarvariabel.
GETVARTYP:	Läsa datatypen för en system- /användarvariabel		

<Namn>:	Namn för system- /användarvariabel	
	Datatyp:	STRING

Exempel:

Programkod	Kommentar
DEF INT result=0	
DEF STRING name="R"	
result=GETVARTYP(name)	; Fastställer typen för R-parametern.
IF (result < 0) GOTO error	

Som resultat returneras värdet 4. Detta motsvarar datatypen REAL.

2.2 Indirekt programmering

2.2.1 Indirekt programmering av adresser

; Vid den indirekta programmeringen av adresser ersätts den utvidgade adressen (<index>) av en variabel av lämplig typ.

Märk

Den indirekta programmeringen av adresser är inte möjlig vid:

- N (blocknummer)
- L (underprogram)
- Inställbara adresser
(t.ex. X[1] i stället för X1 är inte tillåtet)

Syntax

<ADRESSE> [<Index>]

Betydelse

<ADRESSE> [...]:	Fast adress med utvidgning (index)
<Index>:	Variabel t.ex. för spindelnummer, axel, ...

Exempel

Exempel 1: Indirekt programmering av ett spindelnummer

Direkt programmering:

Programkod	Kommentar
S1=300	; Varvtal 300 varv/min för spindeln med nummer 1.

Indirekt programmering:

Programkod	Kommentar
DEF INT SPINU=1	; Definition av variablerna av typ INT och värdetillordning.
S[SPINU]=300	; Varvtal 300 varv/min för den spindel vars nummer är lagrat i variabeln SPINU (i detta exempel spindeln med nummer 1).

Exempel 2: Indirekt programmering av en axel

Direkt programmering:

Programkod	Kommentar
FA[U]=300	; Matning 300 för axeln "U".

Indirekt programmering:

Programkod	Kommentar
DEF AXIS AXVAR2=U	; Definition av en variabel typ AXIS och värdetillordning.
FA[AXVAR2]=300	; Matning 300 för den axel vars adressnamn är lagrat i variabeln med namnet AXVAR2.

Exempel 3: Indirekt programmering av en axel

Direkt programmering:

Programkod	Kommentar
\$AA_MM[X]	; Läs mätprobe-mätvärde (MKS) för axeln "X".

Indirekt programmering:

Programkod	Kommentar
DEF AXIS AXVAR3=X	; Definition av en variabel typ AXIS och värdetillordning.
\$AA_MM[AXVAR3]	; Läs mätprobe-mätvärde (MKS) för den axel vars namn är lagrat i variabeln AXVAR3.

Exempel 4: Indirekt programmering av en axel

Direkt programmering:

Programkod
X1=100 X2=200

Indirekt programmering:

Programkod	Kommentar
DEF AXIS AXVAR1 AXVAR2	; Definition av två variabler av typ AXIS.
AXVAR1=(X1) AXVAR2=(X2)	; Tillordning av axelnamn.
AX[AXVAR1]=100 AX[AXVAR2]=200	; Förflyttning av de axlar vars adressnamn är lagrade i variablerna med namnen AXVAR1 och AXVAR2.

Exempel 5: Indirekt programmering av en axel

Direkt programmering:

Programkod
G2 X100 I20

Indirekt programmering:

Programkod	Kommentar
DEF AXIS AXVAR1=X	; Definition av en variabel typ AXIS och värdetillordning.
G2 X100 IP[AXVAR1]=20	; Indirekt programmering av mittpunktsuppgift för den axel vars adressnamn är lagrat i variabeln med namnet AXVAR1

Exempel 6: Indirekt programmering av fältelement

Direkt programmering:

Programkod	Kommentar
DEF INT FELD1[4,5]	; Definition av fält 1.

Indirekt programmering:

Programkod	Kommentar
DEFINE DIM1 AS 4	; Vid fältdimensioner måste fältstorlekar anges som fasta värden.
DEFINE DIM2 AS 5	
DEF INT FELD[DIM1,DIM2]	
FELD[DIM1-1,DIM2-1]=5	

Exempel 7: Indirekt underprogramanrop

Programkod	Kommentar
CALL "L" << R10	; Anrop av det program vars namn står i R10 (stringlänkning).

2.2.2 Indirekt programmering av G-kommandon

Den indirekta programmeringen av G-kommandon möjliggör en effektiv cykelprogrammering.

Syntax

G[<Gruppe>]=<Nummer>

Betydelse

G[...]:	G-kommando med utvidgning (index)	
<Gruppe>:	Index-parametrar: G-grupp	
	Typ:	INT
<Nummer>:	Variabel för G-kommando-numret	
	Typ:	INT eller REAL

Märk

Som regel kan endast icke syntaxbestämmande G-kommandon programmeras indirekt.

Av de syntaxbestämmande G-kommandona är endast de från G-grupp 1 möjliga.

De syntaxbestämmande G-kommandona från G-grupperna 2, 3 och 4 är inte möjliga.

Märk

I den indirekta G-kommando-programmeringen är inga aritmetik-funktioner tillåtna. En nödvändig beräkning av G-kommando-numret måste göras i en egen detaljprogramrad före den indirekta G-kommando-programmeringen.

Exempel**Exempel 1: Inställbar nollpunktsförflyttning (G-grupp 8)**

Programkod	Kommentar
N1010 DEF INT INT_VAR	
N1020 INT_VAR=2	
...	
N1090 G[8]=INT_VAR G1 X0 Y0	; G54
N1100 INT_VAR=INT_VAR+1	; G-kommando-beräkning
N1110 G[8]=INT_VAR G1 X0 Y0	; G55

Exempel 2: Planval (G-grupp 6)

Programkod	Kommentar
N2010 R10=\$P_GG[6]	; Läs aktivt G-kommando i G-grupp 6
...	
N2090 G[6]=R10	

Litteratur

Informationer till G-grupperna se:
Programmeringshandbok Grunder; Kapitel "G-grupper"

2.2.3 Indirekt programmering av positionsattribut (GP)

Positionsattribut, som t.ex. den inkrementella eller absoluta programmeringen av axelpositionerna, kan i förbindelse med nyckelordet **GP** programmeras indirekt som variabler.

Användning

Den indirekta programmeringen av positionsattribut finner användning i **ersättningscykler**, eftersom här följande fördel gentemot programmeringen av positionsattribut som nyckelord (t.ex. **IC**, **AC**, ...) består:

Genom den indirekta programmeringen som variabler behövs **ingen** CASE-anvisning, som förgrenar sig över alla möjliga positionsattribut.

Syntax

```
<POSITIONIERBEFEHL> [<Achse/Spindel>] =
GP (<Position>, <Positionsattribut>)
<Achse/Spindel> = GP (<Position>, <Positionsattribut>)
```

Betydelse

<POSITIONIERBEFEHL> []:	Följande positioneringskommandon kan programmeras tillsammans med nyckelordet GP: POS, POSA, SPOS, SPOSA Dessutom möjliga: • alla i kanalen förefintliga axel-/spindelbeteckningar: <Achse/Spindel> • Variabel axel-/spindelbeteckning AX
<Achse/Spindel>:	Axel/spindel som ska positioneras
GP ():	Nyckelord för positionering
<Position>:	Parameter 1 Axel-/spindelposition som konstant eller variabel
<Positionsattribut>:	Parameter 2 Positionsattribut (t.ex. positionsinkörningsmode) som variabel (t.ex. B. \$P_SUB_SPOSMODE) eller som nyckelord (IC, AC, ...)

De av variablerna levererade värdena har följande betydelse:

Värde	Betydelse	Tillåten vid:
0	Ingen ändring av positionsattributet	
1	AC	POS, POSA, SPOS, SPOSA, AX, axeladress
2	IC	POS, POSA, SPOS, SPOSA, AX, axeladress
3	DC	POS, POSA, SPOS, SPOSA, AX, axeladress
4	ACP	POS, POSA, SPOS, SPOSA, AX, axeladress
5	ACN	POS, POSA, SPOS, SPOSA, AX, axeladress
6	OC	-
7	PC	-
8	DAC	POS, POSA, AX, axeladress
9	DIC	POS, POSA, AX, axeladress
10	RAC	POS, POSA, AX, axeladress
11	RIC	POS, POSA, AX, axeladress
12	CAC	POS, POSA
13	CIC	POS, POSA
14	CDC	POS, POSA
15	CACP	POS, POSA
16	CACN	POS, POSA

Exempel

Vid en aktiv synkronspindelkoppling mellan styrspindeln S1 och följdspindeln S2 anropas med SPOS-kommandot i huvudprogrammet följande ersättningsscykel för positioneringen av spindeln.

Positioneringen görs via anvisningen i N2230:

```
SPOS[1]=GP($P_SUB_SPOSIT,$P_SUB_SPOSMODE) SPOS[2]=GP($P_SUB_SPOSIT,
$P_SUB_SPOSMODE)
```

Den position som ska uppsökas läsas ur systemvariabeln \$P_SUB_SPOSIT, positionsinkörningsmode ur systemvariabeln \$P_SUB_SPOSMODE.

Programkod	Kommentar
N1000 PROC LANG_SUB DISPLOF SBLOF	
...	
N2100 IF(\$P_SUB_AXFCT==2)	
N2110	; Ersättning av SPOS / SPOSA / M19-kommandot vid aktiv synkronspindelkoppling
N2185 DELAYFSTON	; Början av Stopp-Delay-område
N2190 COUPOF(S2,S1)	; Inaktiver synkronspindelkoppling
N2200	; Positionera styr- och följdspindel
N2210 IF(\$P_SUB_SPOS==TRUE) OR (\$P_SUB_SPOSA==TRUE)	
N2220	; Positionera spindel med SPOS:
N2230 SPOS[1]=GP(\$P_SUB_SPOSIT, \$P_SUB_SPOSMODE)	
SPOS[2]=GP(\$P_SUB_SPOSIT, \$P_SUB_SPOSMODE)	
N2250 ELSE	
N2260	; Positionera spindel med M19:
N2270 M1=19 M2=19	; Positionera styr- och följdspindel
N2280 ENDIF	
N2285 DELAYFSTOF	; Slut på Stopp-Delay-område
N2290 COUPON(S2,S1)	; Aktivera synkronspindelkoppling
N2410 ELSE	
N2420	; Avfrågning om ytterligare ersättningar
...	
N3300 ENDIF	
...	
N9999 RET	

Randvillkor

- I synkronaktioner är den indirekta programmeringen av positionsattribut inte möjlig.

Litteratur

Funktionshandbok Grundfunktioner; BAG, kanal, programdrift, Reset-beteende (K1),
Kapitel: Ersättning av NC-funktioner med underprogram

2.2.4 Indirekt programmering av detaljprogramrader (EXECSTRING)

Med detaljprogramkommandot `EXECSTRING` är det möjligt att utföra en tidigare skapad String-variabel som detaljprogramrad.

Syntax

`EXECSTRING` programmeras i en egen detaljprogramrad:
`EXECSTRING(<String-Variable>)`

Betydelse

<code>EXECSTRING:</code>	Kommando för utförande av en String-variabel som detaljprogramsrad
<code><String-Variable>:</code>	Variabel av typ <code>STRING</code> som innehåller de detaljprogramrader som egentligen ska utföras

Märk

Med `EXECSTRING` kan med undantag av kontrollstrukturer (Sida 112) alla detaljprogramkonstruktioner avföras som kan programmeras i **programdelen till ett detaljprogram**. Uteslutna är därmed `PROC`- och `DEF`-anvisningar samt generellt användningen i `INI`- och `DEF`-filer.

Exempel

Programkod	Kommentar
N100 DEF STRING[100] MY_BLOCK	; Definition av String-variabeln för upptagande av de detaljprogramrader som ska utföras.
N110 DEF STRING[10] MFCT1="M7"	
...	
N200 EXECSTRING(MFCT1 << "M4711")	; Utföra detaljprogramrad "M7 M4711".
...	
N300 R10=1	
N310 MY_BLOCK="M3"	
N320 IF(R10)	
N330 MY_BLOCK = MY_BLOCK << MFCT1	
N340 ENDIF	
N350 EXECSTRING(MY_BLOCK)	; Utföra detaljprogramrad "M3 M7".

2.3 Räknefunktioner

Operator / räknefunktion	Betydelse
+	Addition
-	Subtraktion
*	Multiplikation
/ ¹⁾	Division ¹⁾
DIV ¹⁾	Heltals-division ¹⁾
MOD ¹⁾	Modulo-division (levererar resten från heltals-divisionen) ¹⁾
:	Länkingsoperator (för FRAME-variabler)
SIN ()	Sinus
COS ()	Cosinus
TAN ()	Tangens
ASIN ()	Arcussinus
ACOS ()	Arcuscosinus
ATAN2 (,) ¹⁾	Arcustangens2 ¹⁾
SQRT ()	Kvadratrot
ABS ()	Belopp
POT ()	2:a potens (kvadrat)
TRUNC ()	Heltalig del Noggrannhet vid jämförelsekommando inställbar med TRUNC (se "Noggrannhetskorrigerig vid jämförelsefel (TRUNC) (Sida 77)")
ROUND ()	Avrundning till ett helt tal
LN ()	Naturlig logaritm
EXP ()	Exponentialfunktion
MINVAL ()	Minsta värdet av två variabler (se "Minimum, maximum och område med variabler (MINVAL, MAXVAL, BOUND) (Sida 79)")
MAXVAL ()	Största värdet av två variabler (se "Minimum, maximum och område med variabler (MINVAL, MAXVAL, BOUND) (Sida 79)")
BOUND ()	Variabelvärde som ligger i det definierade värdeområdet (se "Minimum, maximum och område med variabler (MINVAL, MAXVAL, BOUND) (Sida 79)")
CTRANS ()	Förflyttning
CROT ()	Rotation
CSCALE ()	Skalförändring
CMIRROR ()	Spegling
1) Se avsnitt "Exempel"	

Programmering

Vid räknefunktionerna gäller det vanliga matematiska skrivsättet. Prioriteter vid genomarbetningen sätts med runda parenteser. För de trigonometriska och deras inversa funktioner gäller graduppgift (höger vinkel = 90°).

Exempel

Division: /

(Typ REAL) = Typ INT eller Typ REAL) / (Typ INT eller Typ REAL);

Exempel: $3 / 4 = 0.75$

Heltals-division: DIV

(Typ INT) = (Typ INT eller REAL) / (Typ INT eller REAL);

Exempel: $7 \text{ DIV } 4.1 = 1$

Modulo-division (levererar resten från heltals-divisionen): MOD

(Typ REAL) = (Typ INT eller REAL) MOD (Typ INT eller REAL);

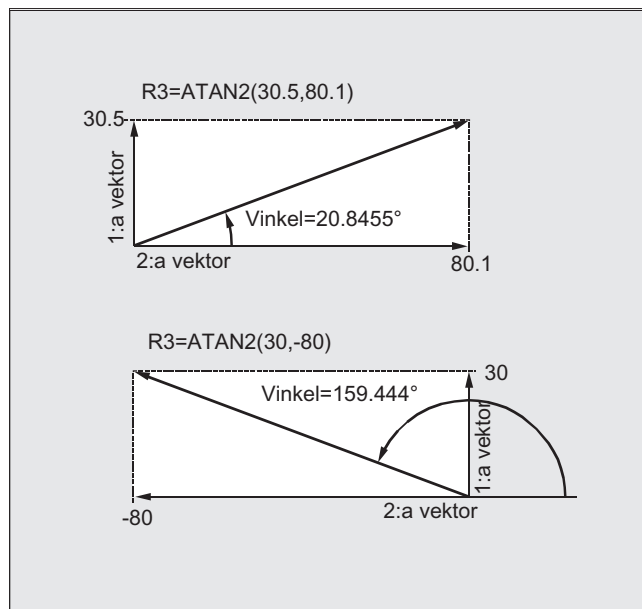
Exempel: $7 \text{ MOD } 4.1 = 2.9$

Arcustangens2: ATAN2

Räknefunktionen ATAN2 beräknar ur två mot varandra vinkelrätt stående vektorer vinkeln för sumnavektorn.

Resultatet ligger inom området för fyra kvadranter ($-180^\circ < 0 < +180^\circ$).

Bas för vinkelreferensen är alltid det 2:a värdet i positiv riktning.



Programmeringsexempel

Programkod	Kommentar
R1=R1+1	; Nytt R1 = gammalt R1 + 1
R1=R2+R3 R4=R5-R6 R7=R8*R9	
R10=R11/R12 R13=SIN(25.3)	
R14=R1*R2+R3	; Punkträkning går före streckräkning.

Programkod	Kommentar
R14=(R1+R2)*R3	; Uttryck inom parentes beräknas först.
R15=SQRT(POT(R1)+POT(R2))	; Inre parenteser löses upp först: R15 = kvadratroten av (R1 ² + R2 ²)
RESFRAME=FRAME1:FRAME2	; FRAME-förbindelse med länkingsoperator
FRAME3=CTrans(...):CROT(...)	Värdetillordning till en Frame-komponent

2.4 Jämförelse- och logiska operationer

Jämförelseoperationer kan t.ex. användas till formulering av ett hoppvillkor. Jämförbara är därvid också komplexa uttryck.

Jämförelseoperationerna är användbara för variabler av typ `CHAR`, `INT`, `REAL` och `BOOL`. Vid typen `CHAR` jämförs kodvärdet.

Vid typerna `STRING`, `AXIS` och `FRAME` är möjliga: `==` und `<>`, som också kan användas i synkronaktioner för operationer av typ `STRING`.

Resultatet av jämförande operationer är alltid av typ `BOOL`.

Logiska operatorer tjänar till sammanfogande av sanningsvärden.

De logiska operationerna är användbara endast på variabler av typ `BOOL`. Via intern typkonvertering är de användbara också på datatyperna `CHAR`, `INT`, och `REAL`.

Vid de logiska (boolska) operationerna gäller för datatyperna `BOOL`, `CHAR`, `INT` och `REAL`:

- 0 motsvarar: `FALSE`
- skilt från 0 motsvarar: `TRUE`

Bitvis logiska operatorer

Med variablerna av typ `CHAR` och `INT` kan också bitvis logiska operationer göras. Eventuellt följer automatiskt en typkonvertering.

Programmering

Jämförelseoperator	Betydelse
<code>==</code>	lika med
<code><></code>	skilt från
<code>></code>	större än
<code><</code>	mindre än
<code>>=</code>	större än eller lika med
<code><=</code>	mindre än eller lika med

Logisk operator	Betydelse
<code>AND</code>	OCH
<code>OR</code>	ELLER
<code>NOT</code>	Negation
<code>XOR</code>	Exklusivt-ELLER

Bitvis logisk operatorer	Betydelse
<code>B_AND</code>	bitvis OCH
<code>B_OR</code>	bitvis ELLER
<code>B_NOT</code>	bitvis negation
<code>B_XOR</code>	bitvis exklusivt-ELLER

Märk

I aritmetiska uttryck kan genomarbetningsföljden för alla operatorer fastläggas genom runda parenteser och därmed avvika från de normala prioritetsreglerna.

Märk

Mellanslag måste skrivas mellan BOOLSKA operander och operatorer.

Märk

Operatören `B_NOT` hänför sig till endast en operand. Denna står efter operatören.

Exempel

Exempel 1: Jämförelseoperatorer

```
IF R10>=100 GOTO F ZIEL
```

eller

```
R11=R10>=100
```

```
IF R11 GOTO F ZIEL
```

Resultatet av jämförelsen `R10>=100` sparas först intermediärt i R11.

Exempel 2: Logiska operatorer

```
IF (R10<50) AND ($AA_IM[X]>=17.5) GOTO F ZIEL
```

eller

```
IF NOT R10 GOTO B START
```

`NOT` hänför sig till endast en operand.

Exempel 3: Bitvis logiska operatorer

```
IF $MC_RESET_MODE_MASK B_AND 'B10000' GOTO F ACT_PLANE
```


2.5 Noggrannhetskorrigerig vid jämförelsefel (TRUNC)

TRUNC-kommandot skär av den med en noggrannhetsfaktor multiplicerade operanden.

Inställbar noggrannhet vid jämförelsekommandon

Detaljprogramdata av typ REAL framställs internt i IEEE-format med 64 bit. På grund av denna framställningsform kan decimaltal avbildas inexakt, vilket vid en jämförelse med idealt beräknade värden kan leda till oväntade resultat.

Relativ likhet

Så att inte de genom framställningsformen uppkomna inexakta resultaten förfalskar programflödet, kontrolleras vid jämförelsekommandona inte med avseende på absolut likhet utan med avseende på relativ likhet.

Syntax

Noggrannhetskorrigerig vid jämförelsefel

TRUNC (R1*1000)

Betydelse

TRUNC:	Avskärning av siffror efter komma
--------	-----------------------------------

Beaktad relativ likhet för 10^{-12} vid

- Likhet: (==)
- Olikhet: (<>)
- Större än - lika med: (>=)
- Mindre än - lika med: (<=)
- Större än/mindre än: (><) med absolut likhet
- Större än: (>)
- Mindre än: (<)

Kompatibilitet

Av kompatibilitetsskäl kan kontrollen med avseende på relativ likhet vid (>) och (<) inaktiveras genom att ställa in maskindatum MD10280 \$MN_ PROG_FUNCTION_MASK Bit0 = 1.

Märk

Jämförelser med data av typ REAL är av nämnda orsaker generellt behäftade med viss bristande noggrannhet. Vid ej acceptabla avvikelser måste bytas till INTEGER-räkning genom att operanden multipliceras med en noggrannhetsfaktor och därefter skärs av med TRUNC.

Synkronaktioner

Det beskrivna beteende för jämförelsekommandon gäller också för synkronaktioner.

Exempel

Exempel 1: Noggrannhetsgranskningar

Programkod	Kommentar
N40 R1=61.01 R2=61.02 R3=0.01	; tillordning av begynnelsevärden
N41 IF ABS(R2-R1) > R3 GOTOF FEHLER	; Hopp skulle hittills utföras
N42 M30	; Programslut
N43 FEHLER: SETAL(66000)	
R1=61.01 R2=61.02 R3=0.01	; tillordning av begynnelsevärden
R11=TRUNC(R1*1000) R12=TRUNC(R2*1000)	; Noggrannhetskorrigerig
R13=TRUNC(R3*1000)	
IF ABS(R12-R11) > R13 GOTOF FEHLER	; Hopp kommer inte längre att utföras
M30	; Programslut
FEHLER: SETAL(66000)	

Exempel 2: Bilda kvot av båda operanderna och utvärdera

Programkod	Kommentar
R1=61.01 R2=61.02 R3=0.01	; tillordning av begynnelsevärden
IF ABS((R2-R1)/R3)-1) > 10EX-5 GOTOF FEHLER	; Hopp utförs inte
M30	; Programslut
FEHLER: SETAL(66000)	

2.6 Minimum, maximum och område med variabler (MINVAL, MAXVAL, BOUND)

Med kommandona `MINVAL` och `MAXVAL` kan värdena för två variabler jämföras med varandra. Som resultat skickas det mindre värdet (vid `MINVAL`) resp. det större värdet (vid `MAXVAL`) tillbaka.

Med kommandot `BOUND` kan kontrolleras om värdet för en kontrollvariabel ligger inom det definierade värdeområdet.

Syntax

```
<Kleinerer Wert>=MINVAL(<Variable1>,<Variable2>)
<Größerer Wert>=MAXVAL(<Variable1>,<Variable2>)
<Rückgabewert>=BOUND(<Minimum>,<Maximum>,<Prüfvariable>)
```

Betydelse

MINVAL:	Fastställer det mindre värdet av två variabler (<Variable1>, <Variable2>)
<Kleinerer Wert>:	Resultatvariabel för kommandot <code>MINVAL</code> Ställs in på det mindre variabelvärdet.
MAXVAL:	Fastställer det större värdet av två variabler (<Variable1>, <Variable2>)
<Größerer Wert>:	Resultatvariabel för kommandot <code>MAXVAL</code> Ställs in på det större variabelvärdet.
BOUND:	Kontrollerar om en variabel (<Prüfvariable>) ligger inom ett definierat värdeområdet.
<Minimum>:	Variabel som definierar minimivärdet i värdeområdet
<Maximum>:	Variabel som definierar maximivärdet i värdeområdet
<Rückgabewert>:	Resultatvariabel för kommandot <code>BOUND</code> Om värdet för kontrollvariabeln ligger inom det definierade värdeområdet då ställs resultatvariabeln in på värdet för kontrollvariabeln. Om värdet för kontrollvariabeln är större än maximivärdet då ställs resultatvariabeln in på maximivärdet för definitionsområdet. Om värdet för kontrollvariabeln är mindre än minimivärdet då ställs resultatvariabeln in på minimivärdet för definitionsområdet.

Märk

`MINVAL`, `MAXVAL` och `BOUND` kan också programmeras i synkronaktioner.

Märk

Beteende vid likhet

Vid likhet levereras vid `MINVAL`/`MAXVAL` detta samma värde. Vid `BOUND` lämnas värdet för den variabel som ska kontrolleras tillbaka.

Exempel

Programkod	Kommentar
DEF REAL rVar1=10.5, rVar2=33.7, rVar3, rVar4, rVar5, rValMin, rValMax, rRetVar	
rValMin=MINVAL(rVar1,rVar2)	; rValMin ställs in på värdet 10.5.
rValMax=MAXVAL(rVar1,rVar2)	; rValMax ställs in på värdet 33.7.
rVar3=19.7	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 ligger inom gränserna, rRetVar ställs in på 19.7.
rVar3=1.8	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 ligger utanför minimigränsen, rRetVar ställs in på 10.5.
rVar3=45.2	
rRetVar=BOUND(rVar1,rVar2,rVar3)	; rVar3 ligger ovanför maximigränsen, rRetVar ställs in på 33.7.

2.7 Prioritet för operationerna

Varje operator är tillordnad en prioritet. Vid utvärderingen av ett uttryck behandlas alltid först operatorerna med den högsta prioriteten. Vid operatorer med samma rang sker utvärderingen från vänster till höger.

I aritmetiska uttryck kan genomarbetningsföljden för alla operatorer fastläggas genom runda parenteser och därmed avvika från de normala prioritetsreglerna.

Ordningsföljde för operatorerna

Från den högsta till den lägsta prioriteten

1.	NOT, B_NOT	Negation, bitvis negation
2.	*, /, DIV, MOD	Multiplikation, division
3.	+, -	Addition, subtraktion
4.	B_AND	bitvis OCH
5.	B_XOR	bitvis exklusivt ELLER
6.	B_OR	bitvis ELLER
7.	AND	OCH
8.	XOR	exklusivt ELLER
9.	OR	ELLER
10.	<<	Länkning av strings, resultattyp STRING
11.	==, <>, >, <, >=, <=	Jämförelseoperatorer

Märk

Länkingsoperator ":" för frames får inte förekomma med andra operatorer i ett uttryck. En prioritetsbedömning för denna operator är därför inte erforderlig.

Exempel If-anvisning

```
If (otto==10) and (anna==20) gotof end
```

2.8 Möjliga typkonverteringar

Funktion

Typkonvertering vid tillordning

Det konstanta siffervärdet, variabeln eller det uttryck som tillordnas en variabel måste vara kompatibelt med typen för denna variabel. Är detta uppfyllt så omvandlas automatiskt vid tillordningen av typen.

Möjliga typkonverteringar

	till	REAL	INT	BOOL	CHAR	STRING	AXIS	FRAME
från								
REAL		ja	ja*	ja ¹⁾	ja*	–	–	–
INT		ja	ja	ja ¹⁾	ja ²⁾	–	–	–
BOOL		ja	ja	ja	ja	ja	–	–
CHAR		ja	ja	ja ¹⁾	ja	ja	–	–
STRING		–	–	ja ⁴⁾	ja ³⁾	ja	–	–
AXIS		–	–	–	–	–	ja	–
FRAME		–	–	–	–	–	–	ja

Förklaringar

- * Vid typomvandling från REAL till INT avrundas uppåt vid brutet värde ≥ 0.5 , annars avrundas nedåt (jfr. funktion ROUND)
- 1) Värde $\neq 0$ motsvarar TRUE, värde $= 0$ motsvarar FALSE
- 2) När värdet ligger inom det tillåtna talområdet
- 3) När bara 1 tecken
- 4) Stringlängd 0 = >FALSE, annars TRUE

Märk

Är vid konverteringen ett värde större än målområdet kommer ett felmeddelande.

Uppträder i ett uttryck blandade typer så genomförs en typanpassning. Typomvandlingar är möjliga också i synkronaktioner se kapitel "Rörelsesynkronaktioner, Implicit typomvandling".

2.9 Stringoperationer

Stringoperationer

Förutom de klassiska operationerna "Tillordning" och "Jämförelse" är följande stringoperationer möjliga:

- Typkonvertering enligt STRING (AXSTRING) (Sida 83)
- Typkonvertering av STRING (NUMBER, ISNUMBER, AXNAME) (Sida 84)
- Länkning av strings (<<) (Sida 85)
- Omvandling i små/stora bokstäver (TOLOWER, TOUPPER) (Sida 86)
- Bestämma längd för en string (STRLEN) (Sida 87)
- Söka tecken/string i string (INDEX, RINDEX, MINDEX, MATCH) (Sida 88)
- Val av en delstring (SUBSTR) (Sida 89)
- Läsning och skrivning av enskilda tecken (Sida 89)
- Formatera string (SPRINT) (Sida 91)

Specialbetydelse för 0-tecknet

0-tecknet interpreteras internt som slutidentifikation för en string. Ersätts ett tecken av 0-tecknet förkortas stringen därmed.

Exempel:

Programkod	Kommentar
DEF STRING[20] STRG="Achse . steht"	
STRG[6]="X"	
MSG(STRG)	; Levererar meddelandet "Axel X står".
STRG[6]=0	
MSG(STRG)	; Levererar meddelandet "Axel".

2.9.1 Typkonvertering enligt STRING (AXSTRING)

Genom funktionen "Typkonvertering enligt STRING" låter sig variabler av olika typ användas som beståndsdel i ett meddelande (MSG).

Följer vid användning av operatoren << implicit för datatyperna INT, REAL, CHAR och BOOL (se "Länkning av strings (<<) (Sida 85)").

Ett INT-värde omvandlas till den normalt läsbara formen. Vid REAL-värden anges upp till 10 siffror efter komma.

Med kommandot AXSTRING kan variabel av typ AXIS omvandlas till STRING.

Syntax

```
<STRING_ERG> = << <bel._Typ>
<STRING_ERG> = AXSTRING(<Achsbezeichner>)
```

Betydelse

<STRING_ERG>:	Variabel för resultatet av typkonverteringen	
	Typ:	STRING
<bel._Typ>:	Variabel-typer INT, REAL, CHAR, STRING och BOOL	
AXSTRING:	Kommandot AXSTRING levererar den angivna axelbeteckningen som string.	
<Achsbezeichner>:	Variabel för axelbeteckning	
	Typ:	AXIS

Märk

FRAME-variabler kan inte konverteras.

2.9.2 Typkonvertering av STRING (NUMBER, ISNUMBER, AXNAME)

Med kommandot **NUMBER** konverteras från STRING till REAL. Konverterbarheten kan kontrolleras med kommandot **ISNUMBER**.

Med kommandot **AXNAME** konverteras en string till datatypen **AXIS**.

Syntax

```
<REAL_ERG>=NUMBER("<String>")
<BOOL_ERG>=ISNUMBER("<String>")
<AXIS_ERG>=AXNAME("<String>")
```

Betydelse

NUMBER:	Kommandot NUMBER levererar det av <String> framställda talet tillbaka som REAL-värde.		
<String>:	Variabel som ska konverteras från typ STRING		
<REAL_ERG>:	Variabel för resultatet av typkonverteringen med NUMBER		
	Typ:	REAL	
ISNUMBER:	Med kommandot ISNUMBER kan kontrolleras om <String> kan omvandlas till ett giltigt tal.		
<BOOL_ERG>:	Variabel för resultatet av avfrågningen med ISNUMBER		
	Typ:	BOOL	
	Värde:	TRUE	ISNUMBER levererar värdet TRUE, när <String> enligt språkreglerna föreställer ett giltigt REAL-tal.
		FALSE	Levererar ISNUMBER värdet FALSE, utlöses vid anrop av NUMBER larm med samma <String>.

AXNAME:	Kommandot AXNAME omvandlar den angivna <String> till en axelbeteckning. Observera: Kan <String> inte tillordnas någon projekterad axelbeteckning utlöses ett larm.	
<AXIS_ERG>:	Variabel för resultatet av typkonverteringen med AXNAME	
	Typ:	AXIS

Exempel

Programkod	Kommentar
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

2.9.3 Länkning av strings (<<)

Funktionen "Länkning av strings" skaffar möjligheten att sätta samman en string av enskilda beståndsdelar.

Länkningen realiseras vid operatorm "<<". Denna operator har STRING som måltyp för alla kombinationerna av bas typerna CHAR, BOOL, INT, REAL och STRING. En eventuellt nödvändig konvertering görs enligt bestående regler.

Syntax

```
<bel._Typ> << <bel._Typ>
```

Betydelse

<bel._Typ>:	Variabel av typ CHAR, BOOL, INT, REAL eller STRING
<< :	Operator för länkningen av variabler (<bel._Typ>) till en sammansatt teckenföljd (typ STRING). Denna operator är också den enda så kallade "unitära" varianten som är tillgänglig. Så är det möjligt att utföra en explicit typomvandling till STRING (inte för FRAME och AXIS): << <bel._Typ>

Exempelvis låter sig så ett meddelande eller ett kommando sammansättas av textlistor och parametrar (som ett komponentnamn) infogas:

```
MSG (STRG_TAB [LOAD_IDX] <<BAUSTEIN_NAME)
```

Märk

Intermediära resultat vid stringlänknigen får inte överskrida den maximala stringlängden.

Märk

Typerna FRAME och AXIS kan inte användas tillsammans med operatoren "<<".

Exempel**Exempel 1: Länkning av strings**

Programkod	Kommentar
DEF INT IDX=2	
DEF REAL VALUE=9.654	
DEF STRING[20] STRG="INDEX:2"	
IF STRG=="Index:"<<IDX GOTOF NO_MSG	
MSG("Index:"<<IDX<<"/Wert:"<<VALUE)	; Indikering: "Index:2/värde:9.654"
NO_MSG:	

Exempel 2: Explicit typkonvertering med <<

Programkod	Kommentar
DEF REAL VALUE=3.5	
<<VALUE	; Den angivna variabeln av typ REAL konverteras till typ STRING.

2.9.4 Omvandling i små/stora bokstäver (TOLOWER, TOUPPER)

Funktionen "Omvandling i små/stora bokstäver" tillåter att omvandla alla bokstäver i en teckenföljd till en enhetlig framställning.

Syntax

```
<STRING_ERG>=TOUPPER("<String>")
<STRING_ERG>=TOLOWER("<String>")
```

Betydelse

TOUPPER:	Med kommandot TOUPPER omvandlas alla bokstäver i en teckenföljd till stora bokstäver.
TOLOWER:	Med kommandot TOLOWER omvandlas alla bokstäver i en teckenföljd till små bokstäver.

<String>:	Teckenföljd som ska omvandlas	
	Typ:	STRING
<STRING_ERG>:	Variabel för resultatet av omvandlingen	
	Typ:	STRING

Exempel

Eftersom det också är möjligt att utlösa inmatningar vid användargränssnittet kan en enhetlig framställning med små eller stora bokstäver uppnås:

Programkod

```
DEF STRING [29] STRG
...
IF "LEARN.CNC"==TOUPPER(STRG) GOTOF LOAD_LEARN
```

2.9.5 Bestämma längd för en string (STRLEN)

Med kommandot `STRLEN` är det möjligt att bestämma längden på en teckenföljd.

Syntax

```
<INT_ERG>=STRLEN("<STRING>")
```

Betydelse

STRLEN:	Med kommandot <code>STRLEN</code> bestäms längden för den angivna teckenföljden. Det antal tecken matas tillbaka som - räknat från teckenföljdens början - inte är några 0-tecken.	
<String>:	Teckenföljd vars längd ska bestämmas	
	Typ:	STRING
<INT_ERG>:	Variabel för resultatet av bestämningen	
	Typ:	INT

Exempel

Funktionen i sammanhang med åtkomsten av enkeltecken gör det möjligt att bestämma teckenföljdens slut:

Programkod

```
IF (STRLEN(BAUSTEIN_NAME)>10) GOTOF FEHLER
```

2.9.6 Söka tecken/string i string (INDEX, RINDEX, MINDEX, MATCH)

Denna funktionalitet tillåter det att söka enskilda tecken resp. en string i en ytterligare string. Funktionsresultaten anger vid vilken position i stringen tecknet/ stringen hittades i den string som undersöks.

Syntax

INT_ERG=INDEX (STRING, CHAR) ; **Resultattyp: INT**

INT_ERG=RINDEX (STRING, CHAR) ; **Resultattyp: INT**

INT_ERG=MINDEX (STRING, STRING) ; **Resultattyp: INT**

INT_ERG=MATCH (STRING, STRING) ; **Resultattyp: INT**

Semantik

Sökfunktioner: De skickar tillbaka den position i stringen (första parameter) där sökningen var framgångsrik. Kan tecknet/stringen inte hittas ges värdet -1 tillbaka. Det första tecknet har därvid positionen 0.

Betydelse

INDEX:	söker det som andra parameter angivna tecknet (framifrån) i den första parametern.
RINDEX:	söker det som andra parameter angivna tecknet (bakifrån) i den första parametern.
MINDEX:	motsvarar funktionen INDEX såvida inte en lista med tecken (som string) överlämnas från vilka indexet i det första hittade tecknet matas tillbaka.
MATCH:	söker en string i en string.

Så låter sig strings delas upp enligt bestämda kriterier till exempel vid positioner med mellanslag eller skiljetecken för sökväg ("").

Exempel

Uppdelning av en inmatning i sökvägs- och komponentnamn

Programkod	Kommentar
DEF INT PFADIDX, PROGIDX	
DEF STRING[26] EINGABE	
DEF INT LISTIDX	
EINGABE = "/"_N_MPF_DIR/_N_EXECUTE_MPF"	
LISTIDX = MINDEX (EINGABE, "M,N,O,P") + 1	: Som värde i LISTIDX skickas 3 tillbaka; eftersom "N" är det första tecknet i parametern EINGABE, från urvalslisten framifrån.
PFADIDX = INDEX (EINGABE, "/") +1	; därmed gäller: PFADIDX = 1
PROGIDX = RINDEX (EINGABE, "/") +1	; därmed gäller: PROGIDX = 12

Med funktionen SUBSTRING kan valfria delar inom en string läsas.

```
<STRING_ERG>=SUBSTR(<String>,<Index>,<Länge>)  
<STRING_ERG>=SUBSTR(<String>,<Index>)
```

SUBSTR:	Funktionen levererar från <String> en delstring, utgående från <Index> med den angivna <Länge>. Är parametern <Länge> inte given, levererar funktionen en delstring utgående från <Index> fram till stringslut.
<Index>:	Begynnelseposition för delstringen inom stringen. Ligger begynnelsepositionen bakom stringslutet matas en tom string ("") tillbaka. Första tecken i stringen: Index = 0 Värdeområde: 0 ... (stringlängd -1)
<Länge>:	Längd för delstringen. Anges en för stor längd, matas endast delstringen till stringslut tillbaka. Värdeområde: 1 ... (stringlängd -1)

Programkod	Kommentar
DEF STRING[29] ERG	
;	1
;	0123456789012345678
ERG = SUBSTR("QUITTUNG: 10 bis 99", 10, 2)	; ERG == "10"
ERG = SUBSTR("QUITTUNG: 10 bis 99", 10)	; ERG == "10 till 99"

Inom en string kan enskilda tecken läsas och skrivas.

Därvid ska följande randvillkor iakttas:

- endast möjligt vid användardefinierade variabler, inte vid systemvariabler
- enskilda tecken i en string överlämnas vid underprogramanrop endast "call by value"

Syntax

```
<Zeichen>=<String>[<Index>]
<Zeichen>=<String_Array>[<Array_Index>,<Index>]
<String>[<Index>]=<Zeichen>
<String_Array>[<Array_Index>,<Index>]=<Zeichen>
```

Betydelse

<String>:	Valfri string
<Zeichen>:	Variabel av typ CHAR
<Index>:	Position för tecknet inom stringen. Första tecken i stringen: Index = 0 Värdeområde: 0 ... (stringlängd -1)

Exempel

Exempel 1: Variabel meddelande

Programkod	Kommentar
; 0123456789	
DEF STRING[50] MELDUNG = "Achse n hat Position erreicht"	
MELDUNG[6] = "X"	
MSG(MELDUNG)	; "Axel X har uppnått position"

Exempel 2: Utvärdering av en systemvariabel

Programkod	Kommentar
DEF STRING[50] STRG	; Intermediärt minne för systemvariabel
...	
STRG = \$P_MMCA	; Ladda systemvariabel
IF STRG[0] == "E" GOTO ...	; Utvärdering av systemvariabeln

Exempel 3: Parameteröverföring "call by value" och "call by reference"

Programkod	Kommentar
; 0123456	
DEF STRING[50] STRG = "Achse X"	
DEF CHAR CHR	
...	
EXTERN UP_VAL(ACHSE)	; Definition underprogram med "call by value" parameter

Programkod	Kommentar
EXTERN UP_REF(VAR ACHSE)	; Definition underprogram med "call by ref." Parameter
...	
UP_VAL(STRG[6])	; Parameteröverföring "by value"
...	
CHR = STRG[6]	; Spara i intermediärt minne
UP_REF(CHR)	; Parameteröverföring "by reference"

2.9.9 Formatera string (SPRINT)

Med den fördefinierade funktionen SPRINT kan teckenföljder formateras och t.ex. förberedas för uppgiften på det externa instrumentet (se även "Process DataShare - utgåva på en extern apparat/fil (EXTOPEN, WRITE, EXTCLOSE) (Sida 652)").

Syntax

```
"<Ergebnis_String>"=SPRINT("<Format_String>",<Wert_1>,<Wert_2>,...,
<Wert_n>)
```

Betydelse

SPRINT:	Beteckning för en fördefinierad funktion, som levererar ett värde av typ STRING.
"<Format_String>":	Teckenföljd som innehåller fasta och variabla delar. De variabla delarna fastläggs med formatstyrningstecknet % och en efterföljande formatbeskrivning.
< Wert_1>,< Wert_2>,...,< Wert_n>:	Värde i form av en konstant eller NC-variabel, som infogas på det ställe på vilket det n:e formatstyrningstecknet % står, i enlighet med formatbeskrivningen i <Format_String>.
"<Ergebnis_String>":	Formaterad teckenföljd (maximalt 400 bytes)

Disponibla formatbeskrivningar

%B:	Omvandling till stringen "TRUE", när det värde som ska omvandlas: • är skilt från 0. • inte är någon tom string (vid stringvärden). Omvandling till stringen "FALSE", när det värde som ska omvandlas: • är lika med 0. • är en tom string. Exempel: <pre>N10 DEF BOOL BOOL_VAR=1 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF BOOL_VAR:%B", BOOL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF BOOL_VAR:TRUE".
%C:	Omvandling till ett ASCII-tecken. Exempel: <pre>N10 DEF CHAR CHAR_VAR="X" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF CHAR_VAR:%C", CHAR_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF CHAR_VAR:X".
%D:	Omvandling till en string med ett heltaligt värde (INTEGER). Exempel: <pre>N10 DEF INT INT_VAR=123 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF INT_VAR:%D", INT_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF INT_VAR:123".
%<m>D:	Omvandling till en string med ett heltaligt värde (INTEGER). Stringen har en minsta längd på <m> tecken. Platser som saknas fylls med mellanslag från vänster kant. Exempel: <pre>N10 DEF INT INT_VAR=-123 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF INT_VAR:%6D", INT_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF INT_VAR:xx-123" ("x" står i exemplet i stället för mellanslag).
%F:	Omvandling till en string med ett deimaltal med 6 positioner efter komma. Positionerna efter komma avrundas ev. eller fylls med 0. Exempel: <pre>N10 DEF REAL REAL_VAR=-1.2341234EX+03 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%F", REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR: -1234.123400".

%<m>F:	Omvandling till en string med ett decimaltal med 6 positioner efter komma och en total längd på minst <m> tecken. Positionerna efter komma avrundas ev. eller fylls med 0. Tecken som saknas på det totala längden <m> fylls på med mellanslag från vänster kant. Exempel: <pre>N10 DEF REAL REAL_VAR=-1.23412345678EX+03 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15F",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR: xxx-1234.123457" ("x" står i exemplet i stället för mellanslag).
%.<n>F:	Omvandling till en string med ett decimaltal med <n> positioner efter komma. Positionerna efter komma avrundas ev. eller fylls med 0. Exempel: <pre>N10 DEF REAL REAL_VAR=-1.2345678EX+03 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3F",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR: -1234,568".
%<m>.<n>F:	Omvandling till en string med ett decimaltal med <n> positioner efter komma och en total längd på minst <m> tecken. Positionerna efter komma avrundas ev. eller fylls med 0. Tecken som saknas på det totala längden <m> fylls på med mellanslag från vänster kant. Exempel: <pre>N10 DEF REAL REAL_VAR=-1.2341234567890EX+03 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%10.2F",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:xx-1234,12" ("x" står i exemplet i stället för mellanslag).
%E:	Omvandling till en string med ett decimaltal i exponential gestalt. Mantissan skrivs normalt med en position före decimalkomma och 6 positioner efter decimalkomma. Positionerna efter komma avrundas ev. eller fylls med 0. Exponenten börjar med nyckelordet "EX". Förtecknet följer "+" eller "-" och ett två- eller tresiffrigt tal. Exempel: <pre>N10 DEF REAL REAL_VAR=-1234.567890 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%E",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR: -1.234568EX+03".
%<m>E:	Omvandling till en string med ett decimaltal i exponential gestalt med en total längd på minst <m> tecken. Tecken som saknas fylls med mellanslag från vänster kant. Mantissan skrivs normalt med en position före decimalkomma och 6 positioner efter decimalkomma. Positionerna efter komma avrundas ev. eller fylls med 0. Exponenten börjar med nyckelordet "EX". Förtecknet följer "+" eller "-" och ett två- eller tresiffrigt tal. Exempel: <pre>N10 DEF REAL REAL_VAR=-1234,5 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%20E",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:xxxxxx-1.234500EX+03" ("x" står i exemplet i stället för mellanslag).

%.<n>E:	Omvandling till en string med ett deimaltal i exponential gestalt. Mantissan skrivs normalt med en position före decimalkomma och <n> positioner efter decimalkomma. Positionerna efter komma avrundas ev. eller fylls med 0. Exponenten börjar med nyckelordet "EX". Förtecknet följer ("+" eller "-") och ett två- eller tresiffrigt tal. Exempel: <pre>N10 DEF REAL REAL_VAR=-1234,5678 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.2E",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:-1.23EX+03".
%<m>.<n>E:	Omvandling till en string med ett decimaltal i exponential gestalt med en total längd på minst <m> tecken. Tecken som saknas fylls med mellanslag från vänster kant. Mantissan skrivs normalt med en position före decimalkomma och <n> positioner efter decimalkomma. Positionerna efter komma avrundas ev. eller fylls med 0. Exponenten börjar med nyckelordet "EX". Förtecknet följer ("+" eller "-") och ett två- eller tresiffrigt tal. Exempel: <pre>N10 DEF REAL REAL_VAR=-1234,5678 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.2E", REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:xx-1.23EX+03" ("x" står i exemplet i stället för mellanslag).
%G:	Omvandling till en string med ett decimaltal allt efter värdeområde i decimal eller exponential framställning: är det värde som ska framställas beloppsmässigt mindre än 1.0EX-04 eller större än/likat med 1.0EX+06 väljs exponential gestalt, annars decimalframställning. Maximalt sex signifikanta positioner visas, ev. avrundas. Exempel med decimalframställning: <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR: 0,000123457". Exempel med exponential framställning: <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX+06 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%G",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:1.23457EX+06".
%<m>G:	Omvandling till en string med ett decimaltal allt efter värdeområde i decimal eller exponential framställning (som %G). Stringen har en total längd på minst <m> tecken. Tecken som saknas fylls med mellanslag från vänster kant. Exempel med decimalframställning: <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:xxxx0.000123457" ("x" står i exemplet i stället för mellanslag). Exempel med exponential framställning: <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX+06 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%15G",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:xxx1.23457EX+06" ("x" står i exemplet i stället för mellanslag).

%.<n>G:	Omvandling till en string med ett decimaltal allt efter värdeområde i decimal eller exponential framställning. Maximalt <n> signifikanta positioner visas, ev. avrundas. Är det värde som ska framställas beloppsmässigt mindre än 1.0EX-04 eller större än/lika med 1.0EX(+<n>) väljs exponential gestalt, annars decimalframställning. Exempel med decimalframställning: <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR: 0,000123". Exempel med exponential framställning: <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX+03 N20 DEF STRING[80] RESULT N30 RESULT = SPRINT("CONTENT OF REAL_VAR:%.3G",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:1.23EX+03".
%<m>.<n>G:	Omvandling till en string med ett decimaltal allt efter värdeområde i decimal eller exponential framställning (som %.<n>G). Stringen har en total längd på minst <m> tecken. Tecken som saknas fylls med mellanslag från vänster kant. Exempel med decimalframställning: <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX-04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)</pre> Resultat: Stringvariabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:xxx0.0001235" ("x" står i exemplet i stället för mellanslag). Exempel med exponential framställning: <pre>N10 DEF REAL REAL_VAR=1.234567890123456EX+04 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF REAL_VAR:%12.4G",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF REAL_VAR:xx1.235EX+06" ("x" står i exemplet i stället för mellanslag).

%.<n>P:	Omvandling av ett REAL-värde till ett INTEGER-värde under hänsynstagande till <n> positioner efter komma. INTEGER-värdet matas ut som 32 bits binärtal. Kan det värde som ska omvandlas inte framställas med 32 bit avbryts bearbetningen med larm. Eftersom en med formatanvisningen %.<n>P skapad byte-följd också kan innehålla binära nollor, motsvarar den så skapade total-stringen inte längre konventionerna för NC-datatypen STRING. Den kan därför varken sparas i en variabel av typen STRING, eller bearbetas vidare med string-kommandona i NC-språket. Den enda möjliga användningen är parameteröverföringen till WRITE-kommandot med utmatning på ett passande externt instrument (se följande exempel). Så snart som <Format_String> innehåller en formatbeskrivning av typ %P, matas hela stringen, med undantag av det med %.<n>P genererade binäretal, ut enligt MD10750 \$MN_SPRINT_FORMAT_P_CODE i tecken-code ASCII, ISO (DIN6024) eller EIA (RS244). Programmeras ett tecken som inte kan omvandlas, avbryts bearbetningen med larm. Exempel: <pre> 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 ... ; Fehlerhandling N200 WRITE(ERROR,EXT_DEVICE,SPRINT("INTEGER BINARY CODED:%. 3P",REAL_VAR) N210 IF ERROR <> 0 ... ; Fehlerhandling </pre> Resultat: Stringen "INTEGER BINARY CODED: 'H0001E23A'" överförs till utmatningsinstrumentet /ext/dev/1. Det hexadecimala värdet 0x0001E23A motsvarar det decimala värdet 123450.
---------	--

%<m>.<n>P:	Omvandling av ett REAL-värde motsvarande inställningen i maskindatum MD10751 \$MN_SPRINT_FORMAT_P_DECIMAL till en string med: - ett heltal med <m> + <n> siffror eller - ett decimaltal med maximalt <m> positioner före komma och exakt <n> positioner efter komma. Som vid formatbeskrivningen %.<n>P sparas hela stringen i den av MD10750 \$MN_SPRINT_FORMAT_P_CODE fastlagda tecken-coden. Omvandling vid MD10751 = 0: REAL-värdet omvandlas till en string med ett heltal med <m> + <n> siffror. Ev. avrundas positioner efter komma till <n> positioner eller fylls på med 0. Positioner före komma som saknas fylls med mellanslag. Förtecknet minus läggs till från vänster, i stället för förtecknet plus sätts ett mellanslag. Exempel: <pre>N10 DEF REAL REAL_VAR=-123,45 N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("PUNCHED TAPE FORMAT:%5.3P",REAL_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "PUNCHED TAPE FORMAT:-xx123450" ("x" står i exemplet i stället för mellanslag). Omvandling vid MD10751 = 1: REAL-värdet omvandlas till en string med ett decimaltal med maximalt <m> positioner före komma och exakt <n> positioner efter komma. Ev. kortas positionerna före komma och positionerna efter komma avrundas eller fylls ut med 0. Är <n> lika med 0 utgår också decimalkommat(-punkten). Exempel: <pre>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)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "PUNCHED TAPE FORMAT:-123.450 VAR2:23".
%S:	Infogande av en string. Exempel: <pre>N10 DEF STRING[16] STRING_VAR="ABCDEFGH" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%S",STRING_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF STRING_VAR:ABCDEFGH".
%<m>S:	Infogande av en string med minst <m> tecken. Positioner som saknas fylls med mellanslag. Exempel: <pre>N10 DEF STRING[16] STRING_VAR="ABCDEFGH" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10S",STRING_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF STRING_VAR:xxxABCDEFGH" ("x" står i exemplet i stället för mellanslag).

%.<n>S:	Infogande av <n> tecken i en string (med början med det första tecknet). Exempel: <pre>N10 DEF STRING[16] STRING_VAR="ABCDEFGH" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%.3S",STRING_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF STRING_VAR:ABC".
%<m>.<n>S:	Infogande av <n> tecken i en string (med början med det första tecknet). Den totala längden på den skapade stringen har minst <m> tecken. Positioner som saknas fylls med mellanslag. Exempel: <pre>N10 DEF STRING[16] STRING_VAR="ABCDEFGH" N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("CONTENT OF STRING_VAR:%10.5S", STRING_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "CONTENT OF STRING_VAR:xxxxxABCDE" ("x" står i exemplet i stället för mellanslag).
%X:	Omvandling av ett INTEGER-värde till en string med hexadecimal framställning. Exempel: <pre>N10 DEF INT INT_VAR='HA5B8' N20 DEF STRING[80] RESULT N30 RESULT=SPRINT("INTEGER HEXADECEIMAL:%X",INT_VAR)</pre> Resultat: String-variabeln RESULT beskrivs med teckenföljden "INTEGER HEXA-DECIMAL:A5B8".

Märk

Egenskapen i NC-språket, att inte skilja mellan stora och små bokstäver beträffande beteckningar och nyckelord, gäller också för formatbeskrivningarna. De kan därför programmeras utan funktionell skillnad med små eller stora bokstäver.

Kombinationsmöjligheter

Den följande tabellen ger upplysningar om vilka NC-datatyper som kan kombineras med vilken formatbeskrivning. Reglerna för implicit datatypomvandling gäller (se "Datatyper (Sida 56)").

	NC-datatyper						
	BOOL	CHAR	INT	REAL	STRING	AXIS	FRAME
%B	+	+	+	+	+	-	-
%C	-	+	-	-	+	-	-
%D	+	+	+	+	-	-	-
%F	-	-	+	+	-	-	-
%E	-	-	+	+	-	-	-
%G	-	-	+	+	-	-	-
%S	-	+	-	-	+	-	-
%X	+	+	+	-	-	-	-
%P	-	-	+	+	-	-	-

Märk

Tabellen visar att NC-datatyperna AXIS och FRAME inte kan användas direkt i SPRINT-funktionen. Men möjligt är:

- att omvandla datatypen AXIS med funktionen AXSTRING till en string, som sedan kan bearbetas vidare med SPRINT.
 - att läsa de enskilda värdena i datatypen FRAME per Frame-komponentåtkomst. Därigenom får man ett datum av typ REAL, som kan bearbetas vidare med SPRINT.
-

2.10 Programhopp och -förgreningar

2.10.1 Återhopp till programbörjan (GOTOS)

Med kommandot `GOTOS` är det möjligt att hoppa tillbaka till början av ett huvud- eller underprogram för upprepning av programmet.

Via maskindata kan ställas in att vid varje återhopp till programbörjan:

- programkörningstiden sätts på "0".
- räkningen av arbetsstycken höjs med värdet "1".

Syntax

`GOTOS`

Betydelse

GOTOS:	Hoppanvisning med hoppmål programbörjan.	
	Utförandet styrs via NC/PLC-gränssnittssignalen: DB21, ... DBX384.0 (styra programförgrening)	
	Värde:	Betydelse:
	0	Inget återhopp till programbörjan. Programbearbetningen fortsätts med nästa detaljprogramblock efter <code>GOTOS</code> .
	1	Återhopp till programbörjan. Detaljprogrammet upprepas.

Randvillkor

- `GOTOS` utlöser internt ett `STOPRE` (fördekoderingsstopp).
- Vid ett detaljprogram med datadefinitioner (LUD-variabler) hoppas med `GOTOS` till det första programblocket efter definitionsavsnittet, dvs. datadefinitionerna utförs inte på nytt. De definierade variablerna behåller därför det i `GOTOS`-blocket uppnådda värdet och återställs inte på de i definitionsavsnittet programmerade standardvärdena.
- I synkronaktioner och teknologi-cykler står kommandot `GOTOS` inte till förfogande.

Exempel

Programkod	Kommentar
N10 ...	; Programbörjan
...	
N90 GOTOS	; Hopp till programbörjan.
...	

2.10.2 Programhopp till hoppmärken (GOTOB, GOTOF, GOTO, GOTOC)

I ett program kan hoppmärken (Labels) sättas till vilka kan hoppas från andra ställen inom samma program med kommandona GOTOF, GOTOB, GOTO resp. GOTOC.

Programbearbetningen fortsätts sedan med den anvisning som följer omedelbart efter hoppmärket. Därigenom kan förgreningar inom programmet realiseras.

Förutom hoppmärkena är även huvud- och sidoblocksnummer möjliga som hoppmål.

Om ett hoppvillkor (IF ...) är formulerat före hoppanvisningen, då följer programhoppet endast när hoppvillkoret är uppfyllt.

Syntax

```
GOTOB <Sprungziel>
IF <Sprungbedingung> == TRUE GOTOB <Sprungziel>

GOTOF <Sprungziel>
IF <Sprungbedingung> == TRUE GOTOF <Sprungziel>

GOTO <Sprungziel>
IF <Sprungbedingung> == TRUE GOTO <Sprungziel>

GOTOC <Sprungziel>
IF <Sprungbedingung> == TRUE GOTOC <Sprungziel>
```

Betydelse

GOTOB:	Hoppanvisning med hoppmål i riktning program början.	
GOTOF:	Hoppanvisning med hoppmål i riktning programslut.	
GOTO:	Hoppanvisning med sökning av hoppmål. Sökningen görs först i riktning programslut sedan i riktning program början.	
GOTOC:	Verkan som GOTO med skillnaden att larmet 14080 "Hoppmål inte hittat" undertrycks. Det betyder att programbearbetningen i händelse av en resultatlös sökning av hoppmål inte avbryts, utan fortsätter med den på kommandot GOTOC följande programraden.	
<Sprungziel>:	Hoppmålsparameter Möjliga uppgifter är:	
	<Sprungmarke>:	Hoppmål är det i programmet satta hoppmärket med användardefinierat namn: <Sprungmarke>:
	<Satznummer>:	Hoppmål är ett huvud- eller sidoblocksnummer (t.ex.: 200, N300)
	Variabel av typ STRING:	Variabelt hoppmål. Variabeln står för ett hoppmärke eller ett blocksnummer.
IF:	Nyckelord för formulering av hoppvillkor. Hoppvillkor tillåter alla jämförelse- och logiska operationer (resultat: TRUE eller FALSE). Programhoppet utförs när resultatet av denna operation är TRUE.	

Märk**Hoppmärken (Labels)**

Hoppmärken står alltid i början av blocket. När ett programnummer finns står hoppmärket omedelbart efter blocknumret.

För beteckningen av hoppmärken gäller följande regler:

- Antal tecken:
 - minst 2
 - högst 32
- Tillåtna tecken är:
 - bokstäver
 - siffror
 - nedsänkta streck
- De första båda tecknen måste vara bokstäver eller nedsänkta streck.
- Efter namnet på hoppmärket följer en dubbelpunkt (":").

Randvillkor

- Hoppmål kan endast vara ett block med hoppmärke eller blocknummer som ligger inom programmet.
- En hoppanvisning utan hoppvillkor måste programmeras i ett separat block. Vid hoppanvisningar med hoppvillkor gäller inte denna inskränkning. Här kan flera hoppanvisningar formuleras i ett block.
- Vid program med hoppanvisningar utan hoppvillkor måste programslutet M2/M30 inte nödvändigtvis stå vid programslutet.

Exempel**Exempel 1: Hopp till hoppmärken**

Programkod	Kommentar
N10 ...	
N20 GOTOF Label_1	; Hopp i riktning programslut till ; hoppmärket "Label_1".
N30 ...	
N40 Label_0: R1=R2+R3	; Hoppmärke "Label_0" har satts.
N50 ...	
N60 Label_1:	; Hoppmärke "Label_1" har satts.
N70 ...	
N80 GOTOB Label_0	; Hopp i riktning programbörjan ; till hoppmärket "Label_0".
N90 ...	

Exempel 2: Indirekt hopp till blocknummer

Programkod	Kommentar
IF <Bedingung> == TRUE	
R10=100	; Tilldela hoppmål
ELSE	
R10=110	; Tilldela hoppmål
ENDIF	
; Hopp i riktning programslut till det block vars blocknummer står i R10.	
N10 GOTOF "N"<<R10	
...	
N90 ...	
N100 ...	; Hoppmål
N110 ...	
...	

Exempel 3: Hopp till variabelt hoppmål

Programkod	Kommentar
DEF STRING[20] ZIEL	
IF <Bedingung> == TRUE	
ZIEL = "Marke1"	; Tilldela hoppmål
ELSE	
ZIEL = "Marke2"	; Tilldela hoppmål
ENDIF	
; Hopp i riktning programslut till det variabla hoppmålet "Innehåll i ZIEL"	
GOTOF ZIEL	
Marke1: T="Bohrer1"	; Hoppmål 1
...	
Marke2: T="Bohrer2"	; Hoppmål 2
...	

Exempel 4: Hopp med hoppvillkor

Programkod	Kommentar
N40 R1=30 R2=60 R3=10 R4=11 R5=50 R6=20	; tillordning av begynnelsevärden
N41 LA1: G0 X=R2*COS(R1)+R5 Y=R2*SIN(R1)+R6	; Hoppmärke LA1
N42 R1=R1+R3 R4=R4-1	
; IF Sprungbedingung == TRUE	
; THEN Sprung in Richtung Programmanfang zur Sprungmarke LA1	
N43 IF R4>0 GOTOB LA1	
N44 M30	; Programslut

2.10.3 Programförgrening (CASE ... OF ... DEFAULT ...)

CASE-funktionen erbjuder möjligheten att kontrollera det aktuella värdet (typ: INT) för en variabel eller en räknefunktion och beroende av resultatet hoppa till olika ställen i programmet.

Syntax

```
CASE(<Ausdruck>) OF <Konstante_1> GOTOF <Sprungziel_1> <Konstante_2>
GOTOF <Sprungziel_2> ... DEFAULT GOTOF <Sprungziel_n>
```

Betydelse

CASE:	Hoppanvisning	
<Ausdruck>:	Variabel eller räknefunktion	
OF:	Nyckelord för formulering av de villkorliga programförgreningarna	
<Konstante_1>:	Första angivna konstanta värde för variabeln eller räknefunktionen	
	Typ:	INT
<Konstante_2>:	Andar angivna konstanta värde för variabeln eller räknefunktionen	
	Typ:	INT
DEFAULT:	För de fall i vilka variabeln eller räknefunktionen inte antar någon av de angivna konstanta värdena, kan ett hoppmål bestämmas med anvisningen DEFAULT. Observera: Om DEFAULT-anvisningen inte är programmerad, blir i dessa fall det block som följer på CASE-anvisningen till hoppmål.	
GOTOF:	Hoppanvisning med hoppmål i riktning programslut. I stället för GOTOF är också alla andra GOTO-kommandon programmerbara (se tema "Programhopp till hoppmärken").	
<Sprungziel_1>:	Det förgrenas till detta hoppmål när värdet för variabeln eller räknefunktionen motsvarar den första angivna konstanten. Hoppmålet kan anges på följande sätt:	
	<Sprungmarke>:	Hoppmål är det i programmet satta hoppmärke med användardefinierat namn: <Sprungmarke>:
	<Satznummer>:	Hoppmål är ett huvud- eller sidoblocknummer (t.ex.: 200, N300)
	Variabel av typ STRING:	Variabelt hoppmål. Variabeln står för ett hoppmärke eller ett blocknummer.
<Sprungziel_2>:	Det förgrenas till detta hoppmål när värdet för variabeln eller räknefunktionen motsvarar den andra angivna konstanten.	
<Sprungziel_n>:	Det förgrenas till detta hoppmål när värdet för variabeln inte antar något av de angivna konstanta värdena.	

Exempel

Programkod

```
...
N20 DEF INT VAR1 VAR2 VAR3
```

Programkod

```
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  
...
```

CASE-anvisningen från N30 definierar följande möjligheter till programförgreningar:

1. När värdet för räknefunktionen $VAR1+VAR2-VAR3 = 7$, hoppa då till blocket med hoppmärkesdefinition "Label_1" ($\rightarrow$ N40).
2. När värdet för räknefunktionen $VAR1+VAR2-VAR3 = 9$, hoppa då till blocket med hoppmärkesdefinition "Label_2" ($\rightarrow$ N50).
3. När värdet för räknefunktionen $VAR1+VAR2-VAR3$ varken uppgår till 7 eller 9, hoppa då till blocket med hoppmärkesdefinition "Label_3" ($\rightarrow$ N60).

2.11 Upprepning av programdel (REPEAT, REPEATB, ENDLABEL, P)

Programdelsupprepningen möjliggör upprepningen av redan skrivna programdelar inom ett program i valfri sammansättning.

De programrader resp. programområden som ska upprepas markeras med hoppmärken (Labels).

Märk

Hoppmärken (Labels)

Hoppmärken står alltid i början av blocket. När ett programnummer finns står hoppmärket omedelbart efter blocknumret.

För beteckningen av hoppmärken gäller följande regler:

- Antal tecken:
 - minst 2
 - högst 32
 - Tillåtna tecken är:
 - bokstäver
 - siffror
 - nedsänkta streck
 - De första båda tecknen måste vara bokstäver eller nedsänkta streck.
 - Efter namnet på hoppmärket följer en dubbelpunkt (":").
-

Syntax

1. Upprepa enstaka programrad:

```
<Sprungmarke>:  
...  
REPEATB <Sprungmarke> P=<n>  
...
```

2. Upprepa programområde mellan hoppmärke och REPEAT-anvisning:

```
<Sprungmarke>:  
...  
REPEAT <Sprungmarke> P=<n>  
...
```

3. Upprepa område mellan två hoppmärken:

```
<Start-Sprungmarke>: ...  
...  
<End-Sprungmarke>: ...  
...  
REPEAT <Start-Sprungmarke> <End-Sprungmarke> P=<n>
```

...

Märk

Det är inte möjligt att förbinda REPEAT-anvisningen med de båda hoppmärkena. Hittas <Start-Sprungmarke> före REPEAT-anvisningen och uppnås <End-Sprungmarke> inte före REPEAT-anvisningen, då görs upprepningen mellan <Start-Sprungmarke> och REPEAT-anvisning.

4. Upprepa område mellan hoppmarke och ENDLABEL:

```
<Sprungmarke>:
...
ENDLABEL: ...
...
REPEAT <Sprungmarke> P=<n>
...
```

Märk

Det är inte möjligt att förbinda REPEAT-anvisningen med <Sprungmarke> och ENDLABEL. Hittas <Sprungmarke> före REPEAT-anvisningen och uppnås ENDLABEL inte före REPEAT-anvisningen, då görs upprepningen mellan <Sprungmarke> och REPEAT-anvisning.

Betydelse

REPEATB:	Kommando för upprepning av en programrad
REPEAT:	Kommando för upprepning av ett programområde
<Sprungmarke>:	<Sprungmarke> kännetecknar: - den programrad som ska upprepas (vid REPEATB) resp. - början på det programområde som ska upprepas (vid REPEAT) Den med <Sprungmarke> kännetecknade programraden kan stå före eller efter REPEAT-/REPEATB-anvisningen. Det söks först i riktning programbörjan. Hittas hoppmärket inte i denna riktning så söks i riktning programslut. Undantag: När programområdet mellan hoppmarke och REPEAT-anvisning ska upprepas (se 2. under Syntax), då måste den med <Sprungmarke> kännetecknade programraden stå före REPEAT-anvisningen, eftersom det i detta fall endast söks i riktning programbörjan. Innehåller raden med <Sprungmarke> ytterligare anvisningar så utförs dessa vid varje upprepning.
ENDLABEL:	Nyckelord som markerar slutet på ett programområde som upprepas Innehåller raden med ENDLABEL ytterligare anvisningar så utförs dessa vid varje upprepning. ENDLABEL kan användas flera gånger i programmet.
P:	Adress för angivande av antalet upprepningar

<n>:	Antal programdelsupprepningar	
	Typ:	INT
	Den programdel som ska upprepas blir upprepad <n> gånger. Efter den sista upprepningen fortsätts programmet med den rad som följer på REPEAT-/REPEATB-raden. Observera: Är ingen P=<n> angiven, upprepas den programdel ska upprepas exakt en gång.	

Exempel

Exempel 1: Upprepa enstaka programrad

Programkod	Kommentar
N10 POSITION1: X10 Y20	
N20 POSITION2: CYCLE(0,,9,8)	; Positionscykel
N30 ...	
N40 REPEATB POSITION1 P=5	; Utför BLOCK N10 fem gånger.
N50 REPEATB POSITION2	; Utför block N20 en gång.
N60 ...	
N70 M30	

Exempel 2: Upprepa programområde mellan hoppmärke och REPEAT-anvisning

Programkod	Kommentar
N5 R10=15	
N10 Begin: R10=R10+1	; Bredd
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	; Utför område N10 till N70 fyra gånger.
N90 Z10	
N100 M30	

Exempel 3: Upprepa område mellan två hoppmärken

Programkod	Kommentar
N5 R10=15	
N10 Begin: R10=R10+1	; Bredd
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	
N50 X=-R10	
N60 Y=-R10	

2.11 Upprepning av programdel (REPEAT, REPEATB, ENDLABEL, P)

Programkod	Kommentar
N70 END: Z=10	
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	; Utför område N10 till N70 tre gånger.
N110 Z10	
N120 M30	

Exempel 4: Upprepa område mellan hoppmärke och ENDLABEL

Programkod	Kommentar
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	; Utför område N110 till N120 tre gånger.
N140 REPEAT BEGIN2 P=2	; Utför område N50 till N80 två gånger.
N150 M100	
N160 REPEAT BEGIN1 P=2	; Utför område N20 till N80 två gånger.
N170 Z10	
N180 X0 Y0	
N190 M30	

Exempel 5: Fräsbearbetning, bearbeta borrarposition med olika teknologier

Programkod	Kommentar
N10 ZENTRIERBOHRER()	; Växla in centrerborr.
N20 POS_1:	; Borrarpositioner 1
N30 X1 Y1	
N40 X2	
N50 Y2	
N60 X3 Y3	
N70 ENDLABEL:	
N80 POS_2:	; Borrarpositioner 2
N90 X10 Y5	
N100 X9 Y-5	
N110 X3 Y3	
N120 ENDLABEL:	

Programkod	Kommentar
N130 BOHRER()	; Växla borr och borrarcykel.
N140 GEWINDE(6)	; Växla in gängtapp M6 och gängcykel.
N150 REPEAT POS_1	; Upprepa programavsnitt från POS_1 en gång till ENDLABEL.
N160 BOHRER()	; Växla borr och borrarcykel.
N170 GEWINDE(8)	; Växla in gängtapp M8 och gängcykel.
N180 REPEAT POS_2	; Upprepa programavsnitt från POS_2 en gång till ENDLABEL.
N190 M30	

Ytterligare informationer

- Programdelsupprepning kan anropas i nivåer. Varje anrop belägger en underprogramnivå.
 - Är M17 eller RET programmerad under bearbetningen av en programdelsupprepning, så avbryts programdelsupprepningen. Programmet fortsätts med det block som följer på REPEAT-raden.
 - I den aktuella programindikeringen visas programdelsupprepningen som egen underprogramnivå.
 - Utlöses nivåavbrott under programdelsbearbetningen så fortsätts programmet efter anrop av programdelsbearbetningen.
- Exempel:

Programkod	Kommentar
N5 R10=15	
N10 BEGIN: R10=R10+1	; Bredd
N20 Z=10-R10	
N30 G1 X=R10 F200	
N40 Y=R10	; Nivåavbrott
N50 X=-R10	
N60 Y=-R10	
N70 END: Z10	
N80 Z10	
N90 CYCLE(10,20,30)	
N100 REPEAT BEGIN END P=3	
N120 Z10	; Fortsätta programbearbetning.
N130 M30	

- Kontrollstrukturer och programdelsupprepning kan användas kombinerade. Det bör dock inte bli några överlappningar. En programdelsupprepning bör ligga inom en kontrollstruktur-förgrening resp. en kontrollstruktur inom en programdelsupprepning.
 - Vid en blandning av hopp och programdelsupprepning genomarbetas blocken rent sekventiellt. Följer t.ex. ett hopp från en programdelsupprepning så bearbetas tills det programmerade programdelsslutet hittas.
- Exempel:

Programkod
N10 G1 F300 Z-10

Programkod

```
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
```

Märk

REPEAT-anvisningen bör stå bakom förflyttningsblocken.

2.12 Kontrollstrukturer

Styrningen arbetar genom NC-blocken standardmässigt i den programmerade ordningsföljden.

Denna ordningsföljd kan varieras genom programmeringen av alternativa programblock och programslingor. Programmeringen av dessa kontrollstrukturer sker med nyckelorden `IF`, `ELSE`, `ENDIF`, `LOOP`, `FOR`, `WHILE` och `REPEAT`.

OBSERVERA

Programmeringsfel

Kontrollstrukturer är möjliga endast inom ett programs anvisningsdel. Definitioner i programhuvudet kan inte utföras villkorligt eller upprepat.

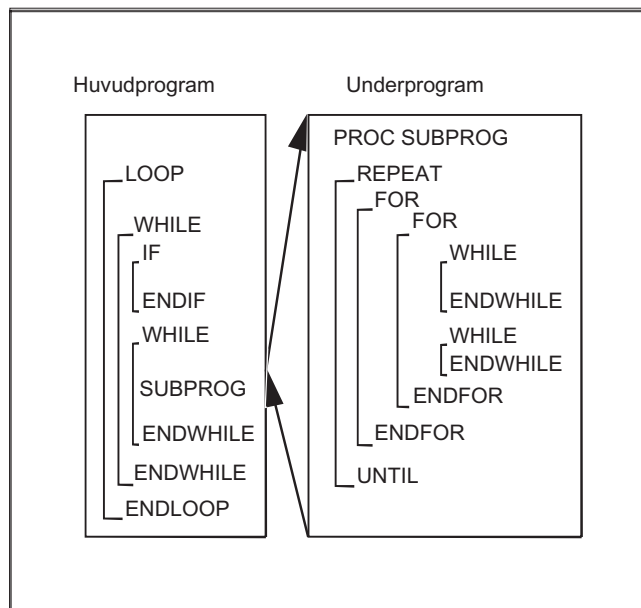
Nyckelord för kontrollstrukturer får liksom hoppmål inte överlagras med makros. En kontroll vid makrodefinitionen äger inte rum.

Verkan

En kontrollstruktur kan inte användas programövergripande.

Kapslingsdjup

Inom varje underprogramnivå är ett kapslingsdjup på upp till 16 kontrollstrukturer möjligt.



Körtidsbeteende

I standardmässigt aktiv interpreterdrift kan ett snabbare programförlopp uppnås genom användning av programhopp än med kontrollstrukturer.

I förkompilerade cykler finns ingen skillnad mellan programhopp och kontrollstrukturer.

Aktuell blockindikering vid programslingor

Utförs inom en programslinga endast förkörningsblock indikeras i den aktuella blockindikeringen det sista huvudkörningsblocket **före** programslingan.

För att t.ex. till diagnosändamål även de genomarbetade förkörningsblocken blir synliga i den aktuella blockindikeringen måste dekoderingsenkelblocket SBL2 aktiveras.

Litteratur

Funktionshandbok Grundfunktioner, kapitel: BAG, Kanal, Programdrift, Reset-beteende (K1)
> Enkelblock > Dekoderingsenkelblock SBL2 med implicit fördekoderingsstopp

Slipning utan huvudkörningsblock

Programmeras inget huvudkörningsblock inom en programslinga, bearbetas slingan så länge i förkörning tills slingvillkoret är uppfyllt.

Därigenom uppstår en hög nyttjandegrad och indikeringen kan påverkas negativt.

Som åtgärd erbjuder sig att infoga kommandot STOPRE eller en fördröjningstid G04 på 0 sekunder i slingan.

Randvillkor

- Block med kontrollstrukturelement kan inte gömmas.
- Hoppmärken (Labels) är inte tillåtna i block med kontrollstrukturelement.
- Kontrollstrukturer genomarbetas interpretativt. Vid identifiering av ett slut för en slinga söks under hänsynstagande till de därvid hittade kontrollstrukturerna efter början på slingan. Därför blir i interpreterdrift ett programs blockstruktur inte komplett kontrollerat.
- Principiellt rekommenderas att inte använda kontrollstrukturer och programhopp blandat.
- Vid förbearbetning av cykler kan den korrekta kapslingen av kontrollstrukturer kontrolleras.

2.12.1 Villkorlig anvisning och förgrening (IF, ELSE, ENDIF)

Villkorlig anvisning: IF - programblock - ENDIF

Vid en villkorlig anvisning utförs endast det mellan **IF** och **ENDIF** stående programblocket när villkoret är uppfyllt.

Förgrening: IF - programblock_1 - ELSE - programblock_2 - ENDIF

Vid en förgrening utförs alltid ett av två programblock.

Är villkoret uppfyllt utförs det mellan **IF** och **ELSE** stående programblock_1.

Är villkoret **inte** uppfyllt utförs det mellan **ELSE** och **ENDIF** stående programblock_2.

Syntax

Villkorlig anvisning

```
IF <Bedingung>
    Programmblock                               ; Utförande vid: <Bedingung> == TRUE
ENDIF
```

Förgrening

```
IF <Bedingung>
    Programmblock_1                             ; Utförande vid: <Bedingung> == TRUE
ELSE
    Programmblock_2                             ; Utförande vid: <Bedingung> == FALSE
ENDIF
```

Betydelse

IF:	Inleder den villkorliga anvisningen resp. förgrening.
ELSE:	Inleder det alternativa programblocket.
ENDIF:	Markerar slutet på den villkorliga anvisningen resp. förgrening.
<Bedingung>:	Logiskt uttryck, vars utvärdering ger TRUE eller FALSE.

Exempel: Verktygsväxlingsunderprogram

Programkod	Kommentar
PROC L6	Verktygsväxlingsrutin
N500 DEF INT TNR_AKTUELL	Variabel för aktivt T-nummer
N510 DEF INT TNR_VORWAHL	Variabel för förvalt T-nummer
	Fastställa aktuellt verktyg
N520 STOPRE	
N530 IF \$P_ISTEST	I programtest-drift blir ...
N540 TNR_AKTUELL = \$P_TOOLNO	... det "aktuella" verktyget
	läst ur programkontexten.
N550 ELSE	I annat fall ...
N560 TNR_AKTUELL = \$TC_MPP6[9998,1]	... läses spindelns verktyg.
N570 ENDIF	
N580 GETSELT(TNR_VORWAHL)	Läsa T-nummer för det förvalda
	verktyget på spindeln.
N590 IF TNR_AKTUELL <> TNR_VORWAHL	När det förvalda verktyget ännu
	inte är det aktuella verktyget,
	då ...
N600 G0 G40 G60 G90 SUPA X450 Y300 Z300 D0	... uppsöka verktygsväxlings-
	punkten ...
N610 M206	... och utförs verktygsväxlingen.
N620 ENDIF	
N630 M17	

2.12.2 Ändlös programslinga (LOOP, ENDLOOP)

Den ändlösa slingan hittar användning i ändlösa program. Vid slingslutet äger alltid återhoppet till slingans början rum.

Syntax

```

LOOP
...
ENDLOOP

```

Betydelse

LOOP:	Inleder den ändlösa slingan.
ENDLOOP:	Markerar slutet på slingan och förorsakar återhopp till slingans början.

Exempel

```

Programkod
...
LOOP
MSG ("Inget verktygsskär aktivt")
M0
STOPRE
ENDLOOP
...

```

2.12.3 Räkneslinga (FOR ... TO ..., ENDFOR)

Räkneslingan används när ett arbetsförlopp ska upprepas med ett fast antal genomgångar.

Syntax

```

FOR <Variable> = <Anfangswert> TO <Endwert>
...
ENDFOR

```

Betydelse

FOR:	Inleder räkneslingan.
ENDFOR:	Markerar slutet på slingan och förorsakar återhopp till slingans början, så länge som slutvärdet för räkningen ännu inte har uppnåtts.

<Variable>:	Räknevariabel som räknas upp från begynnelse- till slutvärdet och som ökar med värdet "1" för varje genomgång.	
	Typ	INT eller REAL Observera: Typen REAL väljs när t.ex. R-parameter programmeras för en räkneslinga. Är räkneslingan av typ REAL, avrundas dess värde till ett heltaligt värde.
<Anfangswert>:	Begynnelsevärde för räkningen Villkor: Begynnelsevärdet måste vara mindre än slutvärdet.	
<Endwert>:	Slutvärde för räkningen	

Exempel

Exempel 1: INTEGER-variabel eller R-parameter som räknevariabel

INTEGER-variabel som räknevariabel:

Programkod	Kommentar
DEF INT iVARIABLE1 R10=R12-R20*R1 R11=6 FOR iVARIABLE1= R10 TO R11 R20=R21*R22+R33 ENDFOR M30	; Räknevariabel = INTEGER-variabel

R-parameter som räknevariabel:

Programkod	Kommentar
R11=6 FOR R10=R12-R20*R1 TO R11 R20=R21*R22+R33 ENDFOR M30	; Räknevariabel = R-parameter (Realvariabel)

Exempel 2: Tillverkning av ett fast detaljstycktal

Programkod	Kommentar
DEF INT STUECKZAHL FOR STUECKZAHL = 0 TO 100 G01 ... ENDFOR M30	; Definierar variabel av typ INT med namnet "STUECKZAHL". ; Inleder räkneslingan. Variabeln "STUECKZAHL" räknas upp från begynnelsevärdet "0" till slutvärdet "100". ; Slut på räkneslingan.

2.12.4 Programslinga med villkor vid slingans början (WHILE, ENDWHILE)

Vid en WHILE-slinga står villkoret i början av slingan. Så länge som villkoret är uppfyllt kommer WHILE-slingan att köras.

Syntax

```
WHILE <Bedingung>
...
ENDWHILE
```

Betydelse

WHILE:	Inleder programslingan.
ENDWHILE:	Markerar slutet på slingan och förorsakar återhopp till slingans början.
<Bedingung>:	Villkor som måste vara uppfyllt för att WHILE-slingan ska köras.

Exempel

Programkod	Kommentar
...	
WHILE \$AA_IW[BOHRACHSE] > -10	; Anrop av WHILE-slingan under följande villkor: det aktuella WKS-börvärdet för borraraxeln måste vara större än -10.
G1 G91 F250 AX[BOHRACHSE] = -1	
ENDWHILE	
...	

2.12.5 Programslinga med villkor vid slingans slut (REPEAT, UNTIL)

Vid en REPEAT-slinga står villkoret i slutet av slingan. REPEAT-slingan körs en gång och upprepas till villkoret är uppfyllt.

Syntax

```
REPEAT
...
UNTIL <Bedingung>
```

Betydelse

REPEAT:	Inleder programslingan.
UNTIL:	Markerar slutet på slingan och förorsakar återhopp till slingans början.
<Bedingung>:	Villkor som måste vara uppfyllt för att REPEAT-slingan inte längre ska köras.

Exempel

Programkod	Kommentar
...	
REPEAT	; Anrop av REPEAT-slingan.
...	
UNTIL ...	; Kontroll om villkoret är uppfyllt.
...	

2.12.6 Programexempel med kapslade kontrollstrukturer

Programkod	Kommentar
LOOP	
IF NOT \$P_SEARCH	; IF ingen blocksökning
G1 G90 X0 Z10 F1000	
WHILE \$AA_IM[X] <= 100	; WHILE (börvärde X-axel <= 100)
G1 G91 X10 F500	; Borrbild
Z-5 F100	
Z5	
ENDWHILE	
ELSE	; ELSE blocksökning
MSG("Under blocksökning borrar inte")	
ENDIF	; ENDIF
\$A_OUT[1] = 1	; Nästa borrarplatta
G4 F2	
ENDLOOP	
M30	

2.13 Koordineringskommandon (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)

En kanal i NC kan principiellt genomarbeta det program som startats i den, oberoende av andra kanaler i dess driftsättgrupp (BAG). Men deltar samtidigt flera program i flera kanaler i BAG i tillverkningen av ett arbetsstycke, måste programförloppen i de olika kanalerna koordineras med de följande koordineringskommandona.

Förutsättning

Alla i programkoordineringen deltagande kanalerna måste höra till **samma** driftsättgrupp (BAG):

MD10010 \$MC_ASSIGN_CHAN_TO_MODE_GROUP[<Kanal>] = <BAG-Nummer>

Kanalnamn i stället för kanalnummer

Som parameter i de fördefinierade procedurerna för programkoordineringen kan i stället för kanalnummer också de i MD20000 \$MC_CHAN_NAME[<Kanalindex>] införda kanalnamnen användas. Användningen av kanalnamnen i NC-programmen måste först frigivas:

MD10280 \$MN_PROG_FUNCTION_MASK, Bit 1 = TRUE

Syntax

```
INIT(<ChanNr>, <Prog>, <AckMode>)
START(<ChanNr>, <ChanNr>, ...)
WAITM(<MarkNr>, <ChanNr>, <ChanNr>, ...)
WAITE(<ChanNr>, <ChanNr>, ...)
WAITMC(<MarkNr>, <ChanNr>, <ChanNr>, ...)
SETM(<MarkNr>, <MarkNr>, ...)
CLEARM(<MarkNr>, <MarkNr>, ...)
```

Betydelse

INIT():	Fördefinierad procedur för val av det NC-program som ska genomarbetas i den angivna kanalen
START():	Fördefinierad procedur för att starta det i respektive kanal valda programmet
WAITM():	Fördefinierad procedur för att vänta på att ett väntemärke uppnås i de angivna kanalerna I den egna kanalen sätts det angivna väntemärket av WAITM. Det föregående blocket avslutas med precisionsstopp. Väntemärket raderas efter synkronisering. Samtidigt kan max. 10 märken sättas per kanal.
WAITE():	Fördefinierad procedur för att vänta på programslutet i en eller flera andra kanaler
WAITMC(): ¹⁾	Fördefinierad procedur för att vänta på att ett väntemärke uppnås i de angivna kanalerna I motsats till WAITM inleds bromsningen av axlarna till precisionsstopp, endast när de andra kanalerna ännu inte har nått väntemärket.
SETM(): ¹⁾	Fördefinierad procedur för att sätta ett eller flera väntemärken för kanalkoordineringen Bearbetningen i den egna kanalen påverkas inte därav. SETM behåller sin giltighet utöver kanal-Reset och NC-start.

CLEARM () : ¹⁾	Fördefinierad procedur för att radera ett eller flera väntemärken i kanalkoordineringen Bearbetningen i den egna kanalen påverkas inte därav. CLEARM () raderar alla väntemärken i kanalen. CLEARM (0) raderar enda väntemärket "0". CLEARM behåller sin giltighet utöver kanal-Reset och NC-start.		
<ChanNr>:	Kanalnummer Numret för den egna kanalen måste inte anges.		
	Typ:	INT	
<Prog>:	Absolut eller relativ sökvägsuppgift (optional) + programnamn		
	Typ:	STRING	
	För sökvägsuppgift se: Litteratur Programmeringshandbok Arbetsförberedelse; Kapitel "Fil- och programförvaltning" > "Programminne" > "Adressering av filer i programminnet" (Sida 213)		
<AckMode>:	Kvitteringsmode (option)		
	Typ:	CHAR	
	Vär- den:	"N"	Utan kvittering Programbearbetningen fortsätts efter sändning av kommandot. Avsän- daren informeras inte när kommandot inte kan utföras med framgång.
		"S"	Synkron kvittering Programgenomarbetningen stoppas tills mottagarkomponenten har kvitterat kommandot. Vid positiv kvittering genomarbetas nästa kom- mando. Vid negativ kvittering matas ett felmeddelande ut.
<MarkNr>:	Nummer på väntemärket Observera I ett system med flera kanaler står maximalt 100 väntemärken till förfogande (vänte- märke 0 ... 99). I ett system med en kanal står endast väntemärket 0 till förfogande.		

1) För användarspecifik kommunikation och / eller koordination av kanaler kan väntemärken, med hjälp av SETM / CLEARM också användas utan användning av det villkorliga väntekommandot WAITMC. Väntemärkena bibehåller därvid sina värden utöver kanal-Reset och NC-start.

Exempel

START med hjälp av kanalnamn från MD20000

• Parametrering

```
MD10280 $MN_PROG_FUNCTION_MASK, Bit 1 = TRUE
$MC_CHAN_NAME[ 0 ] = "BEARBEITUNG" ; Namn på kanal 1
$MC_CHAN_NAME[ 1 ] = "ZUFUEHRUNG" ; Namn på kanal 2
```

• Programmering

Programkod	Kommentar
START(BEARBEITUNG)	; Start av kanal 1
START(ZUFUEHRUNG)	; Start av kanal 2

2.13 Koordineringskommandon (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)

START via lokala "Kanalnamn" och användarvariabel

Programkod	Kommentar
DEF INT MASCHINE = 1	; Definition användarvariabel för kanal 1
DEF INT LADER = 2	; Definition användarvariabel för kanal 2
...	
START(MASCHINE)	; Start av kanal 1
START(LADER)	; Start av kanal 2

START via lokala "Kanalnamn", användarvariabel och parametrerade kanalnamn

Programkod	Kommentar
DEF INT chanNo1	; Definition användarvariabel för kanal 1
DEF INT chanNo2	; Definition användarvariabel för kanal 2
chanNo1 = CHAN_1	; Tillordning parametrerat kanalnamn kanal 1
chanNo2 = CHAN_2	; Tillordning parametrerat kanalnamn kanal 2
...	
START(chanNo1)	; Start av kanal 1
START(chanNo2)	; Start av kanal 2

INIT-kommando med absolut sökvägsuppgift

Val av program /_N_MPF_DIR/_N_ABSPAN1_MPF i kanal 2.

Programkod

```
INIT(2, "/_N_WKS_DIR/_N_WELLE1_WPD/_N_ABSPAN1_MPF")
```

INIT-kommando med programnamn

Val av programmet med namnet "MYPROG". Styrningen söker programmet med hjälp av sökvägen.

Programkod

```
INIT(2, "MYPROG")
```

Programkoordinering med WAITM

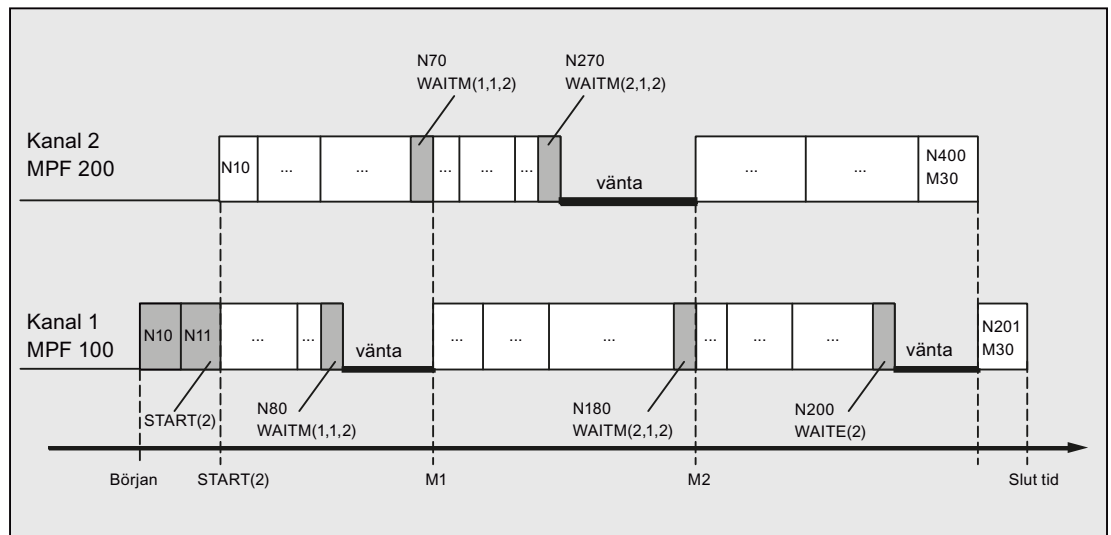
- Kanal 1: Programmet /_N_MPF_DIR/_N_MPF100_MPF har redan valts och startats.

Programkod	Kommentar
	; Program MPF100
N10 INIT(2,"MPF200","N")	; Val program MPF200, kanal 2
N11 START(2)	; Start av kanal 2
...	
N80 WAITM(1,1,2)	; Vänta på WAIT-märke 1 i kanal 1 och 2
N81 ...	; Kanal 1, N81 och kanal 2, N71 börjas ; synkront
...	
N180 WAITM(2,1,2)	; Vänta på WAIT-märke 2 i kanal 1 och 2
N181 ...	; Kanal 1, N181 och kanal 2, N271 börjas ; synkront
...	
N200 WAITE(2)	; Vänta på programslut i kanal 2
N201 ...	; N201 börjas först efter programslut av ; MPF200 i kanal 2
N201 M30	; Programslut kanal 1

- Kanal 2: I kanal 1 väljs och startas via block N10 och N20 programmet MPF200_MPF för kanal 2.

Programkod	Kommentar
;\$PATH=/_N_MPF_DIR	; Program MPF200
...	
N70 WAITM(1,1,2)	Vänta på WAIT-märke 1 i kanal 1 och 2
N71 ...	; Kanal 1, N81 och kanal 2, N71 börjas ; synkront
...	
N270 WAITM(2,1,2)	Vänta på WAIT-märke 2 i kanal 1 och 2
N271 ...	; Kanal 1, N181 och kanal 2, N271 börjas ; synkront
...	
N400 M30	Programslut kanal 2

2.13 Koordineringskommandon (INIT, START, WAITM, WAITMC, WAITE, SETM, CLEARM)



Randvillkor

Ej synkron början av genomarbetningen av följdblock efter WAIT-märken

Vid en kanalkoordinering med hjälp av WAIT-märken kan en ej synkron början av genomarbetningen av följdblocken komma till stånd. Detta beteende uppträder, när en aktion utlöses omedelbart innan det gemensamma WAIT-märket uppnås i en av de kanaler som ska synkroniseras, vilket har restvägsradering med implicit repositionering (REPOSA) i denna till följd.

Antagande: Aktuell axeltillordning i kanalerna 1 och 2

- Kanal 1: Axlar X1 och U
- Kanal 2: Axel X2

Tabell 2-2 Tidsmässigt förlopp i kanalerna 1 och 2

Kanal 1	Kanal 2	Beskrivning
...	...	Valfri bearbetning i kanal 1 och 2
N100 WAITM(20,1,2)		Kanal 1: uppnår WAIT-märket och väntar på synkronisering med kanal 2
<i>Påbörjan av GETD(U) bearbetning:</i>	N200 GETD(U)	Kanal 2: anropar axel U från kanal 1 Kanal 1: Bearbetning av GET(U) i bakgrunden
• Axelbyte	N210 WAITM(20,1,2)	Kanal 2: uppnår WAIT-märket. ⇒ Synkroniseringen av kanalerna 1 och 2 är därmed avslutad
• Restvägsradering	N220 G0 X2=100	Kanal 2: Början av genomarbetningen av N220
• REPOSA		
<i>Slut</i>		
N110 G0 X1=100		Kanal 1: Tidsförskjuten början av genomarbetningen av N110

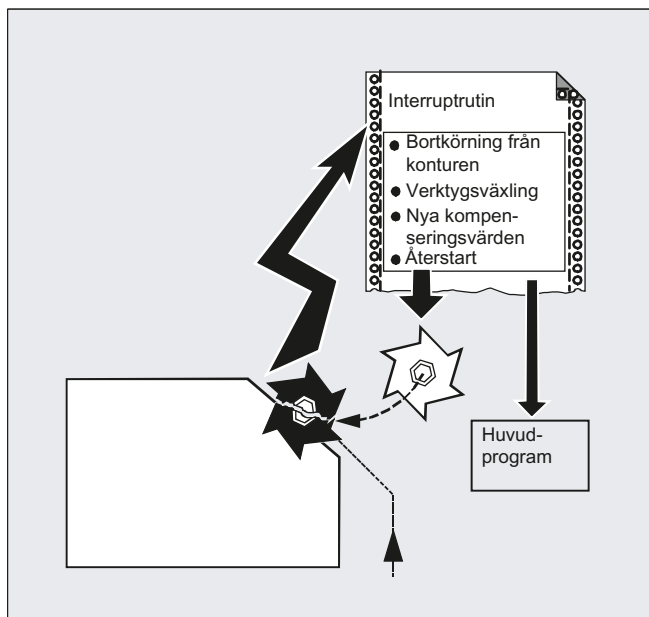
2.14 Interruptrutin (ASUP)

2.14.1 Funktion hos en interruptrutin

Märk

De i den följande beskrivningen omväxlande förekommande begreppen "Asynkront underprogram (ASUP)" och "Interruptrutin" betecknar samma funktionalitet.

Funktionen hos en interruptrutin ska förtydligas med hjälp av ett typiskt exempel:



Under bearbetningen bryts verktyget av. Härigenom utlöses en signal som stoppar det pågående bearbetningsförloppet och samtidigt startar ett underprogram – den så kallade interruptrutinen. I detta underprogram står alla anvisningar som ska utföras i detta fall.

Är underprogrammet genomarbetat (och härigenom driftberedskapen upprättad), hoppar styrningen tillbaka till huvudprogrammet och fortsätter bearbetningen – allt efter `REPOS`-kommando – vid avbrottsstället (se " Körning tillbaka till konturen (Sida 525) ").

⚠ SE UPP

Kollisionsfara

Om inget `REPOS`-kommando är programmerat i underprogrammet, då positioneras på ändpunkten till det block som följer på det avbrutna blocket.

Litteratur

Funktionshandbok Grundfunktioner; BAG, kanal, programdrift, Reset-beteende (K1), Kapitel: "Asynkront underprogram (ASUPs), Interruptrutiner"

2.14.2 Upprätta interruptrutin

Upprätta interruptrutin som underprogram

Interruptrutinen betecknas vid definitionen som ett underprogram.

Exempel:

Programkod	Kommentar
PROC ABHEB_Z	; Programnamn "ABHEB_Z"
N10 ...	; Därefter följer NC-blocken.
...	
N50 M17	; Till slut programslut och retur till huvudprogrammet.

Spara (SAVE) modalt G-kommando

Interruptrutinen kan vid definitionen betecknas med `SAVE`.

Attributet `SAVE` förorsakar att de före anropet av interruptrutinen aktiva modala G-kommandona säkras och när interruptrutinen är slut åter reaktiveras (se "Underprogram med `SAVE`-mekanism (`SAVE`) (Sida 164)").

Därigenom är det möjligt att fortsätta bearbetningen vid avbrottsstället när interruptrutinen är färdig.

Exempel:

Programkod
PROC ABHEB_Z SAVE
N10 ...
...
N50 M17

Tillordna ytterligare interruptrutiner (SETINT)

Inom interruptrutinen kan `SETINT`-anvisningar (se "Tillordna och starta interruptrutiner (`SETINT`)" (Sida 126)) programmeras och härigenom ytterligare interruptrutiner skarpkopplas. Utlösningen sker först genom ingången.

Litteratur

För ytterligare informationer för upprättande av underprogram se Kapitel "Underprogramteknik, makroteknik".

2.14.3 Tillordna och starta interruptrutin (SETINT, PRIO, BLSYNC)

Styrningen förfogar över flera snabba ingångar (ingång 1 ... 8), som alla utlöser en Interrupt (1 ... 8). Varje Interrupt kan via kommandot `SETINT` tilldelas en prioritet och en Interrupt-rutin. Utlöses Interrupten genom att sätta den snabba ingången, avbryts den aktuella bearbetningen i kanalen och Interrupt-rutinen startas.

Interrupt-prioritet

Tilldelas i ett detaljprogram flera ingångar Interrupts, måste Interruptsen tilldelas olika prioriteter.

En Interrupt kan tilldelas ett prioritetsvärde från 1 ... 128. Prioritetsvärde 1 motsvarar den högsta, 128 den lägsta prioriteten.

Syntax

```
SETINT (<n>) <NAME>
SETINT (<n>) PRIO=<Wert> <NAME>
SETINT (<n>) PRIO=<Wert> <NAME> BLSYNC
SETINT (<n>) PRIO=<Wert> <NAME> LIFTFAST
```

Betydelse

SETINT (<n>):	Till interruptsignal <n> tillordnas NC-programm (ASUP) <Name>. Den tillordnade Interruptrutinen startas, så snart interruptsignal <n> == 1 identifierats. Observera: Tillordnas en interruptsignal <n> en annan Interrupt-rutin, blir den tidigare tillordningen överksam.	
<n>:	Nummer för interruptsignalen	
	Typ:	INT
	Värdeområde:	1 ... 32
PRIO= :	Prioritet för Interrupts (tillval)	
<Wert>:	(optional) prioritetsvärde	
	Typ:	INT
	Värdeområde:	1 ... 128 (1 ⇒ högsta prioritet)
<NAME>:	Namn på NC-programmet (ASUP)	
BLSYNC:	(optional) BLSYNC har till följd att efter utlösningen av Interrupten det först väntas tills det aktuella blocket har genomarbetats. Först därefter utförs Interrupt-rutinen.	
LIFTFAST:	(optional) LIFTFAST har till följd att efter utlösningen av Interrupten det först följer en snabblyftning (se Kapitel "Snabblyftning från konturen (SETINT LIFTFAST, ALF) (Sida 129)"). Först därefter utförs Interrupt-rutinen.	

Randvillkor

Interrupt-regler

1. För varje Interrupt, som inte genast kan genomarbetas, eller aktuellt redan är under bearbetning, sparas ett ytterligare Interrupt-krav. Interrupt-krav därutöver går förlorade för denna Interrupt.
2. Bearbetas aktuellt en Interrupt och det utlöses en ytterligare Interrupt med högre prioritet, stoppar denna den lägre prioriterade Interrupten. När den högre prioriterade Interrupten avslutats, fortsätts den lägre prioriterade Interrupten. Inträffar under det att den högre prioriterade Interrupten bearbetas, ytterligare krav för den lägre prioriterade Interrupten, sparas ett krav. Ytterligare går förlorade.
3. Bearbetas aktuellt en Interrupt och det utlöses en ytterligare Interrupt med högre prioritet, stoppar denna den lägre prioriterade Interrupten. Den högre prioriterade Interrupten bearbetas. Utlöses å andra sidan en högre prioriterad Interrupt, stoppas den aktuella Interrupten och den högre prioriterade Interrupten bearbetas. Maximalt är sex aktiva Interrupt-nivåer möjliga. En aktuellt bearbetad Interrupt-nivå och fem Interrupt-nivåer som väntar. För varje aktiv Interrupt-nivå sparas maximalt ett ytterligare Interrupt-krav. Alla ytterligare Interrupt-krav går förlorade. Likaså går Interrupt-krav förlorade, när dessa krävs för ytterligare Interrupt-nivåer (Interrupt-nivå ≥ 7).

Exempel

Exempel 1: Tillordna interruptrutiner och fastlägga prioritet

Programkod	Kommentar
N20 SETINT(3) PRIO=1 ABHEB_Z	; IF Eingang 3 == 1
	; THEN Interruptroutine "ABHEB_Z" startas
N30 SETINT(2) PRIO=2 ABHEB_X	; IF Eingang 2 == 1
	; THEN Interruptroutine "ABHEB_X" startas

Interruptrutinerna genomarbetas i prioritetsvärdenas ordningsföljd efter varandra, när ingångarna väntar samtidigt: först "ABHEB_Z", sedan "ABHEB_X".

Exempel 2: Tillordna interruptrutin på nytt

Programkod	Kommentar
N20 SETINT(3) PRIO=2 ABHEB_Z	; IF Eingang 3 == 1
	; THEN Interruptroutine "ABHEB_Z" startas
...	
N80 SETINT(3) PRIO=1 ABHEB_X	; IF Eingang 3 == 1
	; THEN Interruptroutine "ABHEB_X" startas

2.14.4 Inaktivera/aktivera tillordning för en interruptrutin (DISABLE, ENABLE)

En SETINT-anvisning kan inaktiveras med `DISABLE` och åter aktiveras med `ENABLE` utan att tillordningen ingång → interruptrutin går förlorad.

2.14 Interruptrutin (ASUP)

Syntax

DISABLE (<n>)
ENABLE (<n>)

Betydelse

DISABLE (<n>):	Kommando: Inaktivering av interruptrutin-tillordningen av ingång <n>	
ENABLE (<n>):	Kommando: Reaktivering av interruptrutin-tillordningen av ingång <n>	
<n>:	Parameter: Nummer för interruptsignalen	
	Typ:	INT
	Värdeområde:	1 ... 32

Exempel

Programkod	Kommentar
N20 SETINT(3) PRIO=1 ABHEB_Z	; När ingång 3 kopplar då ska
	; interruptrutinen "ABHEB_Z" starta.
...	
N90 DISABLE(3)	; SETINT-anvisningen från N20 inaktiveras.
...	
N130 ENABLE(3)	; SETINT-anvisningen från N20 aktiveras åter.
...	

2.14.5 Radera tillordning av en interruptrutin (CLRINT)

En med SETINT definierad tillordning av en interruptsignal till ett NC-program (ASUP) kan raderas med CLRINT.

Syntax

CLRINT (<n>)

Betydelse

CLRINT (<n>):	Kommando: Radering av tillordningen av interruptsignalen <n> till med SETINT definierat NC-program (ASUP) <n>	
<n>:	Parameter: Nummer för interruptsignalen	
	Typ:	INT
	Värdeområde:	1 ... 32

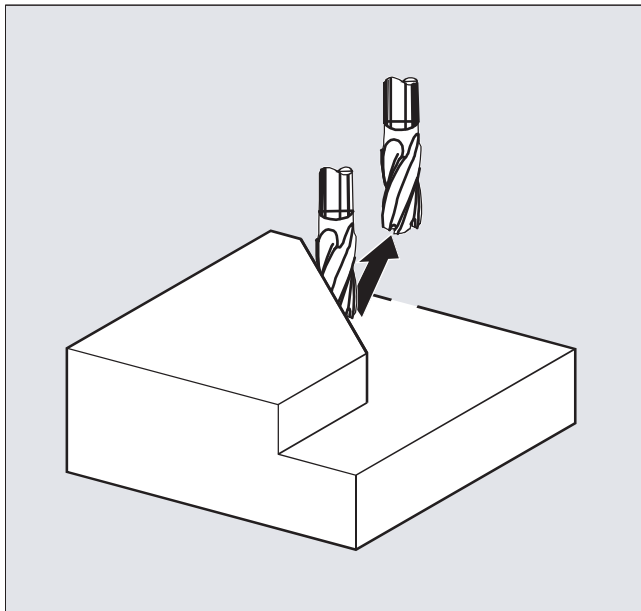
Exempel

Programkod	Kommentar
N20 SETINT(3) PRIO=2 ABHEB_Z	

Programkod	Kommentar
...	
N50 CLRINT(3)	; Tillordningen mellan ingång "3" och interruptrutinen "ABHEB_Z" är raderad.

2.14.6 Snabblyftning från konturen (SETINT LIFTFAST, ALF)

Vid en SETINT-anvisning med LIFTFAST körs vid kopplingen av ingången verktyget bort från arbetsstyckskonturen genom snabb lyftning.



Det ytterligare förloppet är beroende av om SETINT-anvisningen förutom LIFTFAST innehåller en interruptrutin:

Med interruptrutin:	Efter snabblyftningen utförs interruptrutinen.
Utan interruptrutin:	Bearbetningen stoppas efter snabblyftningen med larm.

Syntax

```
SETINT(<n>) PRIO=1 LIFTFAST
SETINT(<n>) PRIO=1 <NAME> LIFTFAST
```

Betydelse

SETINT(<n>):	Kommando: Tillordna ingång <n> till en interruptrutin. Den tillordnade interruptrutinen startar när ingången <n> kopplar.	
<n>:	Parameter: Nummer på ingången	
	Typ:	INT
	Värdeområde:	1 ... 8

2.14 Interruptrutin (ASUP)

PRIO= :	Fastläggande av prioritet	
<Wert>:	Prioritetsvärde	
	Värdeområde:	1 ... 128
	Prioritet 1 motsvarar den högsta prioriteten.	
<NAME>:	Namnet på det underprogram (interruptrutin) som ska genomarbetas.	
LIFTFAST:	Kommando: Snabblyftning från konturen	
ALF=... :	Kommando: Programmerbar förflytningsriktning (står i rörelseblock) Till programmeringsmöjligheterna med ALF se tema "Förflytningsriktning vid snabblyftning från konturen (Sida 131)".	

Randvillkor

Beteende vid aktiv frame med speling

Vid bestämmandet av lyftningsriktningen kontrolleras om en frame med spegling är aktiv. I detta fall bytes höger och vänster vid lyftningsriktningen relaterad till tangentriktningen. Riktningssandelarna i verktygsriktningen speglas inte. Detta beteende aktiveras genom MD-inställning:

MD21202 \$MC_LIFTFAST_WITH_MIRROR = TRUE

Exempel

Ett avbrutet verktyg ska automatiskt ersättas av ett tvillingverktyg. Bearbetningen fortsätts då med det nya verktyget.

Huvudprogram:

Huvudprogram	Kommentar
N10 SETINT(1) PRIO=1 W_WECHS LIFTFAST	; När ingång 1 kopplar körs verktyget bort från konturen med snabblyftning (Code Nr. 7 för verktygsradiekompensering G41). Då genomarbetas interruptrutinen "W_WECHS".
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	

Underprogram:

Underprogram	Kommentar
PROC W_WECHS SAVE	; Underprogram med sparande av det aktuella driftstillståndet
N10 G0 Z100 M5	; Verktygsväxlingsposition, spindelstopp
N20 T11 M6 D1 G41	; Växla verktyg
N30 REPOS L RMBBL M3	; Återuppsöka kontur och återhopp till huvudprogrammet (programmeras i ett block)

2.14.7 Förflyttningsriktning vid snabblyftning från konturen**Återgångsrörelse**

Planet för återgångsrörelsen bestäms av följande G-kommandon:

- **LFTXT**
Planet för återgångsrörelsen bestäms av bantangenten och verktygsriktningen (standardinställning).
- **LFWP**
Planet för återgångsrörelsen är det aktiva arbetsplanet som väljs med G-kommandona G17, G18 eller G19. Riktningen för återgångsrörelsen är oberoende av bantangenten. Därmed är en axelparallell snabblyftning programmerbar.
- **LFPOS**
Återgång för den med POLFMASK / POLFMLIN bekantgjorda axeln till den med POLF programmerade absoluta axelpositionen.
ALF har inget inflytande på lyftningsriktningen för flera axlar samt för flera axlar i linjärt sammanhang.
Litteratur:
Programmeringshandbok Grunder; Kapitel: "Snabbåtergång under gängskärning"

Programmerbar förflyttningsriktning (ALF=...)

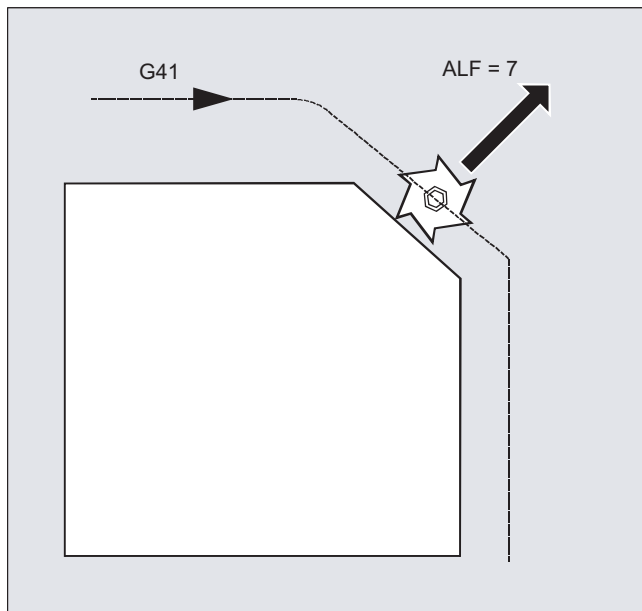
I planet för återgångsrörelsen programmeras riktningen i diskreta steg på 45 grader med ALF.

De möjliga förflyttningsriktningarna är sparade i styrningen under specifika kodnummer och kan anropas under dessa nummer.

Exempel:

Programkod
N10 SETINT(2) PRIO=1 ABHEB_Z LIFTFAST
ALF=7

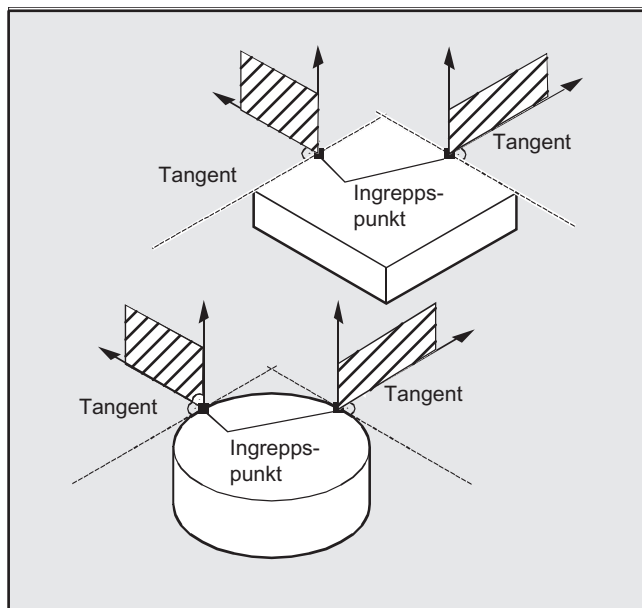
Verktyget kör vid tillkopplad G41 (bearbetningsriktning vänster om konturen) vinkelrätt bort från konturen.



Referensplan för beskrivningen av förflytningsriktningar vid LFTXT

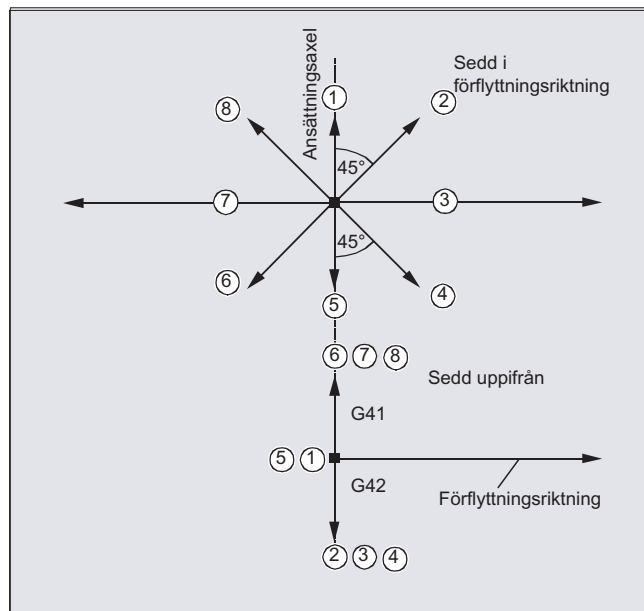
I verktygets ingreppspunkt vid den programmerade konturen spänns ett plan upp som tjänar som referens för uppgiften för lyftningsrörelsen med det motsvarande kodnumret.

Referensplanet spänns upp från verktygslängdaxeln (ansättningsriktning) och en vektor som står mot detta och vinkelrätt mot tangenten i verktygets ingreppspunkt vid konturen.



Code-nummer med förflytningsriktningar vid LFTXT

Utgående från referensplanet hittar du i följande bild kodnumren med förflytningsriktningar.



För $ALF=1$ är återgången i verktygsriktningen fastlagd.

Med $ALF=0$ är funktionen "Snabblyftning" fränkopplad.

⚠ SE UPP

Kollisionsfara

Vid tillkopplad verktygsradiekompensering bör:

- vid G41 kodningen 2, 3, 4
- vid G42 kodningen 6, 7, 8

inte användas eftersom i dessa fall verktyget åker till konturen och skulle kollidera med arbetsstycket.

Code-nummer med förflyttningsriktningar vid LFWP

Vid LFWP bildas riktningen i arbetsplanet enligt följande tillordning:

- G17: X/Y-plan
 $ALF=1$: Återgång i X-riktning
 $ALF=3$: Återgång i Y-riktning
- G18: Z/X-plan
 $ALF=1$: Återgång i Z-riktning
 $ALF=3$: Återgång i X-riktning
- G19: Y/Z-plan
 $ALF=1$: Återgång i Y-riktning
 $ALF=3$: Återgång i Z-riktning

2.14.8 Rörelseförlopp vid interruptrutiner

Interruptrutin utan LIFTFAST

Axelrörelserna bromsas på banan till stillestånd. Sedan startar interruptrutinen.

Stilleståndspositionen sparas som avbrottsposition och startas vid `REPOS` med `RMIBL` i slutet av interruptrutinen.

Interruptrutin med LIFTFAST

Axelrörelserna bromsas på banan. Samtidigt utförs `LIFTFAST`-rörelsen som överlagrad rörelse. När banrörelsen och `LIFTFAST`-rörelsen står stilla, startas interruptrutinen.

Som avbrottsposition sparas den position på konturen vid vilken `LIFTFAST`-rörelsen startade och banan därigenom lämnades.

Interruptrutinen förhåller sig med `LIFTFAST` och `ALF=0` identiskt som vid interruptrutin utan `LIFTFAST`.

Märk

Det belopp med vilket geometriaxlanra körs bort vid snabblyftning från konturen kan ställas in via ett maskindatum.

2.15 Axelbyte, spindelbyte (RELEASE, GET, GETD)

En eller flera axlar resp. spindlar kan alltid interpoleras endast i en kanal. Måste en axel arbeta växelvis i två olika kanaler (t.ex. palettväxling) så måste den först frigivas i den aktuella kanalen och sedan övertas i den andra kanalen. Axeln byts mellan kanalerna.

Axelbytesutvidgningar

En axel/spindel kan med fördekoderingsstopp och synkronisering mellan förkörning och huvudkörning eller alternativt också utan fördekoderingsstopp bytas. Dessutom är ett axelbyte också möjligt via

- axelcontainer-rotation AXCTSWE resp. AXCTWED med hjälp av implicita GET/GETD.
- frame med rotation när denna axel är förbunden häröver med andra axlar.
- Synkronaktioner, se Rörelsesynkronaktioner, "Axelbyte RELEASE, GET".

Maskintillverkare

Följ anvisningarna från maskintillverkaren. Via projekterbara maskindata måste en axel vara entydigt definierad i alla kanaler för axelbyte och axelbytesbeteendet kan också ställas in föränderligt via maskindata.

Syntax

RELEASE (Achssname, Achssname, ...) eller RELEASE (S1)

GET (Achssname, Achssname, ...) eller GET (S2)

GETD (Achssname, Achssname, ...) eller GETD (S3)

Med GETD (GET Directly) hämtas en axel direkt från en annan kanal. Det betyder att till detta GETD inget passande RELEASE måste vara programmerat i en annan kanal. Men det betyder också att en annan kanalkommunikation nu måste byggas upp (t.ex. Waitmärken).

Betydelse

RELEASE (Achssname, Achssname, ...):	Frigivning av ax(eln)(larna)
GET (Achssname, Achssname, ...):	Övertagande av ax(eln)(larna)
GETD (Achssname, Achssname, ...):	Direkt övertagande av ax(eln)(larna)
Achssname:	Axeltillordning i systemet: AX1, AX2, ... eller angivande av maskinaxelnamnen
RELEASE (S1):	Frigivning av spindeln S1, S2, ...
GET (S2):	Övertagande av spindeln S1, S2, ...
GETD (S3):	Direkt övertagande av spindeln S1, S2, ...

GET-krav utan fördekoderingsstopp

Frigivs efter ett GET-krav **utan** fördekoderingsstopp axeln med RELEASE (Achse) eller WAITP (Achse) åter så leder ett efterföljande GET till ett GET **med** fördekoderingsstopp.

**SE UPP****Axeltillordning ändrad**

En axel resp. spindel som övertagits med GET förblir också efter en tangent- eller program-RESET tillordnad till denna kanal.

Vid ny programstart måste tillordningen av de bytta axlarna resp. spindlarna göras programtekniskt, om axeln behövs i sin grundkanal.

Vid POWER ON tillordnas de till den i maskindatamet lagrade kanalen.

Exempel**Exempel 1: Byte mellan två kanaler**

Av 6 axlar används för bearbetningen i kanal 1: 1:a, 2:a, 3:e och 4:e axeln.
5:e och 6:e axeln används i kanal 2 till arbetsstycksväxling.

Axel 2 ska kunna bytas mellan båda kanalerna och vara tillordnad kanal 1 efter POWER ON.

Program "MAIN" i kanal 1:

Programkod	Kommentar
INIT (2,"TAUSCH2")	; Välja program TAUSCH2 i kanal 2.
N... START (2)	; Starta program i kanal 2.
N... GET (AX2)	; Överta axel AX2.
...	
N... RELEASE (AX2)	; Frige axel AX2.
N... WAITM (1,1,2)	; Vänta på WAIT-märke i kanal 1 och 2 för synkronisering i de båda kanalerna.
...	; Fortsatt förlopp efter axelbyte.
N... M30	

Program "TAUSCH2" i kanal 2:

Programmering	Kommentar
N... RELEASE (AX2)	
N160 WAITM(1,1,2)	; Vänta på WAIT-märke i kanal 1 och 2 för synkronisering i de båda kanalerna.
N150 GET(AX2)	; Överta axel AX2.
...	; Fortsatt förlopp efter axelbyte.
N... M30	

Exempel 2: Axelbyte utan synkronisering

Om axeln inte måste synkroniseras skapas inget fördekoderingsstopp av GET.

Programmering	Kommentar
N01 G0 X0	
N02 RELEASE (AX5)	
N03 G64 X10	
N04 X20	
N05 GET (AX5)	; När ingen synkronisering är nödvändig blir detta inget utförbart block.
N06 G01 F5000	; Inget utförbart block.
N07 X20	; Inget utförbart block, eftersom X-Position som i N04.
N08 X30	; Första utförbart block efter N05.
...	

Exempel 3: Aktivering av ett axelbyte utan fördekoderingsstopp

Förutsättning: Axelbyte utan fördekoderingsstopp måste projekteras via ett maskindatum.

Programmering	Kommentar
N010 M4 S100	
N011 G4 F2	
N020 M5	
N021 SPOS=0	
N022 POS[B]=1	
N023 WAITP(B)	; Axel B blir den neutrala axeln.
N030 X1 F10	
N031 X100 F500	
N032 X200	
N040 M3 S500	; Axeln utlöser inget fördekoderingsstopp/REORG.
N041 G4 F2	
N050 M5	
N099 M30	

Förflyttas spindeln resp. axel B omedelbart efter blocket N023 som **PLC-axel** t.ex. till 180 grader och tillbaka till 1 grad då blir denna axel åter den neutrala axeln och utlöser i block N40 inget fördekoderingsstopp.

Ytterligare information**Förutsättningar för axelbytet**

- Axlarna måste vara definierade via maskindata i alla kanaler som axlarna vill använda.
- Via det **axels**specifika maskindatumet måste vara fastlagt, till vilken kanal axeln ska tillordnas efter POWER ON.

Beskrivning

Frige axel: RELEASE

Beakta följande vid axelfrigivningen:

1. Axeln får inte delta i någon transformation.
2. Vid axelkopplingar (tangentialstyrning) måste alla axlar i förbandet frigges.
3. En konkurrerande positioneringsaxel kan inte bytas i detta tillstånd.
4. Vid en Gantry-masteraxel byts också alla följdaxlarna.
5. Vid axelkopplingar (medsläpning, ledvärdeskoppling, elektronisk växel) kan endast styraxeln i förbandet frigges.

Överta axel: GET

Med detta kommando genomförs det egentliga axelbytet. Ansvaret för axeln ligger fullständigt hos den kanal i vilken kommandot programmerades.

Inverkan från GET:

Axelbyte med synkronisering:

En axel måste alltid synkroniseras när den under mellantiden var tillordnad till en annan kanal eller PLC och före GET ingen synkronisering genom "WAITP", G74 eller restvägsradering har ägt rum.

- Ett fördekoderingsstopp följer (som vid STOPRE).
- Bearbetningen stoppas tills bytet har utförts fullständigt.

Automatiskt "GET"

När en axel principiellt är tillgänglig i kanalen, dock för tillfället inte finns som "Kanal-axel" utförs automatiskt ett GET. Om ax(eln)(larna) redan är synkroniserad(e) skapas inget fördekoderingsstopp.

Ställa in axelbytesbeteende föränderligt

Tidpunkten för överlämning av axlar låter sig ställas in via ett maskindatum på följande sätt:

- Automatiskt axelbyte äger rum mellan två kanaler också när axeln fördes till ett neutralt tillstånd genom WAITP (hittillsvarande beteende)
- Vid kravet av en axel-containerrotation hämtas alla axlar i axel-containern som kan tillordnas till den utförande kanalen med hjälp av implicita GET resp. GETD till kanalen. Ett följande axelbyte är tillåtet först efter att axel-containerrotationen avslutats.
- Efter ett inskjutet mellanblock i huvudkörningen kontrolleras om en reorganisation är erforderlig eller inte. Endast när axeltillstånden för detta block **inte** överensstämmer med de aktuella axeltillstånden är en reorganisation erforderlig.
- I stället för ett GET-block med fördekoderingsstopp och synkronisering mellan förkörning och huvudkörning kan ett axelbyte göras också utan fördekoderingsstopp. Då skapas endast ett mellanblock med GET-kravet. I huvudkörningen kontrolleras vid genomarbetningen av detta block om axelns tillstånd i blocket överensstämmer med de aktuella axeltillstånden.

Ytterligare informationer för funktionaliteten hos ett axel- eller spindelbyte se Funktionshandbok Tilläggsfunktioner; BAGs, Kanaler, Axelbyte (K5).

2.16 Överlämna axel till en annan kanal (AXTOCHAN)

Med språkkommandot `AXTOCHAN` kan en axel begäras för att överlämna denna axel till en annan kanal. Axeln kan föras till den motsvarande kanalen både utifrån NC-detaljprogrammet och även utifrån en synkronaktion.

Syntax

```
AXTOCHAN (Achsname, Kanalnummer [, Achsname, Kanalnummer [, ... ]])
```

Betydelse

Element	Beskrivning
AXTOCHAN:	Begära axel för en bestämd kanal
Achsname:	Axeltillordning i systemet: X, Y, ... eller uppgift över de delatagande maskin-axelnamnen. Den utförande kanalen måste inte vara den egna kanalen och det måste inte heller vara den kanal som aktuellt besitter interpoleringsrätten för axeln
Kanalnummer:	Nummer på den kanal som ska tillordnas till axeln

Märk

Konkurrerande positioneringsaxel och uteslutande PLC kontrollerade axlar

En PLC-axel kan som konkurrerande positioneringsaxel inte växla kanalen. En uteslutande av PLC kontrollerad axel kan inte tillordnas NC-programmet.

Litteratur

Funktionshandbok Tilläggsfunktioner; Positioneringsaxlar (P2)

Exempel

AXTOCHAN i NC-programmet

Axlarna X och Y är kända i 1:a och 2:a kanalen. Kanal känd. Aktuellt har kanal 1 interpoleringsrätten och i kanal 1 startas följande program:

Programkod	Kommentar
N110 AXTOCHAN(Y,2)	; Skjuta Y-axeln till den 2:a kanalen.
N111 M0	
N120 AXTOCHAN(Y,1)	; Åter hämta tillbaka Y-axeln (neutral).
N121 M0	
N130 AXTOCHAN(Y,2,X,2)	; Skjuta Y-axeln och X-axeln till den 2:a kanalen (axlar neutrala).
N131 M0	
N140 AXTOCHAN(Y,2)	; Skjuta Y-axeln till den 2:a kanal (NC-program).
N141 M0	

Ytterligare informationer

AXTOCHAN i NC-programmet

Därvid utförs endast vid en begäran av axeln för NC-programmet i den egna kanalen ett `GET` och därmed väntas också på den verkliga tillståndsförändringen. Begärs axeln för en annan kanal eller ska den bli den neutrala axeln i den egna kanalen, endast då stryks kravet.

AXTOCHAN från en synkronaktion

Begärs en axel för den egna kanalen så avbildas `AXTOCHAN` från en synkronaktion på ett `GET` från en synkronaktion. I detta fall blir axeln vid den första begäran för den egna kanalen till neutral axel. Vid den andra begäran tillordnas axeln NC-programmet analogt till `GET`-kravet i NC-programmet. För `GET`-kravet från en synkronaktion se Kapitel "Rörelsesynkronaktioner".

2.17 Ställa in maskindata verksamt (NEWCONF)

Med kommandot `NEWCONF` ställs alla maskindata in verksamt. Funktionen kan också aktiveras i användargränssnittet HMI genom att trycka ner funktionstangenten "Sätta MD verksamt".

Vid utförandet av funktionen "NEWCONF" följer ett implicit fördekoderingsstopp dvs. banrörelsen stoppas.

Syntax

`NEWCONF`

Betydelse

NEWCONF:	Kommando för verksamsättning av alla maskindata till verksamhetsnivån "NEW_CONFIG"
----------	--

Föra bort NEWCONF kanalövergripande från detaljprogrammet

Förändras axiala maskindata utifrån detaljprogrammet och aktiveras sedan med `NEWCONF` så sätter kommandot `NEWCONF` endast de maskindata aktiva som förorsakar ändringar för kanalen till detaljprogrammet.

Märk

För att säkert låta alla ändringar bli verksamma måste kommandot `NEWCONF` utföras i varje kanal i vilken också de av de förändrade maskindata träffade axlearna eller funktionerna aktuellt beräknas.

Vid `NEWCONF` sätts inga axiala maskindata verksamt.

För PLC-kontrollerade axlar måste en axial RESET utföras.

Exempel

Fräsbearbetning: Bearbeta borrarposition med olika teknologier

Programkod	Kommentar
N10 \$MA_CONTOUR_TOL[AX]=1.0	; Ändra maskindatum.
N20 NEWCONF	; Sätta maskindata verksamt.
...	

2.18 Skriva fil (WRITE)

Med `WRITE`-kommandot kan block/data från NC-programmet skrivas i slutet av en fil (protokollfil) som befinner sig i det passiva Filesystemet eller i ett externt programminne. Detta kan också vara det program som just befinner sig i bearbetning.

Märk

En fil som ska skrivas med `WRITE`-kommando nytiläggas, om den inte existerar i programminnet.

Förutsättning

Den aktuellt inställda skyddsnivån måste vara lika med eller större än `WRITE`-rätten för filen. Är detta inte fallet vägras åtkomsten med felmeddelande (returvärde för felvariablerna = 13).

Syntax

```
DEF INT <Fehler>
...
WRITE(<Fehler>,"<Dateiname>"/"<ExtG>","<Satz/Daten>")
```

Betydelse

WRITE:	Kommando för tillägg av ett block resp. data i slutet av den angivna filen		
<Fehler>:	Parameter 1: Variabel för återlämnande av felvärdet		
	Typ:	INT	
	Värde:	0	Inget fel
		1	Sökväg inte tillåten
		2	Sökväg inte hittad
		3	Fil inte hittad
		4	Felaktig filtyp
		10	Fil är full
		11	Fil används
		12	Inga resurser fria
		13	Inga åtkomsträtter
		14	EXTOPEN som saknas eller misslyckats för utmatningsinstrumentet
		15	Fel vid skrivande på externt instrument
		16	Ogiltig extern sökväg programmerad
<Dateiname>:		Parameter 2: Namn på den fil i vilken det angivna blocket resp. de angivna data ska tillfogas	
	Typ:	STRING	
	Före det egentliga filnamnet kan den absoluta sökvägen angivas. Utan sökvägsuppgift söks filen i den aktuella katalogen (= katalog för det valda programmet). Regler för sökvägsuppgift se "Adressering av filer i programminnet (Sida 213)".		

2.18 Skriva fil (WRITE)

<ExtG>:	Ska data matas ut via funktionen "Process DataShare" på en extern apparat/fil, måste i stället för filnamnet den symboliska beteckningen för den externa apparat/fil som ska öppnas anges.	
	Typ:	STRING
	Ytterligare informationer se "Process DataShare - utgåva på en extern apparat/fil (EXTOPEN, WRITE, EXTCLOSE) (Sida 652)". Observera: Beteckningen måste vara identiskt till den i EXTOPEN-kommandot angivna beteckningen.	
<Satz/Daten>:	Parameter 3: Block resp. data som ska läggas till i den angivna filen.	
	Typ:	STRING

Märk

Vid skrivande i det passiva filsystemet eller på ett externt programminne infogar WRITE-kommandot implicit ett "LF"-tecken (LINE FEED = ny rad) i slutet av utmatningsstringen.

För utmatningen på en extern apparat/fil via funktionen "Process DataShare" gäller inte detta beteende. Ska en "LF" matas med ut, måste det explicit anges i utmatningsstringen.

→ Se härtill exempel 3: Implicit/explicit "LF"!

Randvillkor

- **Maximal filstorlek (→ maskintillverkare!)**
Den maximalt möjliga filstorleken för protokollfiler i det passiva filsystemet ställs in med maskindatumet:
MD11420 \$MN_LEN_PROTOCOL_FILE
Den maximala filstorleken gäller för alla filer som läggs till med WRITE-kommandot i det passiva filsystemet. Vid överskridande matas ett felmeddelande ut och blocket resp. data sparas inte. Om minnet räcker till kan en ny fil läggas till.

Exempel**Exempel 1: WRITE-kommando i det passiva filsystemet utan absolut sökväg**

Programkod	Kommentar
N10 DEF INT ERROR	; Definition av felvariablerna.
N20 WRITE(ERROR, "PROT", "PROTOKOLL VOM 7.2.97")	; Skriver texten "PROTOKOLL VOM 7.2.97" i filen _N_PROT_MPF.
N30 IF ERROR	; Felutvärdering
N40 MSG ("Fehler bei WRITE-Befehl:" << ERROR)	
N50 M0	
N60 ENDIF	
...	

Exempel 2: WRITE-kommando i det passiva filesystemet med absolut sökväg**Programkod**

```
...
WRITE(ERROR, "/_N_WKS_DIR/_N_PROT_WPD/_N_PROT_MPF", "PROTOKOLL VOM
7.2.97")
...
```

Exempel 3: Implicit/explicit "LF"

a, Skriva i det passiva filesystemet med implicit skapad "LF"

Programkod

```
...
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
```

Utmatningsresultat:

MY_STRING

MY_STRING

b, Skriva i extern fil utan implicit skapad "LF"

Programkod

```
...
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
```

Utmatningsresultat:

MY_STRINGMY_STRING

c, Skriva i extern fil med explicit programmderad "LF"

För att uppnå samma resultat som vid a, måste följande programmeras:

Programkod

```
...
N200 DEF STRING[30] DEV_1
```

2.18 Skriva fil (WRITE)

Programkod

```
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
```

Utmatningsresultat:

MY_STRING

MY_STRING

2.19 Radera fil (DELETE)

Med `DELETE`-kommandot kan alla filer raderas oberoende av om dessa uppstått per `WRITE`-kommando eller inte. Även filer som upprättats i högsta åtkomstnivån kan raderas med `DELETE`.

Syntax

```
DEF INT <Fehler>
DELETE(<Fehler>,"<Dateiname>")
```

Betydelse

DELETE:	Kommando för att radera den angivna filen		
<Fehler>:	Variabel för återlämnande av felvärdet		
	Typ:	INT	
	Värde:	0	Inget fel
		1	Sökväg inte tillåten
		2	Sökväg inte hittad
		3	Fil inte hittad
		4	Felaktig filtyp
		11	Fil används
		12	Inga resurser fria
20		Övriga fel	
<Dateiname>:	Namn på filen som ska raderas		
	Typ:	STRING	
	Före det egentliga filnamnet kan den absoluta sökvägen angivas. Utan sökväg-suppgift söks filen i den aktuella katalogen (= katalog för det valda programmet). Regler för sökvägsuppgift se "Adressering av filer i programminnet (Sida 213)".		

Exempel

Programkod	Kommentar
N10 DEF INT ERROR	; Definition av felvariablerna.
N15 STOPRE	; Fördekoderingsstopp.
N20 DELETE(ERROR,"/_N_SPF_DIR/_N_TEST1_SPF")	; Radera filen TEST1 i underprogramkatalogen.
N30 IF ERROR	; Felutvärdering
N40 MSG("Fehler bei DELETE-Befehl:" <<ERROR)	
N50 M0	
N60 ENDIF	

2.20 Läs rader i fil (READ)

READ-kommandot läser i den angivna filen en eller flera rader och lägger de lästa informationerna i ett fält av typ STRING. Varje läst rad belägger i detta fält ett fältelement.

Förutsättning

Den aktuellt inställda skyddsnivån måste vara lika med eller större än READ-rätten för filen. Är detta inte fallet vägras åtkomsten med felmeddelande (returvärde för felvariablerna = 13).

Syntax

```
DEF INT <Fehler>
DEF STRING[<Stringlänge>] <Ergebnis>[<n>,<m>]
READ(<Fehler>,"<Dateiname>",<Anfangszeile>,<Zeilenanzahl>,<Ergebnis>
)
```

Betydelse

READ:	Kommando för att läsa rader i den angivna filen och för att lägga dessa rader i ett variabelfält.			
<Fehler>:	Variabel för återlämnande av felvärdet (Call-By-Reference-Parameter)			
	Typ:	INT		
	Värde:	0	Inget fel	
		1	Sökväg inte tillåten	
		2	Sökväg inte hittad	
		3	Fil inte hittad	
		4	Felaktig filtyp	
		11	Fil används	
		13	Åtkomsträtter inte tillräckliga	
		21	Rad finns inte (Parameter <Anfangszeile> eller <Zeilenanzahl> större än antalet rader i den angivna filen).	
		22	Fältlängden för resultatvariabeln (<Ergebnis>) är för liten.	
	23	Radområdet för stort (Parameter <Zeilenanzahl> så stort vald att det läses utöver filslutet).		
<Dateiname>:	Namn på den fil som ska läsas (Call-By-Value-Parameter)			
	Typ:	STRING		
	Före det egentliga filnamnet kan den absoluta sökvägen angivas. Utan sökvägsuppgift söks filen i den aktuella katalogen (= katalog för det valda programmet). Regler för sökvägsuppgift se "Adressering av filer i programminnet (Sida 213)".			

<Anfangszeile>:	Begynnelse rad i det filområdet som ska läsas (Call-By-Value-Parameter)		
	Typ:	INT	
	Värde:	0	Det läses de med parametern <Zeilenanzahl> angivna antalet rader före filslutet.
1 ... n		Nummer på den första raden som ska läsas.	
<Zeilenanzahl>:	Antal rader som ska läsas (Call-By-Value-Parameter)		
	Typ:	INT	
<Ergebnis>:	Resultatvariabel (Call-By-Reference-Parameter)		
	Variabelfält i vilket den lästa texten ska läggas.		
	Typ:	STRING (max. längd: 255)	
	Om i parametern <Zeilenanzahl> mindre rader angivits är fältstorleken [$\langle n \rangle$, $\langle m \rangle$] för resultatvariablerna uppgår till, då förändras inte de resterande fältelementen.		
Avslutningen av en rad med styrtecknen "LF" (Line Feed) eller "CR LF" (Carriage Return Line Feed) lagras inte i resultatvariablerna.			
Lästa rader skärs av när raden är längre än den definierade stringlängden. Det följer inget felmeddelande.			

Märk

Binära filer kan inte läsas in. Felet "felaktig filtyp" (returvärde för felvariablerna = 4) matas ut. Följande filtyper kan inte läsas: _BIN, _EXE, _OBJ, _LIB, _BOT, _TRC, _ACC, _CYC, _NCK.

Exempel

Programkod	Kommentar
N10 DEF INT ERROR	; Definition av felvariablerna.
N20 DEF STRING[255] RESULT[5]	; Definition av resultatvariablerna.
N30 READ(ERROR,"/_N_CST_DIR/_N_TESTFILE_MPF", 1,5,RESULT)	; Filnamn med domain-, filidentifikation och sökuppgift.
N40 IF ERROR <>0	; Felutvärdering
N50 MSG("FEHLER"<<ERROR<<"BEI READ-BEFEHL")	
N60 M0	
N70 ENDIF	
...	

2.21 Kontrollera om en fil finns (ISFILE)

Med ISFILE-kommandot kan kontrolleras om en fil existerar i programminnet.

Syntax

```
<Resultat>=ISFILE("<Filnamn>")
```

Betydelse

ISFILE:	Kommando för att kontrollera om en fil finns		
<Filnamn>:	Namn på den fil vars existens ska kotrolleras.		
	Typ:	STRING	
	Före det egentliga filnamnet kan den absoluta sökvägen angivas. Utan sökvägssuppgift söks filen i den aktuella katalogen (= katalog för det valda programmet). Regler för sökvägssuppgift se "Adressering av filer i programminnet (Sida 213)".		
<Resultat>:	Resultatvariabel för registrering av kontrollresultatet		
	Typ.	BOOL	
	Värde:	TRUE	Fil finns
		FALSE	Fil finns inte

Exempel

Exempel 1

Programkod	Kommentar
N10 DEF BOOL RESULT	; Definition av resultatvariablerna.
N20 RESULT=ISFILE("TESTFILE")	
N30 IF (RESULT==FALSE)	
N40 MSG("FIL FINNS INTE")	
N50 M0	
N60 ENDIF	
...	

Exempel 2

Programkod	Kommentar
N10 DEF BOOL RESULT	; Definition av resultatvariablerna.
N20 RESULT=ISFILE("TESTFILE")	
N30 IF (NOT ISFILE("TESTFILE"))	
N40 MSG("FIL FINNS INTE")	
N50 M0	
N60 ENDIF	
...	

2.22 Läs ut fil-informationer (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO)

Via kommandona FILEDATE, FILETIME, FILESIZE, FILESTAT och FILEINFO kan vissa fil-informationer som datum/tid för den sista skrivande åtkomsten, aktuell filstorlek, fil-status eller summan av dessa informationer läsas ut.

Förutsättning

Den aktuellt inställda skyddsnivån måste vara lika med eller större än Show-rätten för den överordnade katalogen.. Är detta inte fallet vägras åtkomsten med felmeddelande (returvärde för felvariablerna = 13).

Syntax

```
FILE....(<Fel>,"<Filnamn>",<Resultat>)
```

Betydelse

FILEDATE:	Leverar datum för den sista åtkomsten till en fil för skrivning		
FILETIME:	Leverar tidpunkt för den sista åtkomsten till en fil för skrivning		
FILESIZE:	Levererar den aktuella storleken för en fil		
FILESTAT:	Levererar för en fils status beträffande följande rättigheter : • Läs (r: read) • Skriva (w: write) • Utföra (x: execute) • Visa (s: show) • Radera (d: delete) Observera: Dessa skyddsnivåer är speciella egenskaper i det passiva Filesystemet. Vid åtkomst till det externa programminnet levererar FILESTAT därför endast Default-åtkomsträtter (77777).		
FILEINFO:	Levererar för en fil summan av informationer , som kan läsas ut via FILEDATE, FILETIME, FILESIZE och FILESTAT		
<Fel>:	Variabel för återlämnande av felvärdet (Call-By-Reference-Parameter)		
	Typ.	VAR INT	
	Värde:	0	Inget fel
		1	Sökväg inte tillåten
		2	Sökväg inte hittad
		3	Fil inte hittad
		4	Felaktig filtyp
		13	Åtkomsträtter inte tillräckliga
		22	Stringlängden för resultatvariabeln (<Ergebnis>) är för liten.

2.22 Läs ut fil-informationer (FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO)

<Filnamn>:	Namn på den fil från vilken fil-information(er) ska läsas ut			
	Typ:	CHAR[160]		
	Före det egentliga filnamnet kan den absoluta sökvägen angivas. Utan sökvägsuppgift söks filen i den aktuella katalogen (= katalog för det valda programmet).			
	Regler för sökvägsuppgift se "Adressering av filer i programminnet (Sida 213)".			
<Resultat>:	Resultatvariabel (Call-By-Reference-parameter)			
	Variabel i vilken den begärda fil-informationen är lagrad.			
	Typ:	VAR CHAR[8]	vid	FILEDATE Format: "dd.mm.yy"
		VAR CHAR[8]	vid	FILETIME Format: "hh:mm:ss"
		VAR INT	vid	FILESIZE Filstorleken matas ut i byte.
		VAR CHAR[5]	vid	FILESTAT Format: "rwxsd" (r: read, w: write, x: execute, s: show, d: delete)
VAR CHAR[32]		vid	FILEINFO Format: "rwxsd nnnnnnnn dd.mm.yy hh:mm:ss"	

Exempel

Programkod	Kommentar
N10 DEF INT ERROR	; Definition av felvariabler-na.
N20 STRING[32] RESULT	; Definition av resultatvariablerna.
N30 FILEINFO(ERROR, "/_N_MPF_DIR/_N_TESTFILE_MPF", RESULT)	; Filnamn med domain-, filidentifikation och sökvägsuppgift.
N40 IF ERROR <> 0	; Felutvärdering
N50 MSG("FEL"<<ERROR<<"VID FILEINFO-KOMMANDO")	
N60 M0	
N70 ENDIF	
...	

Exemplet skulle i resultatvariabeln RESULT kunna leverera t.ex. följande resultat:

"77777 12345678 26.05.00 13:51:30"

2.23 Avrundning uppåt (ROUNDUP)

Med funktionen "ROUNDUP" kan inmatningsvärden av typ REAL (brutna tal med decimalpunkt) avrundas uppåt till nästa större hela tal.

Syntax

ROUNDUP (<Wert>)

Betydelse

ROUNDUP:	Kommando för avrundning uppåt av ett inmatningsvärde
<Wert>:	Inmatningsvärde av typ REAL

Märk

Inmatningsvärden av typ INTEGER (ett heltal) skickas tillbaka oförändrade.

Exempel

Exempel 1: Olika inmatningsvärden och deras avrundningsresultat

Exempel	Avrundningsresultat
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

Exempel 2: ROUNDUP i NC-programmet

Programkod
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])
...

2.24 Underprogramteknik

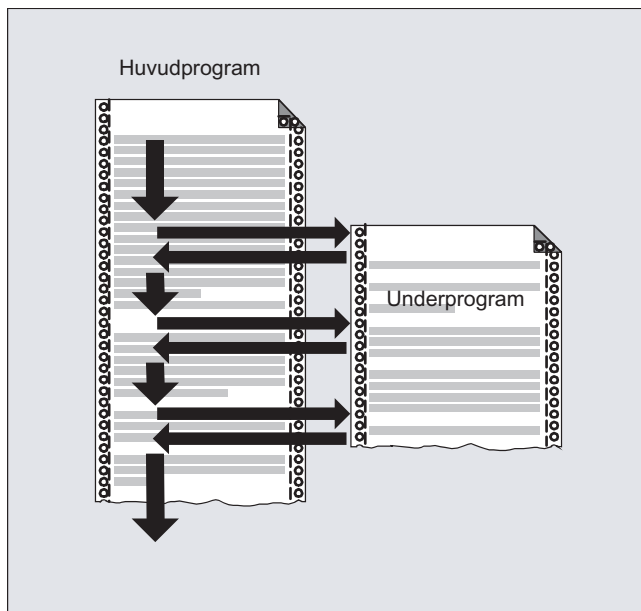
2.24.1 Allmänt

2.24.1.1 Underprogram

Beteckningen "Underprogram" stammar ännu från den tiden när detaljprogram var fast indelade i huvud- och underprogram. Huvudprogram var därvid detaljprogrammen som valdes vid styrning för genomarbetning och sedan startades. Underprogram var detaljprogrammen som anropades från huvudprogrammet.

Denna fasta indelning finns inte längre med det nutida SINUMERIK NC-språket. Varje detaljprogram kan principiellt väljas och startas som huvudprogram eller anropas som underprogram från ett annat detaljprogram.

Därmed menas i fortsättningen med underprogram ett detaljprogram som kan anropas från ett annat detaljprogram.



Användning

Som i alla högre programspråk används också i NC-språket underprogram till att lagra programdelar, som används flera gånger, som självständiga i sig färdiga program.

Underprogrammen erbjuder följande fördelar:

- Förhöjer översiktlighet och läsbarhet hos programmen
- Förhöjer kvaliteten genom återanvändning av testade programdelar
- Erbjuder möjligheten till skapande av specifika bearbetningsbibliotek
- Sparande av minnesplats

2.24.1.2 Underprogramnamn

Benämningsregler

Underprogramnamnet kan väljas fritt under respekterande av följande regler:

- Tillåtna tecken:
 - Bokstäver: A ... Z, a ... z
 - Siffror: 0 ... 9
 - Nedsänkta streck: _
- De första båda tecknen bör vara två bokstäver eller ett nedsänkt streck följt av en bokstav.

Märk

När detta villkor är uppfyllt då kan ett NC-program anropas endast genom angivande av programnamnet som underprogram utifrån ett annat program. Börjar programnamnet däremot med siffror då är underprogramanrop endast möjligt via CALL-anvisningen.

- Maximal längd: 24 tecken

Märk

Stora / små bokstäver

I SINUMERIK NC-språket skiljer man **inte** mellan skrivning av stora och små bokstäver.

Märk

Otillåtna programnamn

För att undvika problem med Windows-applikationer, får följande programnamn **inte** användas:

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

Styrningsinterna utvidgningar

Det vid upprättandet av programmet utdelade underprogramnamnet kompletteras styrningsinternt med ett prefix och ett postfix:

- Prefix: `_N_`
- Postfix: `_SPF`

Användning av programnamnet

Vid användningen av programnamnet t.ex. vid ett underprogramanrop är alla kombinationer av prefix, programnamn och postfix möjliga.

Exempel:

Underprogrammet med programnamnet SUB_PROG kan anropas via följande beteckning:

1. SUB_PROG
2. _N_SUB_PROG
3. SUB_PROG_SPF
4. _N_SUB_PROG_SPF

Namnlikhet för huvud- och underprogram

Existerar ett huvudprogram (.MPF) och ett underprogram (.SPF) med samma programnamn måste vid användningen av programnamnet i NC-programmet också respektive filidentifikationer anges för ett entydig kännetecknande. Annars används det program, som först hittas med det angivna programnamnet längs sökvägen.

2.24.1.3 Förgrening av underprogram

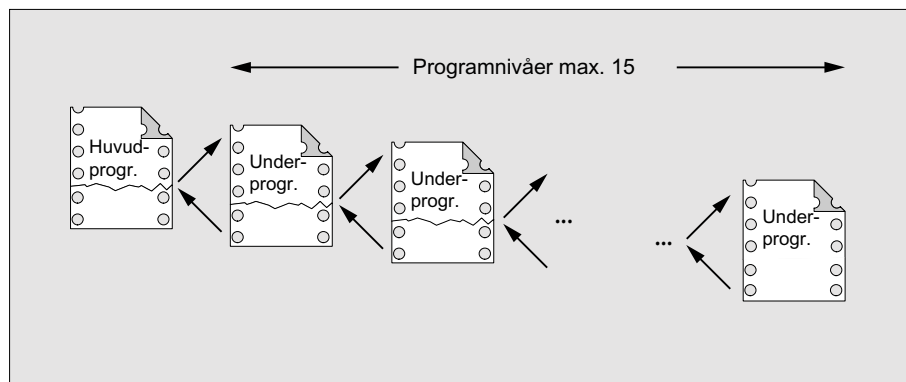
Ett huvudprogram kan anropa underprogram som återigen anropar underprogram. Programmens förlopp är därmed kapslade i varandra. Varje program går därvid på en egen programnivå.

Kapslingsdjup

NC-språket ställer aktuellt 16 programplan till förfogande. Huvudprogrammet går alltid i den översta programnivån 0. Ett underprogram går alltid i den nästa lägre programnivån som följer på anropet. Programnivå 1 är därmed den första underprogramnivån.

Uppdelning i programnivåer:

- Programnivå 0: Huvudprogramnivå
- Programnivå 1 - 15: Underprogramnivå 1 - 15



Interruptrutiner (ASUP)

Anropas inom ramen av en interruptrutin ett underprogram, genomarbetas detta inte i den aktuella i kanalen aktiva programnivån (n), utan likaså i närmast lägre programnivån (n+1). För att detta ska vara möjligt även i den lägsta programnivån, står i sammanhang med interruptrutiner 2 extra programnivåer (16 och 17) till förfogande.

Behövs med än 2 programnivåer måste det explicit tas hänsyn till detta i struktureringen av det i kanalen genomarbetade detaljprogrammet. Dvs. då får endast maximalt så många programnivåer tas i anspråk att fortfarande tillräckligt med programnivåer står till förfogande för interruptbearbetningen.

Behöver interruptbearbetningen t.ex. 4 programnivåer måste detaljprogrammet struktureras så att det belägger maximalt 13 programnivåer. Följer sedan en interrupt står denna de nödvändiga 4 programnivåerna (14 till 17) till förfogande.

Siemens-cykler

Siemens-cykler behöver 3 programnivåer. Anropet av en Siemens-cykel måste därför senast göras i:

- Detaljprogrambearbetning: Programnivå 12
- Interruptrutin: Programnivå 14

2.24.1.4 Sökväg

Vid anrop av ett underprogram utan sökvägsuppgift genomsöker styrningen enligt en fördefinierad sökföljd (se "Sökväg vid underprogramanrop (Sida 217)") de programminnen som finns.

2.24.1.5 Formal- och aktualparametrar

Man talar om formal- och aktualparametrar i samband med definition och anrop av underprogram med parameteröverlämning.

Formalparametrar

Vid definitionen av ett underprogram måste de parametrar som ska överlämnas till underprogrammet, de så kallade formalparametrarna, definieras med typ och parameternamn.

Formalparametrarna definierar därmed underprogrammets gränssnitt.

Exempel:

Programkod	Kommentar
PROC KONTUR (REAL X, REAL Y)	; Formalparametrar: X och Y båda av typ REAL
N20 X1=X Y1=Y	; Förflyttning av axel X1 till position X och axel Y1 till position Y
...	
N100 RET	

Aktualparametrar

Vid anrop av ett underprogram måste absoluta värden eller variabler, de så kallade aktualparametrarna, överlämnas till underprogrammet.

Aktualparametrarna fyller på så sätt underprogrammets gränssnitt med aktuella värden.

Exempel:

Programkod	Kommentar
N10 DEF REAL BREITE	; Variabeldefinition
N20 BREITE=20.0	; Variabeltillordning
N30 KONTUR(5.5, BREITE)	; Underprogramanrop med aktualparametrar: 5.5 och BREITE
...	
N100 M30	

2.24.1.6 Parameteröverföring

Definition av ett underprogram med parameteröverföring

Definitionen av ett underprogram med parameteröverföring görs med nyckelordet `PROC` och en fullständig lista med alla parametrar som underprogrammet väntar på.

Ofullständig parameteröverföring

Vid anrop av underprogrammet måste inte alltid alla i underprogrammets gränssnitt definierade parametrar explicit överlämnas. Utelämnas en parameter ges standardvärdet "0" åt denna parameter.

För entydig markering av parametrarnas ordningsföljd måste dock komman alltid anges som skiljetecken mellan parametrarna. Ett undantag utgör den sista parameter. Utelämnas denna vid anrop kan också det sista kommat utgå.

Exempel:

Underprogram:

Programkod	Kommentar
PROC SUB_PROG (REAL X, REAL Y, REAL Z)	; Formalparametrar: X, Y och Z
...	
N100 RET	

Huvudprogram:

Programkod	Kommentar
PROC MAIN_PROG	
...	
N30 SUB_PROG(1.0,2.0,3.0)	; Underprogramanrop med fullständig parameteröverföring: X=1.0, Y=2.0, Z=3.0
...	
N100 M30	

Exempel för underprogramanrop i N30 med ofullständig parameteröverföring:

```
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
```

OBSERVERA

Parameteröverföring Call-by-Reference

Parametrar som överfärs via Call-by-Reference får inte utelämnas vid underprogramanrop.

OBSERVERA

Datatyp AXIS

Parametrar av datatyp AXIS får inte utelämnas vid underprogramanrop.

Kontroll av överföringsparametrar

Via systemvariabeln \$P_SUBPAR [n] mit n = 1, 2, ... kan kontrolleras i underprogrammet om en parameter överfördes explicit eller utelämnades. Index n hänför sig till ordningsföljden för formalparametrarna. Index n°=°1 hänför sig till den 1:a formalparametern, index n°=°2 till den 2:a formalparametern osv.

Det följande programavsnittet visar med exempel för den 1:a formalparametern hur en kontroll kan realiseras:

Programmering	Kommentar
PROC SUB_PROG (REAL X, REAL Y, REAL Z)	; Formalparametrar: X, Y och Z
N20 IF \$P_SUBPAR[1]==TRUE	; Kontroll av den 1:a formalparametern X.
...	; Dessa aktioner utförs när formalparametern X explicit överfördes.
N40 ELSE	
...	; Dessa aktioner utförs när formalparametern X inte överfördes.
N60 ENDIF	
...	; Allmänna aktioner
N100 RET	

2.24.2 Definition av ett underprogram

2.24.2.1 Underprogram utan parameteröverföring

Vid definitionen av underprogram utan parameteröverföring kan definitionsraden vid programmets början utgå.

Syntax

```
[PROC <Programmname>]
...
```

Betydelse

PROC:	Definitionsanvisning i början av ett program
<Programmname>:	Namn på programmet

Exempel

Exempel 1: Underprogram med PROC-anvisning

Programkod	Kommentar
PROC SUB_PROG	; Definitionsrad
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; Underprogramåterhopp

Exempel 2: Underprogram utan PROC-anvisning

Programkod	Kommentar
N10 G01 G90 G64 F1000	
N20 X10 Y20	
...	
N100 RET	; Underprogramåterhopp

Se även

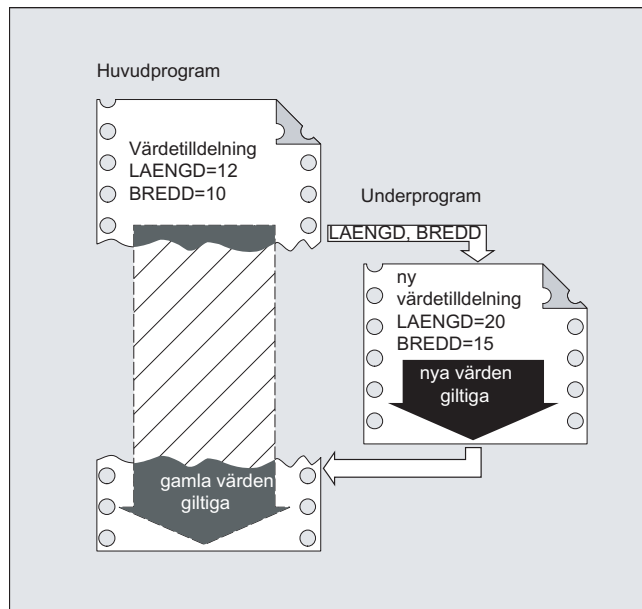
Underprogramanrop utan parameteröverföring (Sida 186)

2.24.2.2 Underprogram med parameteröverföring Call-by-Value (PROC)

Definitionen av ett underprogram med parameteröverföring Call-by-Value görs med nyckelordet `PROC` följt av programnamnet och en fullständig lista med alla parametrar med typ och namn. Definitionsanvisningen måste stå i den första programraden.

Call-by-Value

Det anropande programmet överlämnar vid en parameteröverföring Call-by-Value endast värdet för en av variablerna till underprogrammet. Därmed får underprogrammet ingen direkt åtkomst till variabeln. Därigenom blir vid en ändring av parametervärdet endast det i underprogrammet synliga värdet ändrat. Värdet för den i det anropande programmet definierade variabeln förblir oförändrat. Parameteröverföringen Call-by-Value har därmed inga återverkningar på programmet som anropar.



Syntax

```
PROC <Programname> (<Parametertyp> <Parametername>=<Init_Wert>, ...)
```

Märk

Upp till 127 parametrar kan överföras.

Betydelse

PROC:	Definitionsanvisning i början av ett program
<Programname>:	Namn på programmet
<Parametertyp>:	Datatyp för parametern (t.ex. REAL, INT, BOOL)
<Parametername>:	Namn på parametern
<Init_Wert>:	Optionalt värde för initiering av parametern (optional) Anges ingen parameter vid anrop av underprogrammet, erhåller parametern initieringsvärdet tilldelat.

Exempel

Exempel 1

Definition av ett underprogram SUB_PROG med tre parametrar av typ REAL med defaultvärden:

Programkod

```
PROC SUB_PROG (REAL LENGTH=10.0, REAL WIDTH=20.0, REAL HIGHT=30.0)
```

Exempel 2

Olika anropsvarianter

Programkod

```
PROC MAIN_PROG
  REAL PAR_1 = 100
  REAL PAR_2 = 200
  REAL PAR_3 = 300
  ; Anropsvarianter
  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
```

Se även

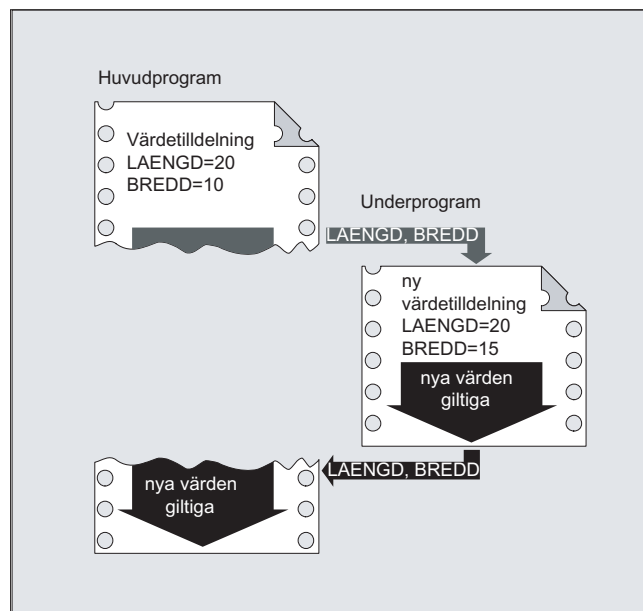
Underprogramanrop med parameteröverföring (EXTERN) (Sida 188)

2.24.2.3 Underprogram med parameteröverföring Call-by-Reference (PROC, VAR)

Definitionen av ett underprogram med parameteröverföring Call-by-Reference görs med nyckelordet `PROC` följt av programnamnet och en fullständig lista med alla parametrar med nyckelordet `VAR`, typ och namn. Definitionsanvisningen måste stå i den första programraden. Som parametrar kan referenser också överföras till fält.

Call-by-Reference

Det anropande programmet överlämnar vid en parameteröverföring Call-by-Reference inte värdet för en av variablerna till underprogrammet utan en referens (markör) till variabeln. Därmed får underprogrammet direkt åtkomst till variabeln. Därigenom blir vid en ändring av parametervärdet inte endast det i underprogrammet synliga värdet ändrat utan värdet för den i det anropande programmet definierade variabeln. Parameteröverföringen Call-by-Reference har därmed också efter avslutningen av underprogrammet återverkningar på programmet som anropar.



Märk

Parameteröverföringen Call-by-Reference är endast erforderlig när den överförda variabeln är definierad lokalt i ett program som anropar (LUD). Kanal-globala eller NC-globala variabler måste inte överföras eftersom åtkomst till dessa också är möjlig direkt från underprogrammet.

Syntax

```
PROC <Programname> (VAR <Parametertyp> <Parametername>, ...)  
PROC <Programname> (VAR <Feldtyp> <Feldname> [<m>,<n>,<o>], ...)
```

Märk

Upp till 127 parametrar kan överföras.

Betydelse

PROC:	Definitionsanvisning i början av ett program
VAR:	Nyckelord för parameteröverföring per referens
<Programname>:	Namn på programmet
<Parametertyp>:	Datotyp för parametern (t.ex. REAL, INT, BOOL)
<Parametername>:	Namn på parametern
<Feldtyp>:	Datotyp för fältelementet (t.ex. REAL, INT, BOOL)
<Feldname>:	Namn på fältet

[<m>, <n>, <o>]:	Fältstorlek	
	Aktuellt är maximalt 3-dimensionella fält möjliga:	
	<m>:	Fältstorlek för 1:a dimensionen
	<n>:	Fältstorlek för 2:a dimensionen
	<o>:	Fältstorlek för 3:e dimensionen

Märk

- Det efter nyckelordet `PROC` angivna programnamnet måste överensstämma med det på användargränssnittet utdelade programnamnet.
- Med fält av obestämd fältlängd som formalparametrar kan underprogram bearbeta fält av variabel längd. Därtill anges vid definitionen av t.ex. ett tvådimensionellt fält som formalparameter inte längden för den 1:a dimensionen. Men kommat måste skrivas ut.
Exempel: `PROC <Programname> (VAR REAL FELD[ ,5])`

Exempel

Definition av ett underprogram med två parametrar som referens av typ `REAL`:

Programkod

```
; Parameter 1: Referens till typ: REAL, Namn: LAENGE
; Parameter 2: Referens till typ: REAL, Namn: BREITE
PROC SUB_PROG(VAR REAL LAENGE, VAR REAL BREITE)
```

Se även

Underprogramanrop med parameteröverföring (`EXTERN`) (Sida 188)

2.24.2.4 Säkra modala G-funktioner (SAVE)

Attributet `SAVE` har till följd att de före underprogramanrop aktiva modala G-kommandona säkras och efter underprogramslut åter reaktiveras.

OBSERVERA**Avbrott i banstyrningsdriften**

Anropas vid aktiv banstyrningsdrift ett underprogram med attributet `SAVE` stoppas banstyrningsdriften i slutet av underprogrammet (återhopp).

Syntax

```
PROC <Underprogramname> SAVE
```


Betydelse

SAVE:	Säkring av de modala G-kommandona före underprogramanrop och återupprättande efter underprogramslut
-------	---

Exempel

I underprogrammet KONTUR verkar det modala G-kommandot G91 (kedjemått). I huvudprogrammet verkar det modala G-kommandot G90 (absolut mått). Genom underprogramdefinitionen med SAVE verkar efter underprogramslut åter G90 i huvudprogrammet.

Underprogram-definition:

Programkod	Kommentar
PROC KONTUR (REAL WERT1) SAVE	; Underprogram-definition med parameter SAVE
N10 G91 ...	; Modalt G-kommando G91: Kedjemått
N100 M17	; Underprogramslut

Huvudprogram:

Programkod	Kommentar
N10 G0 X... Y... G90	; Modalt G-kommando G90: Absolut måttsättning
N20 ...	
...	
N50 KONTUR (12.4)	; Underprogramanrop
N60 X... Y...	; Modalt G-kommando G90 reaktiverat av SAVE

Randvillkor

Frames

Beteendet hos frames beträffande underprogrammen med attributet SAVE är beroende av typen av frames och kan ställas in via maskindata.

Litteratur

Funktionshandbok Grundfunktioner; Axlar, koordinatsystem, frames (K2), Kapitel: "Underprogramåterhopp med SAVE"

2.24.2.5 Undertrycka blockvis bearbetning (SBLOF, SBLON)

Undertryckning av blockvis för hela programmet

Med SBLOF kännetecknade program genomarbetas komplett som ett block vid aktiv undertryckning av blockvis, dvs. för hela programmet undertrycks den blockvisa bearbetningen.

SBLOF står i PROC-raden och gäller till slutet för eller avbrott av underprogrammet. Med återhopp-kommandot fastläggs om det stoppas i slutet av underprogrammet eller inte.

Återhopp med M17: Stopp i slutet av underprogrammet

Återhopp med RET: Inget stopp i slutet av underprogrammet

Undertryckning av blockvis inom programmet

SBLOF måste stå ensam i blocket. Från och med detta block kopplas blockvis bort till:

- nästa SBLON eller
- slutet av den aktiva underprogramnivån

Syntax

Undertryckning av blockvis för hela programmet:

```
PROC ... SBLOF
```

Undertryckning av blockvis inom programmet:

```
SBLOF
```

```
...
```

```
SBLON
```

Betydelse

PROC:	Första anvisningen i ett program
SBLOF:	Kommando för fränkoppling av blockvis bearbetning SBLOF kan stå i ett PROC-block eller ensam i blocket.
SBLON:	Kommando för tillkoppling av blockvis bearbetning SBLON måste stå i ett eget block.

Randvillkor

- **Undertryckning av blockvis och blockindikering**
Den aktuella blockindikeringen kan undertryckas med `DISPLOF` i cykler/underprogram. Programmeras `DISPLOF` tillsammans med `SBLOF` så visas vid enkelblock-stopp inom cykeln/underprogrammet anropet av cykeln/underprogrammet som förut.
- **Undertryckning av blockvis i system-ASUP eller användar-ASUP**
När enkelblock-stoppet i system- eller användar-ASUP undertrycks via inställningarna i maskindatum MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK (Bit0 = 1 resp. Bit1 = 1), då kan enkelblock-stoppet åter aktiveras genom programmering av `SBLON` i ASUP. Undertrycks enkelblock-stoppet i användar-ASUP via inställningen i maskindatum MD20117 \$MC_IGNORE_SINGLEBLOCK_ASUP då kan enkelblock-stoppet **inte** åter aktiveras genom programmering av `SBLON` i ASUP.
- **Egenskaper hos undertryckningen av blockvis vid de olika blockvisa bearbetningstyperna**
Vid aktiv blockvis bearbetning SBL2 (stopp efter varje detaljprogramblock) stoppas **inte** i `SBLON`-blocket, när Bit 12 är inställt på "1" i MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK (förhindra enkelblockstopp).
Vid aktiv blockvis bearbetning SBL3 (stopp efter varje detaljprogramblock även i cykeln) undertrycks kommandot `SBLOF`.

Exempel

Exempel 1: Undertryckning av blockvis inom ett program

Programkod	Kommentar
N10 G1 X100 F1000	
N20 SBLOF	;Koppla från blockvis
N30 Y20	
N40 M100	
N50 R10=90	
N60 SBLON	; Koppla åter till blockvis
N70 M110	
N80 ...	

Området mellan N20 och N60 bearbetas i blockvis drift som ett steg.

Exempel 2: Cykel ska verka som ett kommando för användaren

Huvudprogram:

Programkod
N10 G1 X10 G90 F200
N20 X-4 Y6
N30 CYCLE1
N40 G1 X0
N50 M30

Cykel CYCLE1:

Programkod	Kommentar
N100 PROC CYCLE1 DISPLOF SBLOF	; Undertrycka blockvis
N110 R10=3*SIN(R20)+5	
N120 IF (R11 <= 0)	
N130 SETAL(61000)	
N140 ENDIF	
N150 G1 G91 Z=R10 F=R11	
N160 M17	

Cykeln CYCLE1 genomarbetas vid aktiv blockvis bearbetning, dvs. för bearbetningen av CYCLE1 måste Start-knappen tryckas ner en gång.

Exempel 3: En av PLC startad ASUP för aktivering av ändrad nollpunktsförflyttning och verktygskompenseringar ska inte vara synlig.

Programkod
N100 PROC NV SBLOF DISPLOF
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 _G55: 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

Exempel 4: Med MD10702 Bit 12 = 1 stoppas inte

Utgångssituation:

- Blockvis bearbetning är aktiv.
- MD10702 \$MN_IGNORE_SINGLEBLOCK_MASK Bit12 = 1

Huvudprogram:

Programkod	Kommentar
N10 G0 X0	; Stoppa i denna detaljprogramrad.
N20 X10	; Stoppa i denna detaljprogramrad.
N30 CYCLE	; Av cykel genererat förflyttningsblock.
N50 G90 X20	; Stoppa i denna detaljprogramrad.
M30	

Cykel CYCLE:

Programkod	Kommentar
PROC CYCLE SBLOF	; Undertrycka enkelblock-stopp.
N100 R0 = 1	
N110 SBLON	; På grund av MD10702 Bit12=1 stoppas inte i denna detaljprogramrad.
N120 X1	; I denna detaljprogramrad stoppas.
N140 SBLOF	
N150 R0 = 2	
RET	

Exempel 5: Undertryckning av blockvis vid programkapsling

Utgångssituation:

Blockvis bearbetning är aktiv.

Pogramkapsling:

Programkod	Kommentar
N10 X0 F1000	; I detta block stoppas.
N20 UP1(0)	
PROC UP1(INT _NR) SBLOF	; Undertrycka enkelblock-stopp.
N100 X10	
N110 UP2(0)	
PROC UP2(INT _NR)	
N200 X20	
N210 SBLON	; Koppla till enkelblock-stopp.
N220 X22	; I detta block stoppas.
N230 UP3(0)	
PROC UP3(INT _NR)	
N300 SBLOF	; Undertrycka enkelblock-stopp.
N305 X30	
N310 SBLON	; Koppla till enkelblock-stopp.
N320 X32	; I detta block stoppas.

Programkod	Kommentar
N330 SBLOF	; Undertrycka enkelblock-stopp.
N340 X34	
N350 M17	; SBLOF är aktiv.
N240 X24	; I detta block stoppas. SBLON är aktiv.
N250 M17	; I detta block stoppas. SBLON är aktiv.
N120 X12	
N130 M17	; I detta återhoppblock stoppas. SBLOF i PROC-anvisning är aktiv.
N30 X0	; I detta block stoppas.
N40 M30	; I detta block stoppas.

Ytterligare informationer

Enkelblockspärr för asynkrona underprogram

För att genomarbete en ASUP med enkelblock i ett steg, måste en PROC-anvisning med SBLOF programmeras i ASUP. Detta gäller också för funktionen "Editerbart system-ASUP" (MD11610 \$MN_ASUP_EDITABLE).

Exempel för ett editerbart system-ASUP:

Programkod	Kommentar
N10 PROC ASUP1 SBLOF DISPLOF	
N20 IF \$AC_ASUP=='H200'	
N30 RET	; Ingen REPOS vid driftslägesbyte.
N40 ELSE	
N50 REPOSA	; REPOS i alla övriga fall.
N60 ENDIF	

Programstyrningar i enkelblock

I enkelblockbearbetningen kan användaren genomarbete detaljprogrammet blockvis. Det existerar följande inställningstyper:

- SBL1: IPO-enkelblock med stopp efter varje maskinfunktionsblock.
- SBL2: Enkelblock med stopp efter varje block.
- SBL3: Stopp i cykeln (genom val av SBL3 undertrycks SBLOF-kommandot).

Undertryckning av blockvis vid programkapsling

Programmerades i ett underprogram SBLOF i PROC-anvisningen så stoppas på underprogramåterhopp med M17. Därmed förhindras att nästa block redan utförs i det anropande programmet. Aktiveras i ett underprogram SBLOF, utan SBLOF i PROC-anvisningen, en undertryckning av blockvis, stoppas först efter nästa maskinfunktionsblock i det anropande programmet. Är detta inte önskvärt, måste ännu före återhoppet (M17) SBLON åter programmeras i underprogrammet. Vid ett återhopp med RET stoppas inte i ett överordnat program.

2.24.2.6 Undertrycka aktuell blockindikering (DISPLOF, DISPLON, ACTBLOCNO)

I blockindikeringen visas starndadmässigt det aktuella programblocket. I cykler resp. underprogram kan indikeringen av det aktuella blocket undertryckas med kommandot DISPLOF. I stället för det aktuella blocket visas då anropet av cykeln resp. underprogrammet. Med kommandot DISPLON kan undertryckningen av blockindikeringen åter upphävas.

DISPLOF resp. DISPLON programmeras i programraden med PROC-anvisningen och verkar för hela underprogrammet och implicit för alla av detta underprogram anropade underprogram som inte innehåller något DISPLON- resp. DISPLOF-kommando. Detta beteende gäller också för ASUPs.

Syntax

```
PROC ... DISPLOF
PROC ... DISPLOF ACTBLOCNO
PROC ... DISPLON
```

Betydelse

DISPLOF:	Kommando för att undertrycka den aktuella blockindikeringen.	
	Placering:	I slutet av programraden med PROC-anvisningen
	Verkan:	Fram till återhopp från underprogrammet eller programslut.
	Observera: När ytterligare underprogram anropas från underprogrammet med DISPLOF-kommandot då undertrycks också i dessa underprogram den aktuella blockindikeringen, såvida inte DISPLON är explicit programmerat i dessa.	
DISPLON:	Kommando för att upphäva undertryckningen av den aktuella blockindikeringen	
	Placering:	I slutet av programraden med PROC-anvisningen
	Verkan:	Fram till återhopp från underprogrammet eller programslut.
	Observera: När ytterligare underprogram anropas från underprogrammet med DISPLON-kommandot då visas också i dessa underprogram det aktuella programblocket, såvida inte DISPLOF är explicit programmerat i dessa.	
ACTBLOCNO:	DISPLOF tillsammans med attributet ACTBLOCNO verkar så att i händelse av ett larm numret för det aktuella blocket i vilket larmet har uppkommit matas ut. Detta gäller också när endast DISPLOF har programmerats i en lägre programnivå. Vid DISPLOF utan ACTBLOCNO visas däremot blocknumret för cykel- resp. underprogramanropet från den sista programnivån som inte är markerad med DISPLOF.	

Exempel

Exempel 1: Undertrycka aktuell blockindikering i cykeln

Programkod	Kommentar
PROC CYCLE (AXIS TOMOV, REAL POSITION)	; Undertrycka aktuell blockindikering. I
SAVE DISPLOF	stället ska cykel-anropet visas t.ex.: CYC-
	LE(X,100.0)
DEF REAL DIFF	; Cykel-innehåll

Programkod	Kommentar
G01 ...	
...	
RET	; Underprogram-återhopp. I blockindikering- en visas det block som följer på cykel-anro- pet.

Exempel 2: Blockindikering vid larmutmatning

Underprogram SUBPROG1 (med ACTBLOCNO):

Programkod	Kommentar
PROC SUBPROG1 DISPLOF ACTBLOC-	
NO	
N8000 R10 = R33 + R44	
...	
N9040 R10 = 66 X100	; Utlösa larm 12080
...	
N10000 M17	

Underprogram SUBPROG2 (utan ACTBLOCNO):

Programkod	Kommentar
PROC SUBPROG2 DISPLOF	
N5000 R10 = R33 + R44	
...	
N6040 R10 = 66 X100	; Utlösa larm 12080
...	
N7000 M17	

Huvudprogram:

Programkod	Kommentar
N1000 G0 X0 Y0 Z0	
N1010 ...	
...	
N2050 SUBPROG1	; Larmutmatning = "12080 kanal K1 block N9040 syntax- fel vid text R10="
N2060 ...	
N2350 SUBPROG2	; Larmutmatning = "12080 kanal K1 block N2350 syntax- fel vid text R10="
...	
N3000 M30	

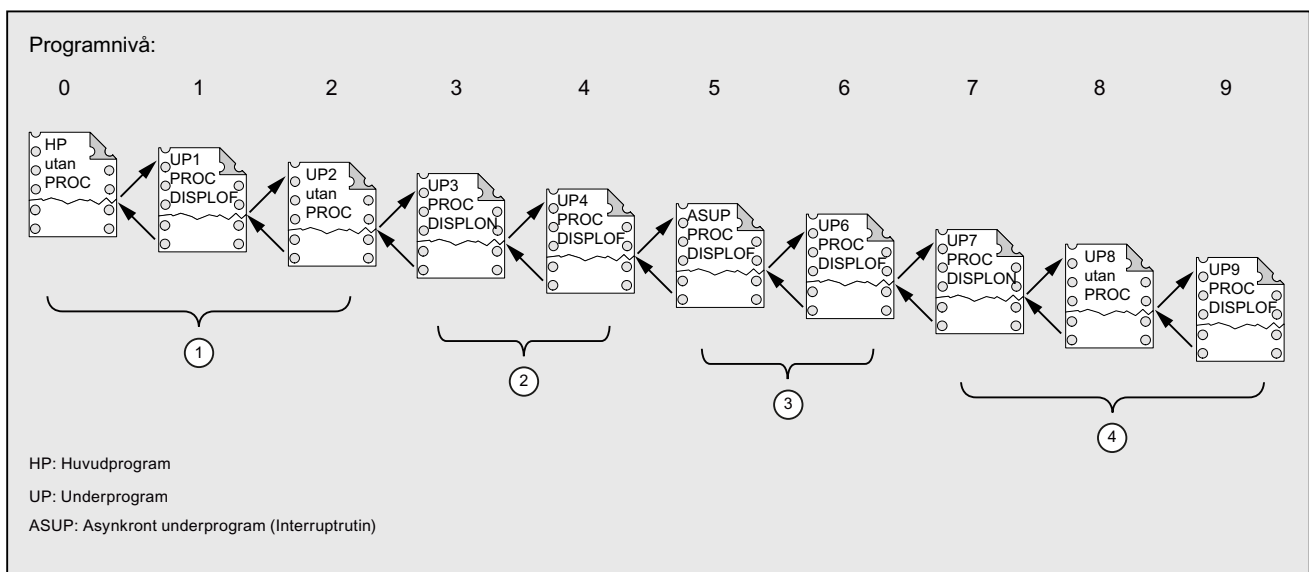
Exempel 3: Upphäva undertryckning av den aktuella blockindikeringen

Underprogram SUB1 med undertryckning:

Programkod	Kommentar
PROC SUB1 DISPLOF	; Undertrycka aktuell blockindikering i underprogrammet SUB1. I stället ska blocket med SUB1-anropet visas.
...	
N300 SUB2	; Anropa underprogram SUB2.
...	
N500 M17	

Underprogram SUB2 utan undertryckning:

Programkod	Kommentar
PROC SUB2 DISPLON	; Upphäva undertryckning av den aktuella blockindikeringen i underprogrammet SUB2.
...	
N200 M17	; Återhopp till underprogram SUB1. I SUB1 undertrycks den aktuella blockindikeringen åter.

Exempel 4: Indikeringsbeteende vid olika DISPLON/DISPLOF-kombinationer

- ① I den aktuella blockindikeringen visas detaljprogramraderna från programnivå 0.
- ② I den aktuella blockindikeringen visas detaljprogramraderna från programnivå 3.
- ③ I den aktuella blockindikeringen visas detaljprogramraderna från programnivå 3.
- ④ I den aktuella blockindikeringen visas detaljprogramraderna från programnivå 7/8.

2.24.2.7 Känneteckna underprogram med förberedelse (PREPRO)

Med nyckelordet `PREPRO` kan under start alla filer kännetecknas i slutet av `PROC`-anvisningsraden.

Märk

Denna typ av programförberedelse är beroende av inställningen av det tillhörande maskindatumet. Följ maskintillverkarens uppgifter.

Litteratur:

Funktionshandbok Specialfunktioner; Förbearbetning (V2)

Syntax

```
PROC ... PREPRO
```

Betydelse

PREPRO:	Nyckelord för känneteckning av alla under start förberedda filer i de i cykelkatalogerna lagrade NC-programmen
---------	--

Läsa in underprogram med förberedelse och underprogramanrop

Både under start förberedda underprogram med parametrar och vid underprogramanrop behandlas cykelkatalogerna i samma ordningsföljd:

1. `_N_CUS_DIR` användarcykler
2. `_N_CMA_DIR` tillverkarcykler
3. `_N_CST_DIR` standardcykler

I händelse av NC-program med samma namn men med olika utformning aktiveras den först hittade `PROC`-anvisningen och den andra `PROC`-anvisningen läses över utan larmmeddelande.

2.24.2.8 Underprogramåterhopp M17

I slutet av ett underprogram står återhopp-kommandot `M17` (resp. detaljprogramslut-kommandot `M30`). Det förorsakar återhopp till det program som anropar på detaljprogramblocket efter underprogramanropet.

Märk

`M17` och `M30` behandlas likvärdigt i NC-språket.

Syntax

```
PROC <Programmname>
...
```

M17/M30

Randvillkor

Återverkan av underprogramåterhoppet på banstyrningsdriften

Står M17 (resp. M30) ensamt i detaljprogramblocket stoppas därigenom en i kanalen aktiv banstyrningsdrift.

För att undvika att banstyrningsdriften stoppas ska M17 (resp. M30) skrivas med i det sista förflytningsblocket. Dessutom måste följande maskindatum vara inställt på "0":

MD20800 \$MC_SPF_END_TO_VDI = 0 (ingen M30/M17-utmatning vid NC/PLC-gränssnittet)

Exempel

1. Underprogram med M17 i eget block

Programkod	Kommentar
N10 G64 F2000 G91 X10 Y10	
N20 X10 Z10	
N30 M17	; Återhopp med stopp i banstyrningsdriften.

2. Underprogram med M17 i det sista förflytningsblocket

Programkod	Kommentar
N10 G64 F2000 G91 X10 Y10	
N20 X10 Z10 M17	; Återhopp utan stopp i banstyrningsdriften.

2.24.2.9 Underprogramåterhopp RET

Som ersättning för återhoppskommandot M17 kan också kommandot RET användas i underprogrammet. RET måste programmeras i ett eget detaljprogramblock. Som M17 förorsakar RET återhopp till det program som anropar på detaljprogramblocket efter underprogramanropet.

Märk

Genom programmeringen av parametrar kan återhoppsbeteendet för RET ändras (se "Parametrebart underprogramåterhopp (RET ...) (Sida 176)").

Användning

RET-anvisningen skall användas när en G64-banstyrningsdrift (G641 ... G645) inte ska stoppas genom återhoppet.

Förutsättning

Kommandot `RET` kan endast användas i underprogram som inte definierades med attributet `SAVE`.

Syntax

```
PROC <Programmname>
...
RET
```

Exempel

Huvudprogram:

Programkod	Kommentar
PROC MAIN_PROGRAM	; Programbörjan
...	
N50 SUB_PROG	; Underprogramanrop: SUB_PROG
N60 ...	
...	
N100 M30	; Programslut

Underprogram:

Programkod	Kommentar
PROC SUB_PROG	
...	
N100 RET	; Återhopp görs till block N60 i huvudprogrammet.

2.24.2.10 Parametrerbart underprogramåterhopp (RET ...)

I allmänhet hoppas med kommandot `RET` från ett underprogram tillbaka till det anropande programmet. Bearbetningen fortsätts sedan med den programrad som följer på underprogramanropet. Ska programbearbetningen fortsättas på ett annat stället står följande möjligheter till förfogande:

- Fortsättning av programbearbetningen efter anrop av avspåningscyklerna i ISO-dialekt-mode (efter konturbeskrivningen).
- Återhopp till huvudprogrammet från en valfri underprogramnivå (även efter ASUP) vid felhantering.
- Återhopp över flera programnivåer för speciella användningar i compilecykler och i ISO-dialekt-mode.

För detta ska kommandot `RET` programmeras med ytterligare parametrar.

<Satz nach Zielsatz>:	I parametern anges, om programbearbetningen ska fortsätta i det under parametern <Zielsatz> angivna blocket eller i det därpå följande blocket.			
	Typ:	INT		
	Värde:	0	Återhoppet sker till det i parametern <Ziel-satz> angivna blocket.	
> 0		Återhoppet sker till det i parametern <Ziel-satz> angivna efterföljande blocket.		
<Anzahl der Rücksprungebenen>:	I parametern anges det antal programnivåer, som ska hoppas tillbaka, för att där söka målblocket och fortsätta programbearbetningen.			
	Typ:	INT		
	Värde:	1	Programmet fortsätts i den "aktuella programnivån - 1" (som RET utan parameter).	
		2	Programmet fortsätts i den "aktuella programnivån - 2" dvs. en nivå hoppas över.	
		3	Programmet fortsätts i den "aktuella programnivån - 3" dvs. två nivåer hoppas över.	
		...		
Värdeområde:	1 ... 15			
<Rücksprung auf Programmanfang>:	I parametern anges om vid ett återhopp till huvudprogrammet och där aktiv ISO-dialekt-mode programmet ska fortsättas vid programbörjan.			
	Typ:	BOOL		
	Värde:	1	När återhoppet görs till huvudprogrammet och där ett ISO-dialekt-mode är aktivt, förgrenas till programbörjan.	

Märk

Vid ett underprogramåterhopp med en teckenföljd som uppgift för målblocksökningen söks i det anropande programmet alltid först efter ett hoppmärke.

När ett hoppmål ska vara entydigt definierat av en teckenföljd får teckenföljden därför inte överensstämma med namnet på ett hoppmärke eftersom annars underprogramåterhoppet alltid utförs på hoppmärket och inte på teckenföljden (se exempel 2).

Randvillkor

Vid återhopp över flera programnivåer utvärderas **SAVE**-anvisningarna för de enskilda programnivåerna.

Är vid ett återhopp över flera programnivåer ett modalt underprogram aktivt och är i ett av de överhoppade underprogrammen bortvalskommandot `MCALL` programmerat för det modala underprogrammet, förblir det modala underprogrammet fortfarande aktivt.

OBSERVERA

Programmeringsfel

Vid ett återhopp över flera programnivåer är användaren ensam ansvarig att sörja för att det fortsätts med de nödvändiga modala inställningarna. Detta kan t.ex. uppnås genom programmering av ett motsvarande huvudblock.

Exempel

Exempel 1: Återlandning i huvudprogrammet efter ASUP-bearbetning

Programmering	Kommentar
N10010 CALL "UP1"	; Programnivå 0 (huvudprogram)
N11000 PROC UP1	; Programnivå 1
N11010 CALL "UP2"	
N12000 PROC UP2	; Programnivå 2
...	
N19000 PROC ASUP	; Programnivå 3 (ASUP-bearbetning)
...	
N19100 RET("N10900", , \$P_STACK)	; Underprogramåterhopp till huvudprogrammet
	; \$P_STACK: aktuell programnivå
N10900	; Målblock i huvudprogrammet
N10910 MCALL	; Koppla från modalt underprogramanrop
N10920 G0 G60 G40 M5	; Initiera ytterligare modala inställningar

Exempel 2: Teckenföljd (<String>) som uppgift för målblocksökningen

Huvudprogram:

Programkod	Kommentar
PROC MAIN_PROGRAM	
N1000 DEF INT iVar1=1, iVar2=4	
N1010 ...	
N1200 subProg1	; Anrop från underprogram "subProg1"
N1210 M2 S1000 X10 F1000	
N1220	
N1400 subProg2	; Anrop från underprogram "subProg2"
N1410 M3 S500 Y20	
N1420 ..	
N1500 lab1: iVar1=R10*44	
N1510 F500 X5	

Programkod	Kommentar
N1520 ...	
N1550 subprog1: G1 X30	; "subProg1" är här definierat som hoppmärke.
N1560 ...	
N1600 subProg3	; Anrop från underprogram "subProg3"
N1610 ...	
N1900 M30	

Underprogram subProg1:

Programkod	Kommentar
PROC subProg1	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg2")	; Återhopp till huvudprogrammet till block N1400

Underprogram subProg2:

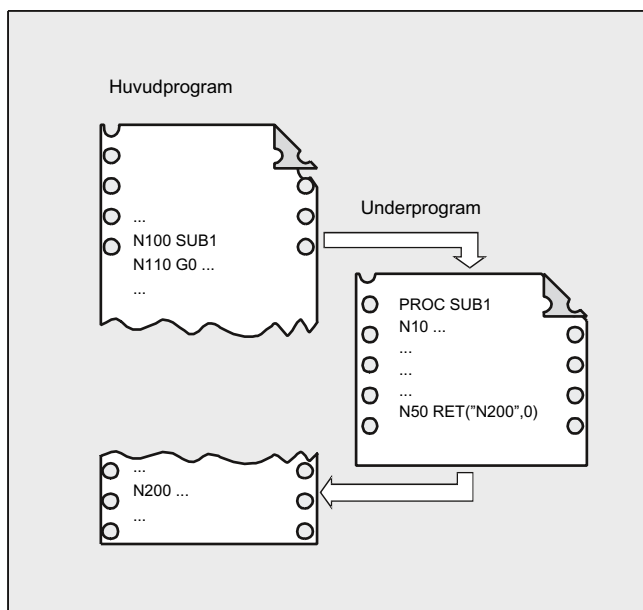
Programkod	Kommentar
PROC subProg2	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("iVar1")	; Återhopp till huvudprogrammet till block N1500

Underprogram subProg3:

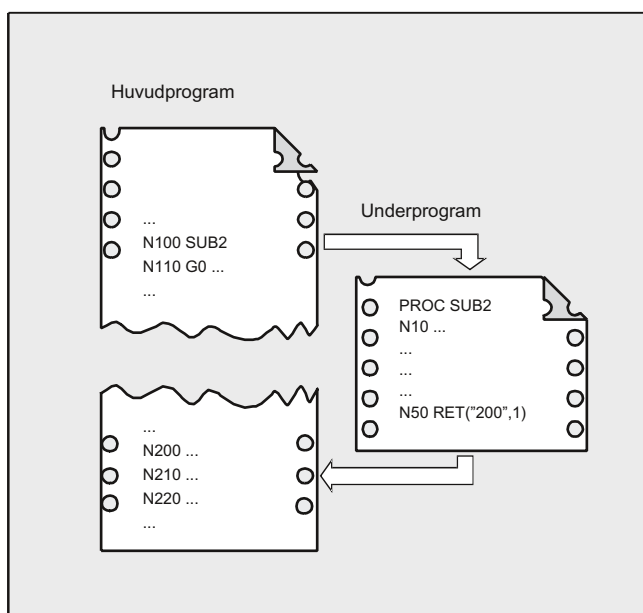
Programkod	Kommentar
PROC subProg3	
N2000 R10=R20+100	
N2010 ...	
N2200 RET("subProg1")	; Återhopp till huvudprogrammet till block N1550

Ytterligare informationer

Följande grafik visar de olika följderna av återhoppsparametrarna

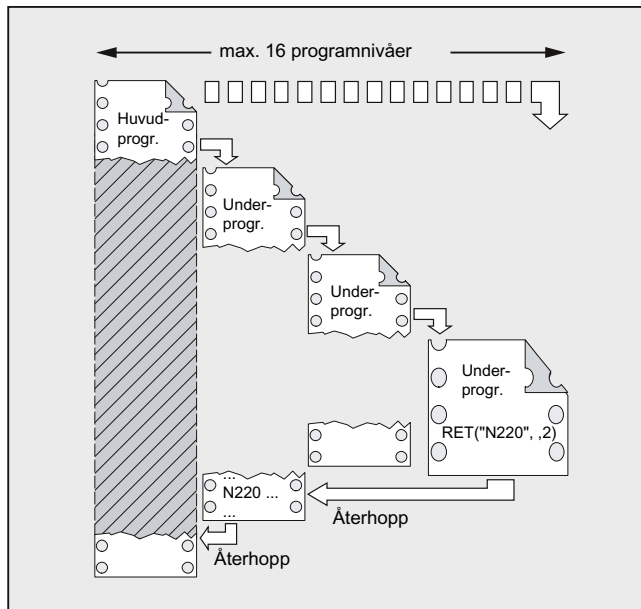
1. <Zielsatz> = "N200", <Satz nach Zielsatz> = 0

Efter RET-kommandot fortsätts programbearbetningen med blocket N200 i huvudprogrammet.

2. <Zielsatz> = "N200", <Satz nach Zielsatz> = 1

Efter RET-kommandot fortsätts programbearbetningen med det block (N210) som följer efter blocket N200 i huvudprogrammet.

3. <Zielsatz> = "N220", <Anzahl der Rücksprungebenen> = 2



Efter RET-kommandot hoppas två programnivåer över och programbearbetningen fortsätts med block N220.

2.24.2.11 Parametrerbart underprogramåterhopp (RETB ...)

I allmänhet hoppas med kommandot RETB från ett underprogram tillbaka till det anropande programmet. Bearbetningen fortsätts sedan med den programrad som följer på underprogramanropet. Ska programbearbetningen fortsättas på ett annat stället står följande möjligheter till förfogande:

- Fortsättning av programbearbetningen efter anrop av avspåningscyklerna i ISO-dialekt-mode (efter konturbeskrivningen).
- Återhopp till huvudprogrammet från en valfri underprogramnivå (även efter ASUP) vid felhantering.
- Återhopp över flera programnivåer för speciella användningar i compilecykler och i ISO-dialekt-mode.

För detta ska kommandot RETB programmeras med ytterligare parametrar.

Sökriktning

Vid angivande av parametern <Zielsatz> sker återhoppet först till blocket efter anropsblocket. Sedan söks målet i riktning mot program**början** för det program till vilket återhoppet görs. Är sökningen inte framgångsrik, söks härnäst i riktning mot programslutet.

Syntax

```
RETB("<Zielsatz>")
RETB("<Zielsatz>",<Satz nach Zielsatz>)
RETB("<Zielsatz>",<Satz nach Zielsatz>,<Anzahl der Rücksprungebenen>)
RETB("<Zielsatz>",<Anzahl der Rücksprungebenen>)
```

```

RETB("<Zielsatz>",<Satz nach Zielsatz>,<Anzahl der Rücksprungebenen>,
<Rücksprung auf Programmanfang>)
RETB( , ,<Anzahl der Rücksprungebenen>,<Rücksprung auf Programmanfang>)

```

Betydelse

RETB:	Underprogramslut			
<Zielsatz>:	I parametern anges det block som hoppmål vid vilket programbearbetningen ska fortsättas. Är parametern <Anzahl der Rücksprungebenen> inte programmerad då befinner sig hoppmålet i det program från vilket det aktuella underprogrammet anropades. Möjliga uppgifter är:			
	<Satznummer>	Nummer på målblocket. Sökningen efter blocknumret sker i det program till vilket återhoppas först i riktning programbörjan.		
	<Sprungmarke>:	Hoppmärke som måste finnas i det program till vilket återhoppas. Sökningen efter hoppmärket sker i det program till vilket återhoppas först i riktning programbörjan.		
	<Zeichenkette>:	Teckenföljd som måste finnas i det program till vilket återhoppas (t. ex. program- eller variabelnamn). Sökningen efter teckenföljden sker i det program till vilket återhoppas först i riktning programbörjan. För programmeringen av teckenföljden i målblocket gäller följande regler: • Mellanslag i slutet (till skillnad från hoppmärket som är kännetecknat av ett ":" på slutet). • Före teckenföljden får endast ett blocknummer och/eller hoppmärke vara satt, inga programkommandon.		
<Satz nach Zielsatz>:	I parametern anges, om programbearbetningen ska fortsätta i det under parametern <Zielsatz> angivna blocket eller i det därpå följande blocket.			
	Typ:	INT		
	Värde:	0	Återhoppet sker till det i parametern <Zielsatz> angivna blocket.	
		> 0	Återhoppet sker till det i parametern <Zielsatz> angivna efterföljande blocket.	

<Anzahl der Rücksprungebenen>:	I parametern anges det antal programnivåer, som ska hoppas tillbaka, för att där söka målblocket och fortsätta programbearbetningen.		
	Typ:	INT	
	Värde:	1	Programmet fortsätts i den "aktuella programnivån - 1" (som <code>RET</code> utan parameter).
		2	Programmet fortsätts i den "aktuella programnivån - 2" dvs. en nivå hoppas över.
		3	Programmet fortsätts i den "aktuella programnivån - 3" dvs. två nivåer hoppas över.
		...	
Värdeområde:	1 ... 15		
<Rücksprung auf Programmanfang>:	I parametern anges om vid ett återhopp till huvudprogrammet och där aktiv ISO-dialekt-mode programmet ska fortsättas vid programbörjan.		
	Typ:	BOOL	
	Värde:	1	När återhoppet görs till huvudprogrammet och där ett ISO-dialekt-mode är aktivt, förgrenas till programbörjan.

Märk

Vid ett underprogramåterhopp med en teckenföljd som uppgift för målblocksökningen söks i det anropande programmet alltid först efter ett hoppmärke.

När ett hoppmål ska vara entydigt definierat av en teckenföljd får teckenföljden därför inte överensstämja med namnet på ett hoppmärke eftersom annars underprogramåterhoppet alltid utförs på hoppmärket och inte på teckenföljden (se exempel 2).

Randvillkor

Vid återhopp över flera programnivåer utvärderas `SAVE`-anvisningarna för de enskilda programnivåerna.

Är vid ett återhopp över flera programnivåer ett modalt underprogram aktivt och är i ett av de överhoppade underprogrammen bortvalskommandot `MCALL` programmerat för det modala underprogrammet, förblir det modala underprogrammet fortfarande aktivt.

OBSERVERA**Programmeringsfel**

Vid ett återhopp över flera programnivåer är användaren ensam ansvarig att sörja för att det fortsätts med de nödvändiga modala inställningarna. Detta kan t.ex. uppnås genom programmering av ett motsvarande huvudblock.

Exempel

Programkod	Kommentar
BEISPIEL.MPF	
...	
N3000 START_CYC(param1, 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	

Programkod	Kommentar
PROC END_CYC(...)	; Anrop i huvudprogrammet rad N3040
N10000 ...	
N15000 if status == 1	
N15010 RETB("START_CYC")	; Återhopp till anropande program BEISPIEL.MPF ; Sökning efter teckenföljd "START_CYC" ; Sökriktning bakåt riktning programbörjan) ; Fortsättning av programbearbetningen med rad N3000
N15020 endif	
N15030 if status == 0	
N15040 RET	; Återhopp till anropande program BEISPIEL.MPF ; Fortsättning av programbearbetningen med rad N3050
N15050 endif	
N16000 RET("START_CYC")	; Återhopp till anropande program BEISPIEL.MPF ; Sökning efter teckenföljd "START_CYC" ; Sökriktning framåt riktning programslut ; Fortsättning av programbearbetningen med rad N4500
N17060 RETB	; Återhopp till anropande program BEISPIEL.MPF ; Fortsättning av programbearbetningen med rad N3050 ; RETB utan parameter är identisk med RET

2.24.3 Anrop av ett underprogram

2.24.3.1 Underprogramanrop utan parameteröverföring

Anropet av ett underprogram görs antingen med adressen L och underprogramnummer eller genom angivande av programnamnet.

Även ett huvudprogram kan anropas som underprogram. Det i huvudprogrammet inställda programslutet M2 eller M30 värderas i detta fall som M17 (programs lut med återhopp till det anropande programmet).

Märk

Motsvarande kan ett underprogram också startas som huvudprogram.

Styrningens sökstrategi:

Finns det *_MPF?

Finns det *_SPF?

Därav följer: Om namnet för det underprogram som ska anropas är identiskt med namnet på huvudprogrammet då anropas åter det anropande huvudprogrammet. Denna som regel icke önskvärda effekt måste undvikas genom entydigt namnval för underprogram och huvudprogram.

Märk

Underprogram som inte kräver någon parameteröverföring kan också anropas från en initieringsfil.

Syntax

L<Nummer>/<Programmname>

Märk

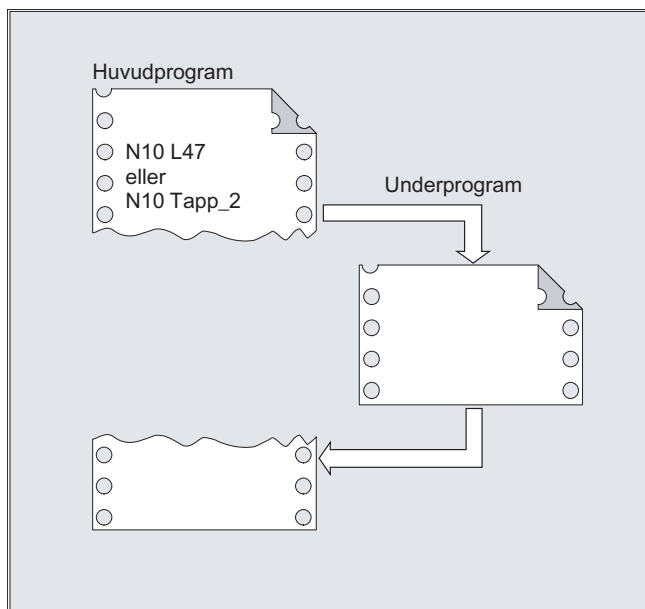
Anropet av ett underprogram måste alltid programmeras i eget NC-block.

Betydelse

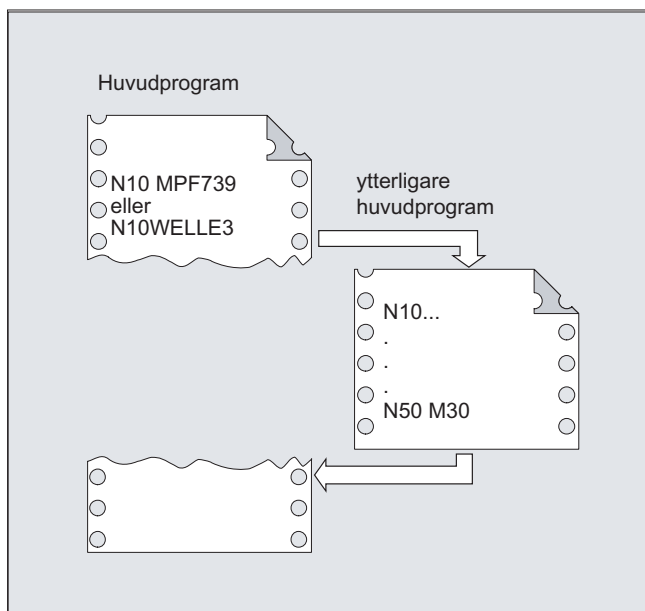
L:	Adress för underprogramanrop	
<Nummer>:	Nummer på underprogrammet	
	Typ:	INT
	Värde:	Maximalt 7 decimalsiffror Viktigt: Ledande nollor är av betydelse vid namngivningen (⇒ L123, L0123 och L00123 är tre olika underprogram).
<Programmname>:	Namn på underprogrammet (eller huvudprogrammet)	

Exempel

Exempel 1: Anrop av ett underprogram utan parameteröverföring



Exempel 2: Anrop av ett huvudprogram som underprogram



Se även

Underprogram utan parameteröverföring (Sida 160)

2.24.3.2 Underprogramanrop med parameteröverföring (EXTERN)

Vid underprogramanrop med parameteröverföring kan variabler eller värden överföras direkt (inte vid VAR-parametrar).

Underprogram med parameteröverföring måste före anropet i huvudprogrammet tillkännages med EXTERN (t.ex. vid programbörjan). Angivet blir därvid namnet på underprogrammet och variabeltyperna med ordningsföljden i överföringen.

OBSERVERA

Förväxlingsfara

Både variabeltyperna och även ordningsföljden för överföringen måste överensstämma med definitionerna som blev överenskomna i underprogrammet under PROC. Parameternamnen kan vara olika i huvud- och underprogram.

Syntax

```
EXTERN <Programmname>(<Typ_Par1>,<Typ_Par2>,<Typ_Par3>)
...
<Programmname>(<Wert_Par1>,<Wert_Par2>,<Wert_Par3>)
```

Märk

Underprogramanropet måste alltid programmeras i eget NC-block.

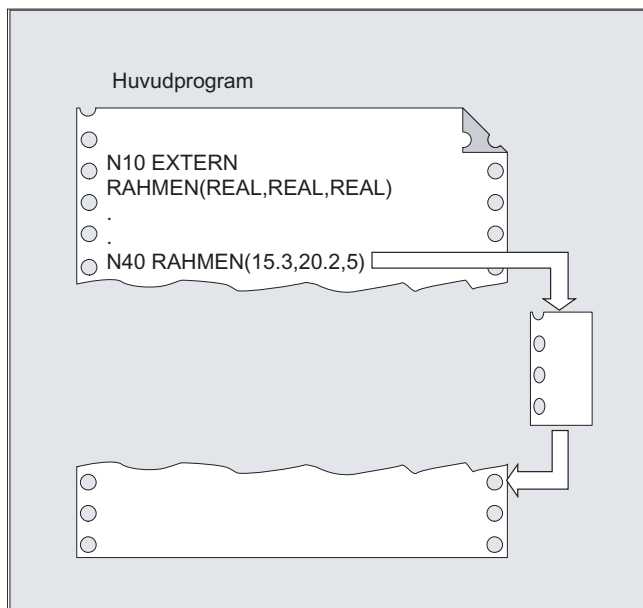
Betydelse

<Programmname>:	Namn på underprogrammet
EXTERN:	Nyckelord för tillkännagivande av ett underprogram med parameteröverföring Observera: EXTERN måste endast angivas när underprogrammet står i arbetsstycks- eller i den globala underprogramkatalogen. Cykler måste inte förklaras som EXTERN.
<Typ_Par1>,<Typ_Par2>,<Typ_Par3>:	Variabeltyper för parametrarna som ska överlämnas med ordningsföljden i överföringen
<Wert_Par1>,<Wert_Par2>,<Wert_Par3>:	Variabelvärden för de parametrar som ska överföras

Exempel

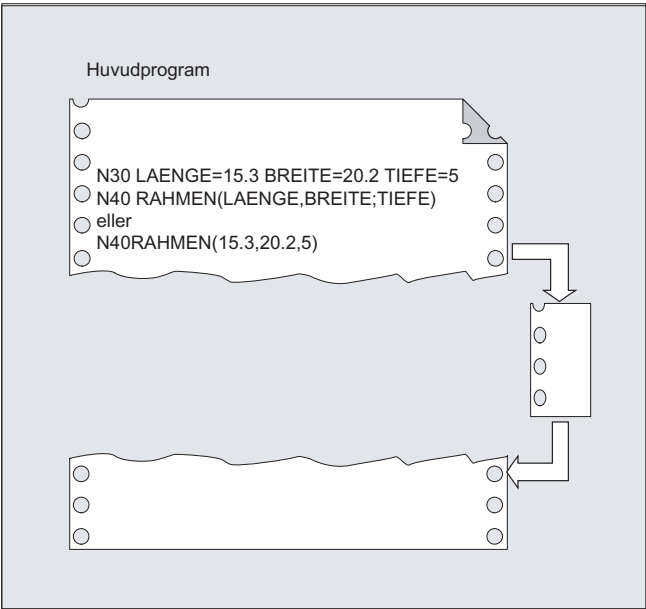
Exempel 1: Underprogramanrop med tidigare tillkännagivande

Programkod	Kommentar
N10 EXTERN RAHMEN (REAL,REAL,REAL)	; Angivande av underprogrammet.
...	
N40 RAHMEN(15.3,20.2,5)	; Anrop av underprogrammet med parameteröverföring.



Exempel 2: Underprogramanrop utan tillkännagivande

Programkod	Kommentar
N10 DEF REAL LAENGE, BREITE, TIEFE	
N20 ...	
N30 LAENGE=15.3 BREITE=20.2 TIEFE=5	
N40 RAHMEN (LAENGE,BREITE,TIEFE)	; eller: N40 RAHMEN(15.3,20.2,5)



Se även

Underprogram med parameteröverföring Call-by-Value (PROC) (Sida 160)

Underprogram med parameteröverföring Call-by-Reference (PROC, VAR) (Sida 162)

2.24.3.3 Antal programupprepningar (P)

Ska ett underprogram genomarbetas flera gånger efter varandra kan i blocket med underprogramanropet det önskade antalet programupprepningar programmeras under adressen P.

 **SE UPP**

Underprogramanrop med programupprepning och parameteröverföring

Parametrar överförs endast vid programanrop resp. första körningen. För de fortsatta upprepningarna förblir parametrarna oförändrade. Om du vill förändra parametrarna vid programupprepningarna måste du fastlägga motsvarande överenskommelser i underprogrammet.

Syntax

<Programmname> P<Wert>

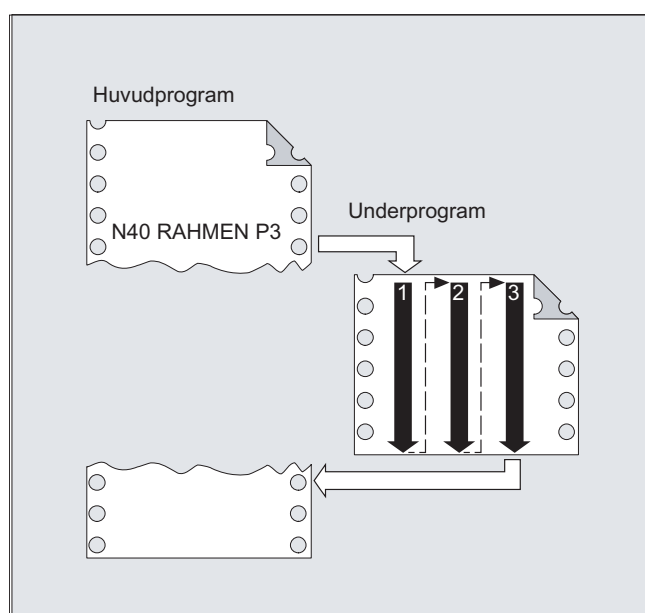
Betydelse

<Programmname>:	Underprogramanrop
P:	Adress för programmering av programupprepningar

<Wert>:	Antal programupprepningar	
	Typ:	INT
	Värdeområde:	1 ... 9999 (utan förtecken)

Exempel

Programkod	Kommentar
...	
N40 RAHMEN P3	; Underprogrammet RAHMEN ska genomarbetas tre gånger efter varandra.
...	



2.24.3.4 Modalt underprogramanrop (MCALL)

Med det modala underprogramanropet `MCALL(<Programmname>)` anropas det angivna underprogrammet inte genast. I stället görs anropet från denna tidpunkt i detaljprogrammet automatiskt efter varje förflytningsblock med banrörelser. Även utöver programnivåer.

Märk

I ett programförlopp verkar alltid endast det sista modala underprogramanropet `MCALL(<Programmname>)`. Det aktuella modala underprogramanropet ersätter det hittills aktiva.

Överförs parametrar till underprogrammet, sker parameteröverföringen endast vid anropet `MCALL(<Programmname>(Par1, Par2, ...))`.

OBSERVERA**Modala underprogramanrop utan banrörelse**

Det modala underprogrammet anropas i följande situationer också utan programmering av en banrörelse:

- Programmering av adresserna *S* eller *F* när *G0* eller *G1* är aktiv.
- När *G0* eller *G1* står ensamt i blocket eller tillsammans med ytterligare *G*-kommandon.

Syntax

```
MCALL <Programname>
...
MCALL
```

Betydelse

MCALL <Programname>:	Koppla till funktion "Modalt underprogramanrop"
<Programname>:	Namn på underprogrammet
MCALL:	Med MCALL utan angivande ett programnamn kopplas funktionen "Modalt underprogramanrop" från.

Randvillkor**ASUP**

Stoppas bearbetningen av ett detaljprogram av en ASUP (se Kapitel "Interruptrutin (ASUP) (Sida 124)") utförs i denna ASUP inga modala underprogramanrop.

Startas en ASUP i kanaltillståndet "Reset" förhåller sig den beträffande de modala underprogramanropen som ett normalt detaljprogram.

; Verktygsväxlingscykel

Sker i verktygsväxlingscykeln ett bortval av funktionen "Modalt underprogramanrop", så ska beaktas att verktygsväxlingscykeln ev. även efter blocksökningen anropas implicit via ASUP-sökning eller manuellt med överföring. I denna situation får funktionen "Modalt underprogramanrop" inte väljas bort eftersom annars sökningsresultatet blir förfalskat. Det rekommenderas därför, att programmera bortvalet av funktionen "Modalt underprogramanrop" i verktygsväxlingscykeln på följande sätt:

Programkod	Kommentar
...	
IF \$AC_ASUP == 0	; Anrop sker inte via ASUP-sökning eller överföring.
MCALL	; Koppla från funktionen "Modalt underprogramanrop".
ENDIF	

Programkod	Kommentar
...	

Exempel

Exempel 1

Programkod	Kommentar
N10 G0 X0 Y0	
N20 MCALL L70	; Koppla till modalt underprogramanrop för L70.
N30 X10 Y10	; X10 Y10 uppsöks, sedan anropas L70.
N40 X20 Y20	; X20 Y20 uppsöks, sedan anropas L70.
...	
N100 MCALL	; Koppla från funktionen "Modalt underprogramanrop".
N110 X0 Y0	; X0 Y0 uppsöks, L70 anropas inte .

Exempel 2

Programkod
N10 G0 X0 Y0
N20 MCALL L70
N30 L80

I detta exempel står de efterföljande NC-blocken med programmerade banaxlar i underprogram L80. L70 anropas av L80.

2.24.3.5 Indirekt underprogramanrop (CALL)

Beroende av de givna villkoren kan vid ett ställe olika underprogram anropas. För detta lagras namnet på underprogrammet i en variabel av typ STRING. Underprogramanropet görs med CALL och variabelnamnet.

Märk

Det indirekta underprogramanropet är möjligt endast för underprogram utan parameteröverföring. För det direkta anropet av ett underprogram lägger du namnet i en STRING-konstant.

Syntax

CALL <Programmname>

Betydelse

CALL:	Kommando för det indirekta underprogramanropet	
<Programmname>:	Namn på underprogrammet (variabel eller konstant)	
	Typ:	STRING

Exempel

Direkt anrop med STRING-konstant:

Programkod	Kommentar
...	
CALL "/_N_WKS_DIR/_N_SUBPROG_WPD/_N_TEIL1_SPF"	; Anropa underprogram TEIL1 direkt med CALL.
...	

Indirekt anrop via variabel:

Programkod	Kommentar
...	
DEF STRING[100] PROGNAME	; Definiera variabel.
PROGNAME="/_N_WKS_DIR/_N_SUBPROG_WPD/_N_TEIL1_SPF"	; Tillordna underprogram TEIL1 variabeln PROGNAME.
CALL PROGNAME	; Indirekt anropa underprogram TEIL1 via CALL och variabeln PROGNAME.
...	

2.24.3.6

Indirekt underprogramanrop med uppgift över den programdel som ska utföras (CALL BLOCK ... TO ...)

Med CALL och nyckelordskombinationen BLOCK ... TO anropas ett underprogram indirekt och den med Start- och Slutmärke markerade programdelen utförs.

Syntax

```
CALL <Programmname> BLOCK <Startmarke> TO <Endmarke>
CALL BLOCK <Startmarke> TO <Endmarke>
```

Betydelse

CALL:	Kommando för det indirekta underprogramanropet	
<Programmname>:	Namn på underprogrammet (variabel eller konstant) som innehåller den programdel som ska bearbetas (Uppgift optional).	
	Typ:	STRING
	Observera: Är inget <Programmname> programmerat söks och utförs den med <Startmarke> och <Endmarke> markerade programdelen i det aktuella programmet.	
BLOCK ... TO ... :	Nyckelordskombination för indirekt programdelsutförande	
<Startmarke>:	Variabel som hänvisar till början av den programdel som ska bearbetas.	
	Typ:	STRING

<Endmarke>:	Variabel som hänvisar till slutet på den programdel som ska bearbetas.	
	Typ:	STRING

Exempel

Huvudprogram:

Programkod	Kommentar
...	
DEF STRING[20] STARTLABEL, ENDLABEL	; Variabeldefinition för start- och slut-
	märke.
STARTLABEL="LABEL_1"	
ENDLABEL="LABEL_2"	
...	
CALL "CONTUR_1" BLOCK STARTLABEL TO ENDLA-	; Indirekt underprogramanrop och marker-
BEL	ing av den programdel som ska utföras.
...	

Underprogram:

Programkod	Kommentar
PROC CONTUR_1 ...	
LABEL_1	; Startmärke: Början av programdelsutförandet
N1000 G1 ...	
...	
LABEL_2	; Slutmärke: Slut på programdelsutförandet
...	

2.24.3.7 Indirekt anrop av ett i ISO-språk programmerat program (ISOCALL)

Med det indirekta programanropet **ISOCALL** kan ett i ett ISO-språk programmerat program anropas. Därvid aktiveras det i maskindata inställda ISO-monet. Vid programslut blir det ursprungliga bearbetningsmodet åter verksamt. Är inget ISO-mode inställt i maskindata görs anropet av underprogrammet i Siemens-mode.

Ytterligare informationer till ISO-mode se:

Litteratur:

Funktionsbeskrivning ISO-dialekter

Syntax

ISOCALL <Programmname>

Betydelse

ISOCALL:	Nyckelord för indirekt underprogramanrop med vilket det i maskindata inställda ISO-modet aktiveras
<Programmname>:	Namn på det i ett ISO-språk programmerade programmet (variabel eller konstant av typ STRING)

Exempel: Anrop kontur med cykelprogrammering utifrån ISO-mode

Programkod	Kommentar
0122_SPF	; Konturbeskrivning i ISO-mode
N1010 G1 X10 Z20	
N1020 X30 R5	
N1030 Z50 C10	
N1040 X50	
N1050 M99	
N0010 DEF STRING[5] PROGNAME = "0122"	; Siemens-detaljprogram(-cykel)
...	
N2000 R11 = \$AA_IW[X]	
N2010 ISOCALL PROGNAME	
N2020 R10 = R10+1	; Bearbeta program 0122.spf i ISO-mode
...	
N2400 M30	

2.24.3.8 Anropa underprogram med sökvägsuppgift och parametrar (PCALL)

Med PCALL kan underprogram anropas med absolut sökvägsuppgift och parameteröverföring.

Syntax

PCALL <Pfad/Programmname>(<Parameter 1>,...,<Parameter n>)

Betydelse

PCALL:	Nyckelord för underprogramanrop med absolut sökvägsuppgift.
<Pfad/Programmname>:	Absolut sökvägsuppgift inklusive underprogramnamn Regler för sökvägsuppgift se "Adressering av filer i programminnet (Sida 213)". Angavs ingen absolut sökväg, förhåller sig PCALL som ett standard-underprogramanrop med programbeteckning. Programnamnet anges utan prefix och utan fil-identifikation. Ska programnamnet programmeras med prefix och fil-identifikation, så måste det förklaras explicit med prefix och fil-identifikation med kommandot EXTERN.
<Parameter 1>, ...:	Aktual-parameter enligt PROC-anvisningen i underprogrammet.

Exempel

Programkod
PCALL/_N_WKS_DIR/_N_WELLE_WPD/WELLE(parameter1,parameter2,...)

2.24.3.9 Utvidga sökvägen vid underprogramanrop (CALLPATH)

Med kommandot `CALLPATH` kan sökvägen för underprogramanrop utvidgas. Därmed kan också underprogram anropas från en arbetsstyckskatalog som inte valts utan att ange det fullständiga, absoluta sökvägsnamnet för underprogrammet.

En ytterligare användningsmöjlighet erbjuder sig i EES-driftmode "EES utan GDIR", när en katalog på ett externt programminne används för att lagra globala underprogram. I detta fall kan sökvägen utökas med `CALLPATH` med denna underprogramkatalog.

Sökvägsutvidgningen görs före inmatningen för användarcyklar (`_N_CUS_DIR`).

Genom följande händelser väljs sökvägsutvidgningen åter bort:

- `CALLPATH` med mellanslag
- `CALLPATH` utan parameter
- Detaljprogramslut
- Reset

Syntax

`CALLPATH("<Pfadname>")`

Betydelse

CALLPATH:	Nyckelord för den programmerbara sökvägsutvidgningen. Programmeras i en egen detaljprogramrad.
<Pfadname>:	Konstant eller variabel av typ STRING. Innehåller den absoluta sökvägsuppgiften för katalogen med vilken sökvägen ska utvidgas. Regler för sökvägsuppgift se "Adressering av filer i programminnet (Sida 213)".

Exempel

Sökvägen ska utvidgas med en bestämd arbetsstyckskatalog:

Programkod
...
CALLPATH("/_N_WKS_DIR/_N_MYWPD_WPD")
...

Därmed ställs följande sökväg in (position 5. är ny):

1. Aktuell katalog/*name*
2. Aktuell katalog/*name_SPF*
3. Aktuell katalog/*name_MPF*
4. //NC:/_N_SPF_DIR/*name_SPF*
5. /_N_WKS_DIR/_N_MYWPD_WPD/*name_SPF*
6. /_N_CUS_DIR/*name_SPF*
7. /_N_CMA_DIR/*name_SPF*
8. /_N_CST_DIR/*name_SPF*

Randvillkor

- `CALLPATH` kontrollerar om det programmerade sökvägsnamnet verkligen finns. I händelse av fel avbryts detaljprogrambearbetningen med kompenseringsblock-larm 14009.
- `CALLPATH` kan också programmeras i INI-filer. Det verkar då för bearbetningstiden för INI-filen (WPD-INI-fil eller initieringsprogram för NC-aktiva data, t.ex. frames i 1:a kanalen `_N_CH1_UFR_INI`). Därefter återställs sökvägen igen.

2.24.3.10 Genomarbete externt underprogram (840D sl) (EXTCALL)

Med kommandot `EXTCALL` kan ett detaljprogram laddas över från ett externt minne och genomarbetas.

Som externa minnen står till förfogande:

- Lokal enhet
- Nätverksenhet
- USB-enhet

Märk

Som gränssnitt för genomarbetning av ett externt program som befinner sig på en USB-enhet får endast USB-gränssnitten på manöverpanelfronten resp. TCU användas.

OBSERVERA

Verktögs-/arbetsstycksskada genom USB-FlashDrive

Det rekommenderas att inte använda en USB-FlashDrive för genomarbetning av ett externt underprogram. Ett kommunikationsavbrott till USB-FlashDrive under genomarbetningen av detaljprogrammet genom kontaktsvårigheter, att den faller ur, avbrott för att den stöts till eller dras ur av misstag leder till omedelbart stopp av bearbetningen. Verktyg och/eller arbetsstycke kan därvid skadas.

Förinställning av den externa programsökvägen

Sökvägen till den externa programkatalogen kan förinställas med settingdatum:

SD42700 \$SC_EXT_PROG_PATH

Tillsammans med den vid `EXTCALL`-anropet angivna programvägen och -beteckningen resulterar därur den totala sökvägen för detaljprogrammet som skall anropas.

Märk

Parametrar

Vid anrop av ett externt program kan inga parametrar överföras till detta.

Syntax

```
EXTCALL("<Pfad/><Programmname>")
```

Betydelse

EXTCALL:	Kommando för anrop av ett externt underprogram	
"<Pfad/><Programmname>":	Konstant/variabel av typ STRING	
	<Pfad/>:	Absolut eller relativ sökvägsuppgift (optional)
	<Programmname>:	Programnamnet anges utan prefix "_N_". Filändelsen ("MPF", "SPF") kan med tecknet "_" eller "." läggas till programnamnet (optional). Exempel: "WELLE" "WELLE_SPF" "WELLE.SPF"

Sökvägsuppgift: Korta beteckningar

Följande korta beteckningar kan användas vid angivandet av sökvägen:

- Lokal enhet: "LOCAL_DRIVE:"
- CF-kort: "CF_CARD:"
- USB-enhet (manöverpanelfront): "USB:"

De korta beteckningarna "CF_CARD:" och "LOCAL_DRIVE:" kan användas alternativt.

Exempel

Genomarbetning från lokal enhet

Huvudprogrammet "MAIN.MPF" befinner sig i NC-minnet och har valts för genomarbetning.

Underprogram "SP_1"

Det externa underprogrammet "SP_1.SPF" resp. "SP_1.MPF" befinner sig på den lokala enheten i katalogen "/user/sinumerik/data/prog/WKS.DIR/WST1.WPD".

Sökvägen till den externa programkatalogen ska ställas in med:

SD42700 \$SC_EXT_PROG_PATH = LOCAL_DRIVE:WKS.DIR/WST1.WPD

Märk

Sökvägsuppgift för anropet av det externa underprogrammet:

- Utan användning av förinställningen: "LOCAL_DRIVE:WKS.DIR/WST1.WPD/SP_1"
- Med användning av förinställningen: "SP_1"

Underprogram "SP_2"

Det externa underprogrammet "SP_2.SPF" resp. "SP_2.MPF" befinner sig i katalogen WKS.DIR/WST1.WPD till USB-enheten. Förinställningen av sökvägen till den externa programkatalogen används för sökvägen till underprogrammet "SP_1" och skrivs heller inte om i huvudprogrammet. Därför måste vid anropet av underprogrammet "SP_2" den fullständiga sökvägen anges.

Huvudprogram "MAIN"

Programkod
N010 PROC MAIN
N020 ...
N030 EXTCALL("SP_1")
N030 EXTCALL("USB:WKS.DIR/WST1.WPD/SP_2")
N050 ...
N060 M30

Ytterligare informationer

EXTCALL-anrop med absolut sökvägsuppgift

Existerar underprogrammet under den angivna sökvägen, utförs det med EXTCALL-anropet. Existerar underprogrammet inte under den angivna sökvägen, avbryts utförandet av programmet med EXTCALL-anropet.

EXTCALL-anrop med relativ sökvägsuppgift / utan sökvägsuppgift

Vid ett EXTCALL-anrop med relativ sökvägsuppgift resp. utan sökvägsuppgift genomförs det förefintliga programminnet enligt följande mönster:

1. Är en sökvägsuppgift förinställd i SD42700 \$SC_EXT_PROG_PATH söks först utgående från denna sökväg efter uppgiften i EXTCALL-anropet (programnamn ev. med relativ sökvägsuppgift). Den absoluta sökvägen resulterar sedan genom sammanställning av tecken från :
 - Förinställd sökvägsuppgift i SD42700 \$SC_EXT_PROG_PATH
 - Skiljetecken "/"
 - Sökvägsuppgift och underprogramnamn i kommandot EXTCALL
2. Hittades underprogrammet inte under 1. genomförs katalogerna i användarminnet.

Sökningen slutar när underprogrammet har hittats första gången. Hittas inte underprogrammet avbryts utförandet av programmet med `EXTCALL`-anropet.

Inställbart omladdningsminne (FIFO-buffert)

För genomarbetningen av ett externt underprogram behövs ett omladdningsminne. Storleken på omladdningsminnet är förinställd med 30 kByte och kan endast förändras av maskintillverkaren (via MD18360 `MM_EXT_PROG_BUFFER_SIZE`).

Märk

Underprogram med hoppanvisningar

Vid externa underprogram, som innehåller hoppanvisningar (`GOTO`, `GOTOB`, `CASE`, `FOR`, `LOOP`, `WHILE`, `REPEAT`, `IF`, `ELSE`, `ENDIF` etc.), måste hoppmålen ligga inom omladdningsminnet.

Märk

ShopMill-/ShopTurn-program

ShopMill- och ShopTurn-program måste på grund av de i filslutet tillfogade konturbeskrivningarna vara fullständigt deponerade i omladdningsminnet.

För parallellt genomarbetade externa underprogram behövs alltid ett eget omladdningsminne.

Reset / Programslut / POWER ON

Genom Reset och POWER ON avbryts externa underprogramanrop och respektive omladdningsminnen raderas.

Ett för "Genomarbetning från Extern" valt underprogram förblir via Reset, Programslut eller POWER ON fortfarande valt för "Genomarbetning från Extern". Beteendet skiljer sig inte från internt valda program, såvida det externa programminnet fortfarande står till förfogande.

Litteratur

Ytterligare informationer över "Genomarbetning från Extern" finns i:

Funktionshandbok grundfunktioner; BAG, kanal, programdrift, Reset-beteende (K1)

2.24.3.11 Genomarbete externt underprogram (828D) (EXTCALL)

Med kommandot `EXTCALL` kan ett detaljprogram laddas över från ett externt minne och genomarbetas.

Som externa minnen står till förfogande:

- Användar CF-kort
- Nätverksenhet
- USB-enhet

Märk

Som gränssnitt för genomarbetning av ett på en USB-enhet förefintligt externt program får endast USB-gränssnittet till manöverpanelfronten (PPU) användas.

OBSERVERA

Verktögs-/arbetsstycksskada genom USB-FlashDrive

Det rekommenderas att inte använda en USB-FlashDrive för genomarbetning av ett externt underprogram. Ett kommunikationsavbrott till USB-FlashDrive under genomarbetningen av detaljprogrammet genom kontaktsvårigheter, att den faller ur, avbrott för att den stöts till eller dras ur av misstag leder till omedelbart stopp av bearbetningen. Verktyg och/eller arbetsstycke kan därvid skadas.

Förinställning av den externa programsökvägen

Sökvägen till den externa programkatalogen kan förinställas med settingdatum:

SD42700 \$SC_EXT_PROG_PATH

Tillsammans med den vid `EXTCALL`-anropet angivna programvägen och -beteckningen resulterar därur den totala sökvägen för detaljprogrammet som skall anropas.

Märk

Parametrar

Vid anrop av ett externt program kan inga parametrar överföras till detta.

Syntax

```
EXTCALL("<Pfad/><Programmname>")
```

Betydelse

EXTCALL:	Kommando för anrop av ett externt underprogram	
"<Pfad/><Programmname>":	Konstant / variabel av typ STRING	
	<Pfad/>:	Absolut eller relativ sökvägsuppgift (optional)
	<Programmname>:	Programnamnet anges utan prefix "_N". Filändelsen ("MPF", "SPF") kan med tecknet "_" eller "." läggas till programnamnet (optional). Exempel: "WELLE" "WELLE_SPF" "WELLE.SPF"

Sökvägsuppgift: Korta beteckningar

Följande korta beteckningar kan användas vid angivandet av sökvägen:

- Användar CF-kort: **"CF_CARD:"**
- USB-enhet (manöverpanelfront): **"USB:"**

Exempel

Huvudprogrammet "MAIN.MPF" befinner sig i NC-minnet och har valts för genomarbetning.

Underprogram "SP_1"

Det externa underprogrammet "SP_1.SPF" resp. "SP_1.MPF" befinner sig på användar CF-kortet i pärmen "/WKS.DIR/WST1.WPD".

Sökvägen till den externa programkatalogen ska ställas in med:

SD42700 \$SC_EXT_PROG_PATH = CF_CARD:WKS.DIR/WST1.WPD

Märk

Sökvägsuppgift för anropet av det externa underprogrammet:

- Utan användning av förinställningen: **"CF_CARD:WKS.DIR/WST1.WPD/SP_1"**
- Med användning av förinställningen: **"SP_1"**

Underprogram "SP_2"

Det externa underprogrammet "SP_2.SPF" resp. "SP_2.MPF" befinner sig i katalogen WKS.DIR/WST1.WPD till USB-enheten. Förinställningen av sökvägen till den externa programkatalogen används för sökvägen till underprogrammet "SP_1" och skrivs heller inte om i huvudprogrammet. Därför måste vid anropet av underprogrammet "SP_2" den fullständiga sökvägen anges.

Huvudprogram "MAIN"

Programkod
N010 PROC MAIN
N020 ...
N030 EXTCALL("SP_1")
N030 EXTCALL("USB:WKS.DIR/WST1.WPD/SP_2")
N050 ...
N060 M30

Ytterligare informationer**EXTCALL-anrop med absolut sökväguppgift**

Existerar underprogrammet under den angivna sökvägen, utförs det med EXTCALL-anropet. Existerar underprogrammet inte under den angivna sökvägen, avbryts utförandet av programmet med EXTCALL-anropet.

EXTCALL-anrop med relativ sökväguppgift / utan sökväguppgift

Vid ett EXTCALL-anrop med relativ sökväguppgift resp. utan sökväguppgift genomförs det förefintliga programminnet enligt följande mönster:

1. Är en sökväguppgift förinställd i SD42700 \$SC_EXT_PROG_PATH söks först utgående från denna sökväg efter uppgiften i EXTCALL-anropet (programnamn ev. med relativ sökväguppgift). Den absoluta sökvägen resulterar sedan genom sammanställning av tecken från :
 - Förinställd sökväguppgift i SD42700 \$SC_EXT_PROG_PATH
 - Skiljetecken "/"
 - Sökväguppgift och underprogramnamn i kommandot EXTCALL
2. Hittades underprogrammet inte under 1. genomförs katalogerna i användarminnet.

Sökningen slutar när underprogrammet har hittats första gången. Hittas inte underprogrammet avbryts utförandet av programmet med EXTCALL-anropet.

Inställbart omladdningsminne (FIFO-buffert)

För genomarbetningen av ett externt underprogram behövs ett omladdningsminne. Storleken på omladdningsminnet är förinställd (se MD18360 MM_EXT_PROG_BUFFER_SIZE).

Märk**Underprogram med hoppanvisningar**

Vid externa underprogram, som innehåller hoppanvisningar (GOTO, GOTOB, CASE, FOR, LOOP, WHILE, REPEAT, IF, ELSE, ENDIF etc.), måste hoppmålen ligga inom omladdningsminnet.

Märk

ShopMill-/ShopTurn-program

ShopMill- och ShopTurn-program måste på grund av de i filslutet tillfogade konturbeskrivningarna vara fullständigt deponerade i omladdningsminnet.

För parallellt genomarbetade externa underprogram behövs alltid ett eget omladdningsminne.

Reset / Programslut / POWER ON

Genom Reset och POWER ON avbryts externa underprogramanrop och respektive omladdningsminne raderas.

Ett för "Genomarbetning från Extern" valt underprogram förblir via Reset, Programslut eller POWER ON fortfarande valt för "Genomarbetning från Extern". Beteendet skiljer sig inte från internt valda program, såvida det externa programminnet fortfarande står till förfogande.

Litteratur

Ytterligare informationer över "Genomarbetning från Extern" finns i:

Funktionshandbok grundfunktioner; BAG, kanal, programdrift, Reset-beteende (K1)

2.25 Makroteknik (DEFINE ... AS)

OBSERVERA

Makroteknik förhöjer komplexiteten i programmeringen

Med makroteknik kan styrningens programmeringsspråk förändras kraftigt. Makroteknik får endast användas med yttersta omsorg.

Som makro betecknar man sammanfattningen av enskilda anvisningar till en hel anvisning med eget namn. Vid anrop av makros under programkörningen genomarbetas de under makronamnet programmerade anvisningarna efter varandra.

I enlighet med giltighetsområdet dvs. det område i vilket makrodefinitionen är verksam, finns det följande kategorier av makros:

- **Lokala makros**
Lokala makros är makros som är definierade i början till ett NC-program, som vid tidpunkten för genomarbetningen inte är huvudprogram. De läggs till vid anrop av NC-programmet och raderas med programslut-reset resp. den nästa styrningsstarten. Till lokala makros finns åtkomst endast inom det NC-program i vilket de är definierade.
- **Programglobala makros**
Programglobala makros är makros, som är definierade i början av ett som huvudprogram använt NC-program. De läggs till vid anrop av NC-programmet och raderas med programslut-reset resp. den nästa styrningsstarten. Till programglobala makros finns åtkomst i huvudprogrammet och i alla underprogram.

Märk

Disponibilitet för programglobala makros

I huvudprogrammet definierade programglobala makros är också disponibla i underprogrammen endast när följande maskindatum är inställt:

MD11120 \$MN_LUD_EXTENDED_SCOPE = 1

Med MD11120 = 0 är de i huvudprogrammet definierade programglobala makros endast disponibla i huvudprogrammet.

- **Globala makros**
Globala makros är NC- resp. kanal-globala makros, som är definierade i en definitionsfil (makrofil) och bibehålls också efter programslut-reset resp. den nästa styrningsstarten. Globala makros kan anropas och genomarbetas i varje valfritt huvud- eller underprogram.

Märk

För att kunna använda makros från en **extern** makrofil i NC-programmet, måste makrofilen laddas i NC.

Makros måste vara definierade före sin användning. Följande regler ska därvid iakttas:

- I makron kan valfria beteckningar, G-, M-, H-funktioner och L-underprogramnamn definieras.
- Makrodefinitionen kan ske i programbörjan eller i en egen definitionsfil (makrofil).
- Lokala och programglobala makros definieras i början på programmet.

- Globala makros måste definieras i en makrofil, t. ex. _N_DEF_DIR/_N_UMAC_DEF.
- G-kommando-makros kan endast definieras som globala makros.
- H- och L-funktioner kan programmeras med 2 platser.
- M- och G-kommandon kan programmeras med 3 platser.

Märk

Nyckelord och reserverade namn får inte skrivas över med makros.

Syntax

Makro-definition:

```
DEFINE <Macro_Name> AS <Operation_1> <Operation_2> ...
```

Anrop i NC-programmet:

```
<Macro_Name>
```

Betydelse

DEFINE ... AS:	Nyckelord-kombination för definition av en makro
<Macro_Name>:	Namn på makron Som makronamn är endast beteckningar tillåtna. Makron anropas utifrån NC-programmet med makronamnet.
<Operation_1>:	Första programmeringsanvisning i makron
<Operation_2>:	Andra programmeringsanvisning i makron

Randvillkor

Kapslingsdjup

En kapsling av makros är inte möjlig.

Exempel

Exempel 1: Makrodefinition i programmets början

Programkod	Kommentar
DEFINE LINIE AS G1 G94 F300	; Makro-definition
...	
N70 LINIE X10 Y20	; Makro-anrop
...	

Exempel 2: Makrodefinitioner i en makrofil

Programkod	Kommentar
DEFINE M6 AS L6	; Vid verktygsväxlingen anropas ett underprogram som övertar den nödvändiga dataöverföringen. I underprogrammet matas den egentliga verktygsväxlings-M-funktionen ut (t.ex. M106).
DEFINE G81 AS DRILL(81)	; Efterbildning av DIN-G-kommandot.
DEFINE G33 AS M333 G333	; Vid gängskärning krävs synkronisering med PLC. De ursprungliga G-kommandot G33 bytte namn till G333 per MD, programmeringen förblir lika för användaren.

Exempel 3: Extern makrofil

Efter inläsningen av den externa makrofilen i styrningen måste makrofilen laddas i NC. Först då kan makros användas i NC-programmet.

Programkod	Kommentar
%_N_UMAC_DEF	
;\$PATH=/_N_DEF_DIR	; Kundenspecifika makros
DEFINE PI AS 3.14	
DEFINE TC1 AS M3 S1000	
DEFINE M13 AS M3 M7	; Spindel höger, skärvätska till
DEFINE M14 AS M4 M7	; Spindel vänster, skärvätska till
DEFINE M15 AS M5 M9	; Spindel stopp, skärvätska från
DEFINE M6 AS L6	; Anrop av verktygsväxlingsprogrammet
DEFINE G80 AS MCALL	; Val borrcykel
M30	

Fil- och programförvaltning

3.1 Programminne

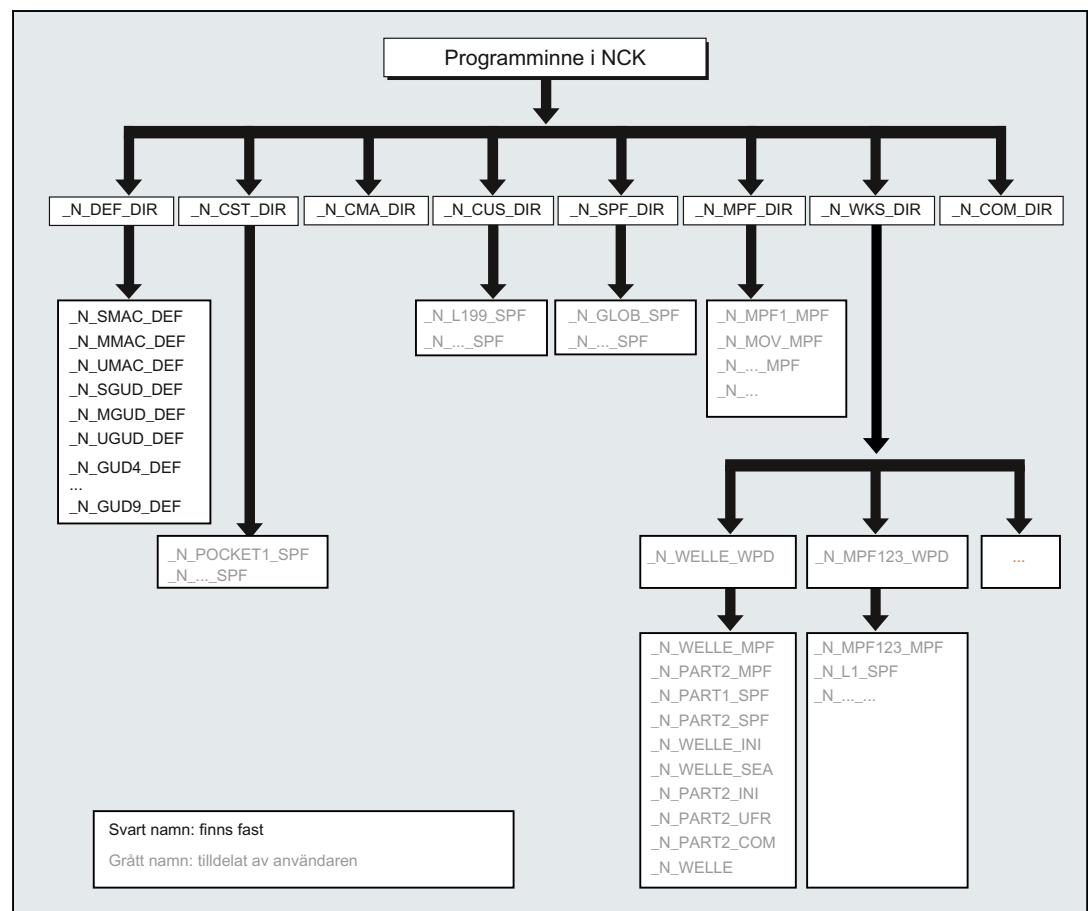
3.1.1 Programminne i NC

I programminnet sparas filer och program (t.ex. huvud- och underprogram, makro-definitioner) varaktigt (→ passivt filsystem).

Litteratur:

Funktionshandbok Tilläggsfunktioner; Minneskonfiguration (S7)

Dessutom finns det ett antal filtyper som kan sparas intermediärt här och vid behov (t.ex. vid bearbetning av ett visst arbetsstycke) överförs till arbetsminnet (t.ex. för initieringsändamål).



Standard-kataloger

Följande kataloger finns standardmässigt:

Katalog	Innehåll
_N_DEF_DIR	Datakomponenter och makrokomponenter
_N_CST_DIR	Standard-cykler
_N_CMA_DIR	Tillverkar-cykler
_N_CUS_DIR	Användar-cykler
_N_WKS_DIR	Arbetsstycken
_N_SPF_DIR	Globala underprogram
_N_MPF_DIR	Huvudprogram
_N_COM_DIR	Kommentarer

Filtyper

I programminnet kan följande filtyper införas:

Filtyp	Beskrivning
<Name>_MPF	Huvudprogram
<Name>_SPF	Underprogram
<Name>_TEA	Maskindata
<Name>_SEA	Settingdata
<Name>_TOA	Verktygskompenseringar
<Name>_UFR	Nollpunktsförflyttningar/frame
<Name>_INI	Initieringsfil
<Name>_GUD	Globala användardata
<Name>_RPA	R-parametrar
<Name>_COM	Kommentar
<Name>_DEF	Definitioner för globala användardata och makros

Arbetsstycke-huvudkatalog (_N_WKS_DIR)

Arbetsstycke-huvudkatalogen är standardmässigt installerad under beteckningen _N_WKS_DIR i programminnet. Arbetsstycke-huvudkatalogen innehåller för alla arbetsstycken som du programmerat de tillhörande arbetsstyckskatalogerna.

Arbetsstyckskataloger (..._WPD)

En arbetsstyckskatalog innehåller alla filer som är nödvändiga för bearbetningen av ett arbetsstycke. Dessa kan vara huvudprogram, underprogram, valfria initieringsprogram och kommentar-filer.

Initieringsprogram utförs efter programvalet en gång med den första detaljprogramstarten (enligt maskindatum MD11280 \$MN_WPD_INI_MODE).

Exempel:

Arbetsstycks katalogen _N_WELLE_WPD, som upprättades för arbetsstycket WELLE, innehåller följande filer:

Fil	Beskrivning
_N_WELLE_MPF	Huvudprogram
_N_PART2_MPF	Huvudprogram
_N_PART1_SPF	Underprogram
_N_PART2_SPF	Underprogram
_N_WELLE_INI	Allmänt initieringsprogram för arbetsstyckets data
_N_WELLE_SEA	Initieringsprogram settingdata
_N_PART2_INI	Allmänt initieringsprogram för data för program PART 2
_N_PART2_UFR	Initieringsprogram för frame-data för program PART 2
_N_WELLE_COM	Kommentarfil

Dessutom kan i en arbetsstycks katalog även data lagras, som inte omedelbart behövs för bearbetningen av NC. Dessa kan förutom ASCII-filer även vara binärfiler som t.ex bilder i JPG-format eller beskrivningar i PDF-format. För att dessa ska kunna interpretas av NC som binärfiler, måste filtilläggen vara kända i NC (inställning vid idrifttagningen via MD17000 \$MN_EXTENSIONS_OF_BIN_FILES; i grundinställningen är följande filtillägg förbelagda: JPG, GIF, PNG, BMP, PDF, ICO, HTM).

Välja arbetsstycke för bearbetning

En arbetsstycks katalog kan väljas för genomarbetningen i en kanal. Befinner sig i denna katalog ett huvudprogram med **samma namn** eller endast ett enda huvudprogram (_MPF), så väljs detta automatiskt för denna genomarbetning.

Litteratur:

Användarhandbok

3.1.2 Externt programminne

Förutom det passiva filsystemet i NC kan även externa programminnen vara tillgängliga på en maskin (t.ex på den lokala enheten eller på en nätverksenhet).

Via funktionerna "Genomarbetning från Extern" eller "EES (Execution from External Storage)" kan detaljprogram genomarbetas **direkt** från externa programminnen.

Litteratur:

Funktionshandbok Grundfunktioner; K1: BAG, kanal, programdrift, Reset-beteende

Globalt detaljprogramminne (GDIR)

Vid överenskommelsen för enheterna kan en av enheterna utnämnas till globalt detaljprogramminne (GDIR).

Litteratur:

Användarhandbok; Kapitel: "Förvalta program" > "Ställa in enheter"

3.1 Programminne

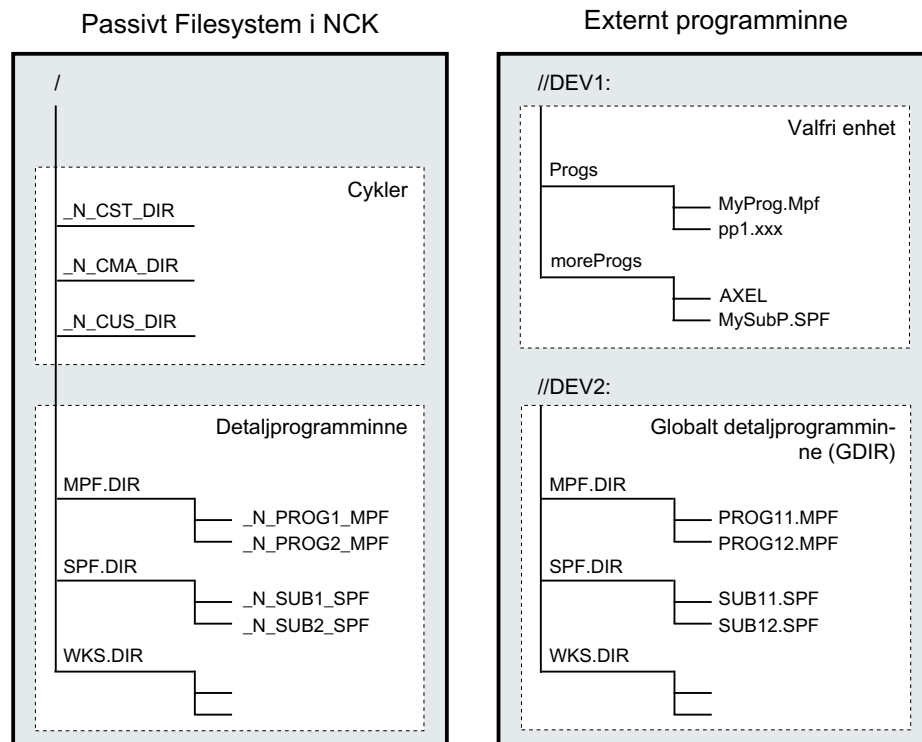
Av systemet anläggs sedan automatiskt katalogerna MPF.DIR, SPF.DIR och WKS.DIR på enheten. Dessa tre kataloger bildar GDIR.

GDIR spelar roll endast för funktionen EES. Beroende på enhetskonfiguration ersätter eller utvidgar GDIR NC-detaljprogramminnet. Inställningen av en GDIR är för EES-driften dock inte absolut nödvändigt.

Katalogerna och filerna i GDIR kan i detaljprogrammet adresseras på samma sätt som i det passiva Filesystemet. Därmed är en kompatibel flyttning av ett NC-program till GDIR möjlig med sökvägsuppgifter från det passiva Filesystemet. Katalogen SPF.DIR till GDIR ingår i sökvägen för underprogram.

Programorganisation

Den följande bilden ska åskådliggöra programorganisationen på externa programminnen:



Case-insensitive filesystem

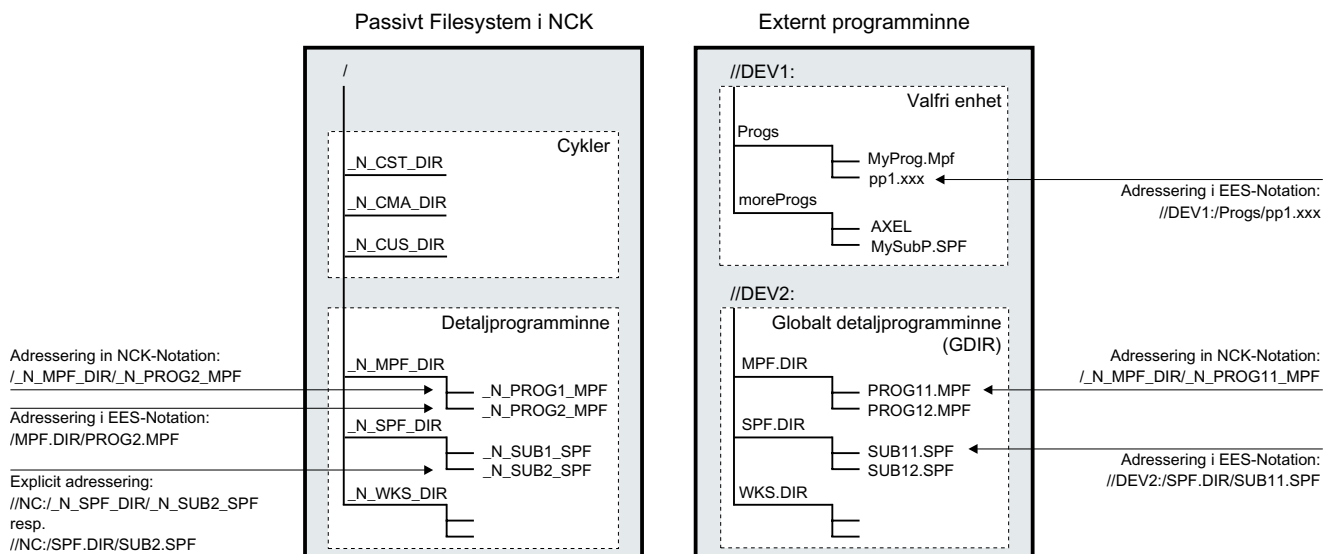
Märk

För att undvika problem genom skrivning av stora och små bokstäver vid fil-adresseringen (se "Adressering av filer i programminnet (Sida 213)") bör **case-insensitive** filesystem användas som externa programminnen.

3.1.3 Adressering av filer i programminnet

En fil i programminnet, som anropas genom ett filbehandlingskommando (t.ex. WRITE, DELETE, READ, ISFILE, FILEDATE, FILETIME, FILESIZE, FILESTAT, FILEINFO) refereras antingen genom en absolut sökväg plus filnamn eller endast genom filnamnet. I det andra fallet används sökvägen för det valda programmet som sökväg för filen.

Adressering i NC-/EES-Notation



Adressering av filer i det passiva Filesystemet

Adresseringen av filer i det passiva Filesystemet sker vanligen i **NC-Notation** (katalog- och filnamn börjar med domänidentifikation "`_N_`", skiljetecken för katalog-/filtillägget är "`_`") utan uppgift för enhetsnamnet. En adressering i **EES-Notation** (utan domänidentifikation "`_N_`", skiljetecken för katalog-/filtillägg är "`.`") är dock även tillåten.

Exempel:

- NC-Notation: `"/_N_SPF_DIR/_N_SUB1_SPF"`
- EES-Notation: `"/SPF.DIR/SUB1.SPF"`

Märk

Adressering av filer i det passiva Filesystemet i EES-Notation omvandlas internt enligt följande regler i NC-Notation:

- Katalog- och filnamn utvidgas med domänidentifikationen "`_N_`".
- Är det fjärde tecknet från slutet i katalog- resp. filnamnet en punkt ("`.`") omvandlas det till ett nedsänkt streck ("`_`").

Via det fördefinierade enhetsnamnet `"/NC:"` kan det passiva Filesystemet också adresseras målinriktat.

Exempel:

- NC-Notation: "//NC:/_N_SPF_DIR/_N_SUB1_SPF"
- EES-Notation: "//NC:/SPF.DIR/SUB1.SPF"

Adressering av filer i ett externt programminne

Adresseringen av filer i ett externt programminne, som inte är utnämnt till GDIR, måste ske i EES-Notation. I början av adresseringssökvägen måste enhetsnamnet (t.ex "//DEV1:") anges. Alla i /user/sinumerik/hmi/cfg/logdrive.ini projekterade symboliska instrumentnamn är tillåtna.

Exempel:

- EES-Notation: "//DEV1:/MyProgDir/pp1.xxx"
- NC-Notation: inte tillåtet

Adressering av filer i det globala detaljprogramminnet (GDIR)

Vid adresseringen av filer i GDIR är förutom sökvägsuppgift i EES-Notation även en sökvägsuppgift i NC-Notation tillåten.

Exempel:

- EES-Notation: "//DEV2:/MPF.DIR/PROG11.MPF"
- NC-Notation: "/_N_MPF_DIR/_N_PROG11_MPF"

Märk

Adressering av filer i GDIR i NC-Notation omvandlas internt enligt följande regler i EES-Notation:

- Domänidentifikationen "_N_" i katalog- och filnamnet tas bort.
 - Är det fjärde tecknet från slutet i katalog- resp. filnamnet ett nedsänkt streck ("_") omvandlas det till en punkt (".").
-

Regler för sökvägsuppgift

En fullständig sökvägsuppgift består av enhetsnamn, sökväg för katalogen och filnamn.

Enhetsnamn

För uppgiften för enhetsnamnet gäller följande regler:

- Alla i /user/sinumerik/hmi/cfg/logdrive.ini projekterade symboliska instrumentnamn är tillåtna.
- I början står tecknet "//", följt av minst en bokstav eller en siffra.
- De följande tecknen kan vara en valfri kombination av bokstäver, siffror, "_" och mellanslag.
- Namnet avslutas med en bokstav eller en siffra följt av ett ":".
- Ytterligare specialtecken är inte tillåtna.

Märk

För det passiva Filesystemet är enhetsnamnet "//NC:" fördefinierat.

Exempel:

- Externt programminne:
 - //Drive1:
 - //Drive_1:
 - //Drive 1:
 - //A B:
 - //1 B C 2:

Sökväg för katalog

För uppgiften för katalogens sökväg gäller följande regler:

- I början och i slutet av katalogens sökväg och som skiljetecken för de enskilda sökvägsdelarna står "/".
-

Märk

Ett dubbelt snedstreck ("/") inom katalogens sökväg är **inte** tillåtet!

- Katalognamn:
 - Katalognamn måste börja med en bokstav eller en siffra. Endast vid adresseringen i NC-Notation börjar katalognamnet med domänidentifikationen "_N_".
 - De följande tecknen kan vara en valfri kombination av bokstäver, siffror och ("_").
-

Märk

Vid externa programminnen är också mellanslag tillåtna i katalognamnet. Detta gäller dock inte, när det externa programminnet är inställt som globalt detaljprogramminne (GDIR).

- Ytterligare specialtecken är inte tillåtna.
 - Katalogtillägg:
 - Katalogtillägg måste bestå av exakt tre bokstäver/siffror.
 - De skiljs åt med "_" (NC-Notation) resp. "." (EES-Notation) från katalognamnet.
-

Märk

I det passiva Filesystemet finns det endast katalogtilläggen _DIR och _WPD.

Exempel:

- Passivt Filesystem resp. GDIR:
 - NC-Notation: _N_WKS_DIR/_N_MYNCPROGS_WPD/...
 - EES-Notation: WKS.DIR/MYPROGS.WPD/...
- Externt programminne:
 - /abc
 - /ab_c.def
 - /ab c1.def
 - /a b c .d11
 - /abc.def/ghi.klm

Filnamn

För filnamn gäller följande regler:

- Endast vid adresseringen i NC-Notation börjar filnamnen med domänidentifikationen "_N_".
- De båda följande tecknen bör vara två bokstäver eller ett nedsänkt streck följt av en bokstav.

Märk

När detta villkor är uppfyllt då kan ett NC-program anropas endast genom angivande av programnamnet som underprogram utifrån ett annat program. Börjar programnamnet däremot med siffror då är underprogramanrop endast möjligt via CALL-anvisningen.

- De följande tecknen kan vara en valfri kombination av bokstäver, siffror och ("_").
- Filtillägg:
 - Filtillägget måste bestå av exakt tre bokstäver/siffror.

Märk

Tillåtna filtillägg i det passiva filsystemet se "Programminne i NC (Sida 209)".

- Det skiljs åt med "_" (NC-Notation) resp. "." (EES-Notation) från filnamnet.

Exempel:

- Passivt Filesystem resp. GDIR:
 - NC-Notation: _N_SUB1_SPF
 - EES-Notation: SUB1.SPF
- Externt programminne:
 - Teil1
 - _Teil1
 - Teil_1.spf
 - Teil1.mpf

DIN underprogramnamn

För DIN underprogramnamn gäller följande regler:

- Det första tecknet är bokstaven "L".
- De följande tecknen är siffror (minst en).
- Filtillägg:
 - Filtillägget måste bestå av exakt tre bokstäver.
 - Det skiljs åt med "_" (NC-Notation) resp. "." (EES-Notation) från filnamnet.

Exempel:

- L123
- L1_SPF (NC-Notation) resp. L1.SPF (EES-Notation)

Maximal sökväglängd

För uppgiften för enhetsnamnet och katalogens sökväg står maximalt 128 bytes till förfogande, längden på filnamnet får maximalt uppgå till 31 bytes. För hela sökvägen resulterar det i en maximal längd på 159 bytes.

3.1.4 Sökväg vid underprogramanrop

Vid underprogramanrop utan sökvägsuppgift fastställs den absoluta sökvägen genom att arbeta igenom en fast sökväg.

Programminnena genomsöks därvid i följande ordningsföljd:

	Katalog	Beskrivning
1	aktuell katalog / <i>name</i>	Den aktuella katalogen är den katalog i vilken programvalet har gjorts. Denna kan vara: • en arbetsstyckskatalog eller standardkatalogen <code>_N_MPF_DIR</code> i NC-detaljprogramminnet resp. globala detaljprogramminnet eller • en valfri katalog i ett externt programminne
2	aktuell katalog / <i>name_SPF</i>	
3	aktuell katalog / <i>name_MPF</i>	
4	a //NC:/_N_SPF_DIR / <i>name_SPF</i>	Underprogramkatalog i NC-detaljprogramminnet
	b //DEV2:/_N_SPF_DIR / <i>name_SPF</i> ¹⁾	Underprogramkatalog i det globala detaljprogramminnet Observera: Detta sökningsmoment utgår, när inget globalt detaljprogramminne är installerat eller programvalet har gjorts i NC-detaljprogramminnet.
5	Med <code>CALLPATH</code> programmerat sökvägstillsätt (se "Utvidga sökvägen vid underprogramanrop (<code>CALLPATH</code>)" (Sida 197)). Observera: Detta sökningsmoment utgår, när ingen <code>CALLPATH</code> har programmerats.	
6	/_N_CUS_DIR / <i>name_SPF</i>	Användarcyklerekatalog

	Katalog	Beskrivning
7	<i>/_N_CMA_DIR / name_SPF</i>	Tillverkarcyklerkatalog
8	<i>/_N_CST_DIR / name_SPF</i>	Standardcyklerkatalog

¹⁾ //DEV2:" står exempelvis för den enhet, på vilken det globala detaljprogramminnet är installerat.

För sökningen gäller följande regler:

- Sökvägen körs igenom för varje enskilt underprogramanrop, dvs. det är irrelevant, var det överordnade programmet befinner sig.
- Beroende på katalog tas det hänsyn till olika filtyper.
- Principiellt söks i en katalog inte i underlagrade, dvs. kapslade kataloger.

3.1.5 Avfrågning av sökväg och filnamn

För avfrågningen av sökvägen och filnamnet för ett NC-program står följande i detaljprogrammet läsbara systemvariabler till förfogande:

Systemvariabel	Typ	Betydelse
\$P_STACK	INT	Levererar den programnivå i vilken det aktuella NC-programmet bearbetas.
\$P_PATH[<n>]	STRING	Levererar sökvägen för det NC-program, som bearbetas i den genom fältindex <n> valda programnivån. Exempel: \$P_PATH[0] levererar sökvägen för huvudprogrammet, t.ex. <i>"/_N_WKS_DIR/_N_WELLE_WPD/"</i> . \$P_PATH[\$P_STACK - 1] levererar sökvägen för det program som anropar. Hänför sig sökvägen till ett NC-program, som är lagrat i det passiva Filesystemet till NC eller i det globala detaljprogramminnet (GDIR), levereras sökvägen i NC-Notation. Hänför sig sökvägen till ett NC-program, som genomarbetas av ett annat externt programminne än det globala detaljprogramminnet, levererar \$P_PATH sökvägen i EES-Notation.
\$P_PROG[<n>]	STRING	Levererar namnet för det NC-program, som bearbetas i den av fältindex <n> valda programnivån. Är NC-programmet lagrat i det passiva Filesystemet till NC eller i det globala detaljprogramminnet (GDIR), levereras programnamnet i NC-Notation. Genomarbetas NC-programmet av en annan extern enhet än det globala detaljprogramminnet, levererar \$P_PROG namnet i EES-Notation.
\$P_PROGPATH	STRING	Levererar sökvägen för det NC-program, som just genomarbetas. Anropet av \$P_PROGPATH är identiskt med \$P_PATH[\$P_STACK].

Systemvariabel	Typ	Betydelse
\$P_IS_EES_PATH[<n>]	BOOL	Avfrågning om den av \$P_PATH[<n>] levererade sökvägen resp. det av \$P_PROG[<n>] levererade programnamnet motsvarar NC-Notation eller EES-Notation.
		= FALSE \$P_PATH[<n>] och \$P_PROG[<n>] levererar NC-Notation. Dvs. i varje beteckning står först ett prefix "_N_". Skiljetecknet för filidentifikationen är "_". Exempel: • Sökväg i NC-Notation: "/_N_WKS_DIR/_N_MYWPD_WPD/" • Programnamn i NC-Notation: "_N_MYPROG_MPF" En sökväg i NC-Notation kan hänföra sig både till det passiva Filesystemet i NC och även till det globala detaljprogramminnet.
		= TRUE \$P_PATH[<n>] och \$P_PROG[<n>] levererar EES-Notation. Dvs. före beteckningen står inget prefix "_N_". Skiljetecknet för filidentifikationen är ".". Exempel: • Sökväg i EES-Notation: "//DEV1:/WKS.DIR/MYWPD.WPD/" • Programnamn i EES-Notation: "MYPROG.MPF"

<n>: Index <n> definierar den programnivå, från vilken sökvägsinformationen ska läsas (värdeområde: 0 ... 17)

Märk

I EES-drift levereras utanför det globala detaljprogramminnet (GDIR) sökvägsnamn i EES-Notation av systemvariablerna \$P_PROG, \$P_PATH och \$P_PROGPATH.

Användarprogram, som utvärderar och vidarebearbetar dessa sökvägsnamn, måste därför utvidgas för EES-driften så att de också kan bearbeta sökvägsnamn i EES-Notation.

3.2 Arbetsminne (CHANDATA, COMPLETE, INITIAL)

Funktion

Arbetsminnet innehåller de aktuella system- och användardata, med vilka styrningen drivs (aktivt filsystem), t.ex.:

- Aktiva maskindata
- Verktygskompenseringsdata
- Nollpunktsförflyttningar
- ...

Initieringsprogram

Härvid handlar det om program med vilka arbetsminnets data förbeläggs (initieras). För detta kan följande filtyper användas:

Filtyp	Beskrivning
name_TEA	Maskindata
name_SEA	Settingdata
name_TOA	Verktgskompenseringar
name_UFR	Nollpunktsförflyttningar/frame
name_INI	Initieringsfil
name_GUD	Globala användardata
name_RPA	R-parametrar

Dataområden

Data kan delas in i olika områden inom vilka de ska gälla. Till exempel kan en styrning förfoga över flera kanaler eller vanligen också över flera axlar.

Det finns:

Identifikation	Dataområden
NC	NC-specifika data
CH<n>	Kanalspecifika data (<n> anger kanalnumret)
AX<n>	Axelspecifika data (<n> anger numret på maskinaxeln)
TO	Verktgysdata
COMPLETE	Alla data

Skapa initieringsprogram på extern PC

Med hjälp av dataområdesidentifikationen och datatypidentifikationen kan de områden bestämmas som betraktas som enhet vid datasäkringen:

_N_AX5_TEA_INI	Maskindata för axel 5
_N_CH2_UFR_INI	Frames för kanal 2
_N_COMPLETE_TEA_INI	Alla maskindata

Efter idrifttagning av styrningen finns ett datablock i arbetsminnet som garanterar en föreskriftsmässig drift av styrningen.

Tillvägagångssätt vid flerkanaliga styrningar (CHANDATA)

CHANDATA (<Kanalnummer>) för fler kanaler är tillåtet endast i filen _N_INITIAL_INI. Det är idrifttagningsfilen med vilken alla data i styrningen initieras.

Programkod	Kommentar
%_N_INITIAL_INI	
CHANDATA (1)	
	; Maskinaxeltillordning kanal 1:
\$MC_AXCONF_MACHAX_USED[0]=1	
\$MC_AXCONF_MACHAX_USED[1]=2	
\$MC_AXCONF_MACHAX_USED[2]=3	
CHANDATA (2)	
	; Maskinaxeltillordning kanal 2:
\$MC_AXCONF_MACHAX_USED[0]=4	
\$MC_AXCONF_MACHAX_USED[1]=5	
CHANDATA (1)	
	; Axiala maskindata:
	; Precisionsstoppfönster grovt:
\$MA_STOP_LIMIT_COARSE[AX1]=0.2	; Axel 1
\$MA_STOP_LIMIT_COARSE[AX2]=0.2	; Axel 2
	; Presisionsstoppfönster fint:
\$MA_STOP_LIMIT_FINE[AX1]=0.01	; Axel 1
\$MA_STOP_LIMIT_FINE[AX1]=0.01	; Axel 2

OBSERVERA

CHANDATA-anvisning

I detaljprogrammet får CHANDATA-anvisningen endast ställas in för den kanal på vilken NC-programmet genomarbetas. Dvs. anvisningen kan användas därtill att skydda NC-program från att genomarbetas på en kanal som inte är avsedd.

I händelse av fel avbryts programgenomarbetningen.

Märk

INI-filerna i jobblistor innehåller inga CHANDATA-anvisningar.

Spara initieringsprogram (COMPLETE, INITIAL)

Filerna i arbetsminnet kan sparas på en extern PC och åter läsas in därifrån.

- Filerna sparas med COMPLETE.
- Med INITIAL skapas en INI-fil (_N_INITIAL_INI) över alla områden.

Läsa in initieringsprogram

OBSERVERA
Dataförlust Läses filen med namnet "INITIAL_INI" in så initieras alla data som inte försörjs i filen med standarddata. Maskindata är undantagna därifrån. Alltså försörjs settingdata, verktygsdata, NPV, GUD-värden, ... med standarddata (normalt "NULL").

För inläsning av enskilda maskindata lämpar sig t.ex. filen COMPLETE_TEA_INI. I denna fil väntar styrningen endast maskindata. Därmed förblir de andra dataområdena oberörda i detta fall.

Ladda initieringsprogram

INI-programmen kan också väljas som detaljprogram och anropas när de användas data endast från en kanal. Så är det också möjligt att initieras programstyrda data.

Skyddsområden

4.1 Definiera skyddsområden (CROTDEF, NROTDEF)

Skyddsområden, som ska skydda maskinelementen för kollisioner, definieras i detaljprogrammet alltid i block. Dessa innehåller följande element:

1. Fastläggande av arbetsplanet
Före den egentliga skyddsområdesdefinitionen måste det arbetsplan väljas, till vilket konturbeskrivningen av skyddsområdet ska hänföra sig till.
2. Definitionsbörjan
Beroende på NC-kommando läggs antingen ett kanalspecifikt eller ett maskinspecifikt skyddsområde till.
3. Konturbeskrivning av skyddsområdet
Konturen för ett skyddsområde beskrivs med hjälp av förflyttningsrörelser. Dessa utförs inte och har ingen förbindelse till föregående eller efterföljande geometribeskrivningar. De definierar endast skyddsområdet.
4. Definitionsslut

Syntax

```
DEF INT <Var>
G17/G18/G19
CROTDEF/NROTDEF (<n>,<t>,<AppLim>,<AppPlus>,<AppMinus>)
G0/G1/... X/Y/Z...
...
EXECUTE (<Var>)
```

Betydelse

DEF INT <Var>:	Definition av en lokal hjälpvariabel av datatyp INTEGER	
<Var>:	Namn på hjälpvariabeln	
G17/G18/G19:	Arbetsplan Observera: Arbetsplanet får inte ändras före definitionsslut. En programmering av applikatan mellan definitionsbörjan och -slut är inte tillåtet.	
CROTDEF () :	Fördefinierad procedur för definition av ett kanals specifikt skyddsområde	
NROTDEF () :	Fördefinierad procedur för definition av ett maskins specifikt skyddsområde	
<n>:	Nummer på det definierade skyddsområdet	
	Datatyp:	INT

4.1 Definiera skyddsområden (CROTDEF, NPROTDEF)

<t>:	Typ för skyddsområdet	
	Datatyp:	BOOL
	Värde:	TRUE Verktysrelaterat skyddsområde
		FALSE Arbetsstycksrelaterat skyddsområde
<AppLim>:	Typ av begränsning i den 3:e dimensionen	
	Datatyp:	INT
	Värde:	0 ingen begränsning
		1 Begränsning i plus-riktning
		2 Begränsning i minus-riktning
		3 Begränsning i plus- och minus-riktning
<AppPlus>:	Värde för begränsningen i plus-riktning för 3:e dimensionen	
	Datatyp:	REAL
<AppMinus>:	Värde för begränsningen i minus-riktning för 3:e dimensionen	
	Datatyp:	REAL
G0/G1/... X/Y/Z... :	Konturen för ett skyddsområde beskrivs med maximalt 11 förflyttningsrörelser i det valda arbetsplanet. Därvid är den första förflyttningsvägen rörelsen till konturen. Den sista punkten i konturbeskrivningen måste alltid sammanfalla med den första punkten i konturbeskrivningen. Som skyddsområde gäller området till vänster om konturen. - Inre skyddsområde Konturen för ett inre skyddsområde ska beskrivas moturs. - Yttre skyddsområde (endast tillåtet för arbetsstycksrelaterade skyddsområden!) Konturen för ett yttre skyddsområde ska beskrivas medurs. Följande konturelement är tillåtna: - G0, G1 för raka konturelement - G2 för cirkelavsnitt medurs Endast tillåtet för arbetsstycksrelaterade skyddsområden! Inte tillåtet för verktygsrelaterade skyddsområden, eftersom de endast får vara konvexa. - G3 för cirkelavsnitt moturs Observera: Ett skyddsområde kan inte beskrivas med en sluten cirkel. En sluten cirkel måste delas upp i två delcirkel. Observera: Följden G2 → G3 resp. G3 → G2 är inte tillåten! Mellan de båda cirkelblocken måste ett kort G1-block infogas.	
EXECUTE(<Var>):	Fördefinierad procedur, som markerar slutet på definitionen Med EXECUTE kopplas tillbaka till den normala programbearbetningen.	

Exempel

Se exempel under "Aktivera/inaktivera skyddsområden (CROT, NPROT) (Sida 227)".

Ytterligare informationer

Maskinspecifika skyddsområden

Ett maskinspecifikt skyddsområde resp. dess kontur definieras med hjälp av geometriaxlarna, dvs. relaterade till baskoordinatsystemet (BKS) för en kanal. För att en korrekt skyddsområdesövervakning ska kunna äga rum i alla kanaler, i vilka det maskinspecifika skyddsområdet är aktivt, måste baskoordinatsystemet (BKS) vara identiskt för alla ifrågakvarande kanaler:

- Läge för koordinaternas ursprung relaterat till maskinens nollpunkt
- Orientering för koordinataxlarna

Referenspunkt för konturbeskrivningen

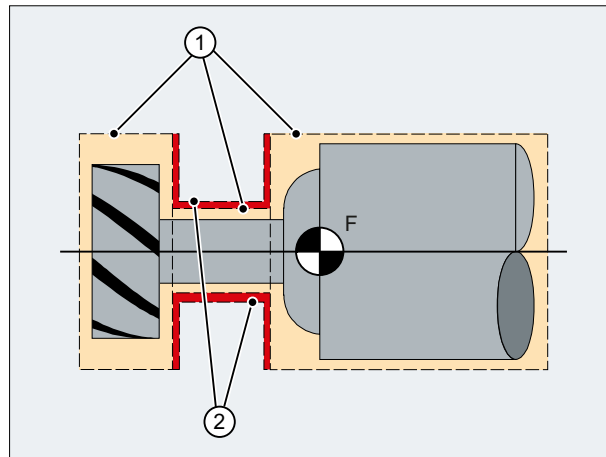
- Verktysrelaterade skyddsområden
Koordinater för **verktys**relaterade skyddsområden ska anges absolut, relaterat till **verktysbärarens referenspunkt F**.
- Arbetsstycksrelaterade skyddsområden
Koordinater för **verksstycks**relaterade skyddsområden ska anges absolut, relaterat till nollpunkten i **baskoordinatsystemet (BKS)**.

Rotationssymmetriska skyddsområden

Vid rotationssymmetriska skyddsområden (t.ex. spindelchuck) måste hela konturen beskrivas, inte endast konturen till rotationsmitten.

Verktysrelaterade skyddsområden

Verktysrelaterade skyddsområden måste alltid vara konvexa. Om ett konkavt skyddsområde önskas ska detta delas upp i flera konvexa skyddsområden.



- ① Konvexa skyddsområden
- ② Konkava skyddsområden (**inte tillåtet!**)
- F Verktysbärarreferenspunkt

Randvillkor

Under definitionen av ett skyddsområde får följande funktioner inte vara aktiva resp. användas:

- Verktygsradiekompensering (fräsradierekompensering, skärradierekompensering)
- Transformation
- Referenspunktskörning (G74)
- Fastpunktskörning (G75)
- Fördröjningstid (G4)
- Blockförkörnings-stopp (STOPRE)
- Programslut (M17, M30)
- M-funktioner: M0, M1, M2

4.2 Aktivera/inaktivera skyddsområden (CProt, NProt)

Tidigare i detaljprogrammet definierade skyddsområden kan alltid aktiveras resp. föraktiveras av PLC-användarprogrammet för en senare aktivering. Aktiva skyddsområden kan alltid åter inaktiveras.

Vid aktiveringen resp. inaktiveringen finns dessutom möjligheten, att förflytta skyddsområdets referenspunkt relativt.

Märk

Det tas hänsyn till ett skyddsområde först efter referenskörning av alla geometriaxlarna i den kanal i vilken det aktiverades.

Märk

Övervakning av skyddsområdena

Är inget verktygsrelaterat skyddsområde aktivt kontrolleras verktygsbanan mot de arbetsstycksrelaterade skyddsområdena.

Är inget arbetsstycksrelaterat skyddsområdet aktivt äger ingen skyddsområdesövervakning rum.

Syntax

CProt (<n>, <Status>, <XMov>, <YMov>, <ZMov>)

NProt (<n>, <Status>, <XMov>, <YMov>, <ZMov>)

Betydelse

CProt:	Fördefinierad procedur för aktivering av ett kanalspecifikt skyddsområde			
NProt:	Fördefinierad procedur för aktivering av ett maskinspecifikt skyddsområde			
<n>:	Nummer på skyddsområdet			
	Datotyp:	INT		
<Status>:	Med denna parameter sätts den kanalspecifika aktiveringsstatu- sen			
	Datotyp:	INT		
	Värde:	0	Inaktivera skyddsområde	
		1	Föraktivera skyddsområde	
		2	Aktivera skyddsområde	
		3	Föraktivera skyddsområden med villkorligt stopp	

<XMov>, <YMov>, <ZMov>:	Additiva förflyttningsvärden i X/Y/Z-riktning	
	Förflyttningen kan göras i 1, 2 eller 3 dimensioner. Förflyttningsvärdena hänför sig till: - maskinens nollpunkt vid arbetsstycksrelaterat skyddsområde - verktygsbärarens referenspunkt F vid verktygsrelaterat skyddsområde	
	Datotyp:	REAL

Exempel

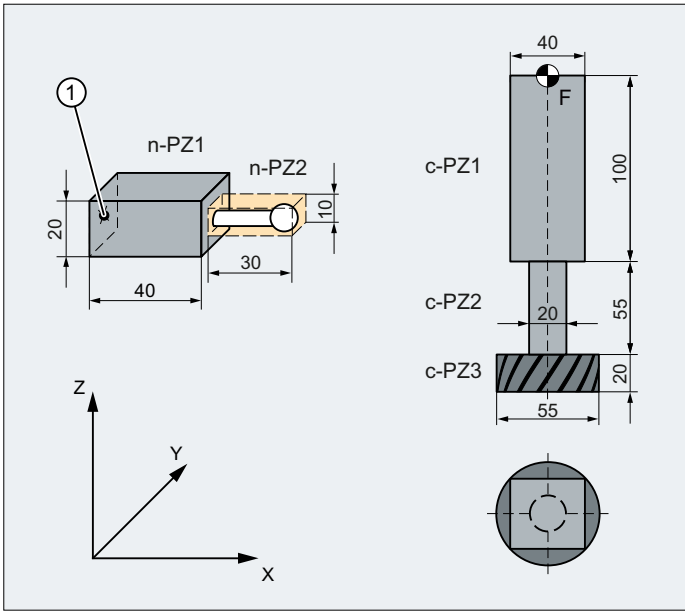
För en fräsmaskin ska en möjlig kollision av fräsen med mätproben övervakas. Läget för mätproben ska vid aktiveringen anges med en förflyttning.

Följande skyddsområden definieras för detta :

- Alltid ett maskinspecifikt och arbetsstycksrelaterat skyddsområde för mätprobehållaren (n-PZ1) och för mätproben själv (n-PZ2).
- Alltid ett kanalspecifikt och verktygsrelaterat skyddsområde för fräshållaren (c-PZ1), fräskäftet (c-PZ2) och för fräsen själv (c-PZ3).

Orienteringen för alla skyddsområden ligger i Z-riktningen.

Läget för mätprobens referenspunkt vid aktiveringen ska ligga vid X = -120, Y = 60 och Z = 80.



- ① Referenspunkt för skyddsområdet till mätproben
- F Verktygsbärarreferenspunkt

Programkod	Kommentar
DEF INT PROTZONE	; Definition av en hjälpvariabel
G17	; Arbetsplan XY
; Definition av skyddsområdena:	

4.2 Aktivera/inaktivera skyddsområden (CPROT, NPROT)

Programkod	Kommentar
NPROTDEF(1,FALSE,3,10,-10)	; Skyddsområde n-PZ1
G01 X0 Y-10	
X40	
Y10	
X0	
Y-10	
EXECUTE(PROTZONE)	
NPROTDEF(2,FALSE,3,5,-5)	; Skyddsområde n-PZ2
G01 X40 Y-5	
X70	
Y5	
X40	
Y-5	
EXECUTE(PROTZONE)	
CPROTDEF(1,TRUE,3,0,-100)	; Skyddsområde c-PZ1
G01 X-20 Y-20	
X20	
Y20	
X-20	
Y-20	
EXECUTE(PROTZONE)	
CPROTDEF(2,TRUE,3,-100,-150)	; Skyddsområde c-PZ2
G01 X0 Y-10	
G03 X0 Y10 J10	
X0 Y-10 J-10	
EXECUTE(PROTZONE)	
CPROTDEF(3,TRUE,3,-150,-170)	; Skyddsområde c-PZ3
G01 X0 Y-27.5	
G03 X0 Y27.5 J27.5	
X0 Y27.5 J-27.5	
EXECUTE(PROTZONE)	
; Aktivering av skyddsområden:	
NPROT(1,2,-120,60,80)	; Aktivera skyddsområde n-PZ1 med förflyttning
NPROT(2,2,-120,60,80)	; Aktivera skyddsområde n-PZ2 med förflyttning
CPROT(1,2,0,0,0)	; Aktivera skyddsområde c-PZ1
CPROT(2,2,0,0,0)	; Aktivera skyddsområde c-PZ2
CPROT(3,2,0,0,0)	; Aktivera skyddsområde c-PZ3

Ytterligare informationer

Aktiveringsstatus efter start av styrningen

Ett skyddsområde kan redan efter starten av styrningen och referenskörningen av axlarna vara aktivt. Detta är fallet när följande systemvariabel har satts på TRUE för skyddsområdet:

- \$SN_PA_ACTIV_IMMED[<n>] (för maskinspecifikt skyddsområde) resp.
- \$SC_PA_ACTIV_IMMED[<n>] (för kanalspecifikt skyddsområde)
Index "<n>" motsvarar numret för skyddsområdet: 0 = 1. Skyddsområde

Skyddsområdet aktiveras med status = 2 och utan förflyttning.

Flerfaldig aktivering av ett skyddsområde

Ett maskinspecifikt skyddsområde kan vara verksamt i flera kanaler samtidigt (t.ex. skyddsområde för pinolen med två mittemot liggande slider). Övervakningen av skyddsområdena görs endast när alla geometriaxlar är referenskörda.

Ett skyddsområde kan inte aktiveras samtidigt i en kanal med olika förflyttningar.

Skyddsområdesövervakning vid aktiv verktygsradiekompensering

Vid aktiv verktygsradiekompensering är en funktionsduglig skyddsområdesövervakning endast möjlig när planet för verktygsradiekompenseringen är identiskt med planet för skyddsområdesdefinitionerna.

4.3 Kontroll av kränkning av skyddsområden, arbetsfältsbegränsning och mjukvarugränsbrytare (CALCPOSI)

Funktion

Funktionen CALCPOSI kontrollerar i arbetsstyckskoordinatsystemet (WKS), om utgående från startpositionen **geometriaxlarna** kan köra en föreskriven väg, utan att kränka aktiva begränsningar. För den fall att förflyttningsvägen på grund av begränsningar inte kan köras fullständigt, returneras ett positivt, decimalt kodat statusvärde och den maximalt möjliga förflyttningsvägen.

Definition

```
INT CALCPOSI (VAR REAL[3] <Start>, VAR REAL[3] <Dist>, VAR REAL[5]
<Limit>, VAR REAL[3] <MaxDist>, BOOL <MeasSys>, INT <TestLim>)
```

Syntax

```
<Status> = CALCPOSI (VAR <Start>, VAR <Dist>, VAR <Limit>, VAR
<MaxDist>, <MeasSys>, <TestLim>)
```

Betydelse

CALCPOSI(...):	Fördefinierad funktion för test av begränsningskränkningar med avseende på geometriaxlarna.	
	Fördekoderingsstopp:	nej
	Ensam i blocket:	ja

<Status>: (del 1)	Funktionens returvärde. Negativa värden indikerar feltillstånd.	
	Datatyp:	INT
	Värdeområde:	$-8 \leq x \leq 100000$
	Värde:	0 Förflyttningsvägen kan köras fullständigt
		-1 I <Limit> är minst en komponent negativ.
		-2 Fel vid en transformationsberäkning. Exempel: Förflyttningsvägen leder genom en singularitet, så att axelpositionerna inte är definierade.
		-3 Den angivna förflyttningsvägen <Dist> och den maximalt möjliga förflyttningsvägen <MaxDist> är linjärt beroende. Observera Kan endast uppträda i samband med <TestLim>, bit 4 == 1.
		-4 Projektionen av den i <Dist> ingående förflyttningsriktningen på begränsningsytan är nollvektorn resp. förflyttningsriktningen står vinkelrätt mot den kränkta begränsningsytan. Observera Kan endast uppträda i samband med <TestLim>, bit 5 == 1.
		-5 I <TestLim> är bit 4 == 1 OCH bit 5 == 1
		-6 Minst en maskinaxel, som måste betraktas för kontrollen av förflyttningsgränserna, är inte referenskörd.
		-7 Funktion Undvikande av kollision: Ogiltig definition av den kinematiska kedjan eller skyddsområdena.
		-8 Funktion Undvikande av kollision: Funktionen kan inte utföras på grund av minnesbrist.
<Status>: (del 2)	Entalsplats	
	Observera Har flera gränser samtidigt kränkts, anmäls den som leder till den största inskränkningen av den föreskrivna förflyttningsvägen.	
	Värde:	1 Mjukvarugränsbrytare begränsar förflyttningsvägen
		2 Arbetsfältsbegränsning begränsar förflyttningsvägen
		3 Skyddsområden begränsar förflyttningsvägen
		4 Funktion Undvikande av kollision: Skyddsområden begränsar förflyttningsvägen
	Tiotalsplats	
	Värde:	1x Begynnelsevärdet kränker gränsen
		2x Den föreskrivna räta linjen kränker gränsen. Detta värde returneras också när slutpunkten själv inte kränker någon gräns men på vägen från start- till slutpunkt en kränkning av ett gränsvärde skulle kunna uppträda (t.ex. körning genom ett skyddsområde, böjda mjukvarugränsbrytare i WKS vid ej linjära transformationer, t.ex. Transmit).

4.3 Kontroll av kränkning av skyddsområden, arbetsfältsbegränsning och mjukvarugränsbrytare (CALCPOSI)

<Status>: (del 3)	Hundratalsplats		
	Värde:	1xx	OCH entalsplats == 1 eller 2: Det positiva gränsvärdet är kränkt.
			OCH entalsplats == 3 ¹⁾ : Ett NC-spec. skyddsområde har kränkts.
		2xx	OCH entalsplats == 1 eller 2: Det negativa gränsvärdet är kränkt.
			OCH entalsplats == 3 ¹⁾ : Ett kanalspecifikt skyddsområde har kränkts.
<Status>: (del 4)	Tusentalsplats		
	Värde:	1xxx	OCH entalsplats == 1 eller 2: Faktor, med vilken axelnumret multipliceras, som kränker gränsen. Räkningen av axlarna börjar vid 1. Referens: - Mjukvarugränsbrytare: Maskinaxlar - Arbetsfältsbegränsning: Geometriaxlar OCH entalsplats == 3 ¹⁾ : Faktor, med vilken numret för det kränkta skyddsområdet multipliceras.
<Status>: (del 5)	Etthundratusentalsplats		
	Värde:	0xxxxx	Etthundratusentalsplats == 0: <Dist> förblir oförändrat
		1xxxxx	I <Dist> returneras en riktningvektor som definierar den fortsatta rörelseriktningen på begränsningsytan. Kan endast uppträda vid följande randvillkor: - Mjukvarugränsbrytare eller arbetsfältsbegränsning kränkt (inte i startpunkten) - En transformation är inte aktiv <TestID>, bit 4 eller bit 5 == 1
<Start>:	Referens till en vektor med startpositionerna: <Start> [0]: 1:a geometriaxeln <Start> [1]: 2:a geometriaxeln <Start> [2]: 3:e geometriaxeln		
	Parametertyp:	Ingång	
	Datatyp:	VAR REAL[3]	
	Värdeområde:	-max. REAL-värde ≤ x[<n>] ≤ +max. REAL-värde	

<Dist>:	Referens till en vektor.	
	Ingång: inkrementell förflyttningsväg • <Dist> [0]: 1:a geometriaxeln • <Dist> [1]: 2:a geometriaxeln • <Dist> [2]: 3:e geometriaxeln	
	Utgång (endast vid inställd etthundratusentalsplats i <Status>):	
	<Dist> innehåller som utgångsvärde en enhetsvektor v , som definierar den fortsatta förflyttningsriktningen i WKS.	
	Fall 1: Bildande av vektorn v vid <TestID>, bit 4 == 1 Ingångsvektorena <Dist> och <MaxDist> spänner upp rörelseplanet. Detta plan skärs med den kränkta rörelseytan. Skärningslinjen mellan de båda planen definierar riktningen för vektorn v . Därvid väljs orienteringen (förtecknet) så att vinkeln mellan ingångsvektorn <MaxDist> och v inte är större än 90 grader.	
	Fall 2: Bildande av vektorn v vid <TestID>, bit 5 == 1 Vektorn v är enhetsvektorn i riktningen för projektionen för den i <Dist> ingående förflyttningsvektorn på begränsningsytan. Är projektionen av förflyttningsvektorn på begränsningsytan för nollvektorn returneras ett fel.	
<Limit>:	Parametertyp:	In-/utgång
	Datatyp:	VAR REAL[3]
	Värdeområde:	-max. REAL-värde ≤ x[<n>] ≤ +max. REAL-värde
	Referens till ett fält med längden 5.	
	• <Limit> [0 - 2]: Minsta avstånden för geometriaxlarna till begränsningarna: – <Limit> [0]: 1:a geometriaxeln – <Limit> [1]: 2:a geometriaxeln – <Limit> [2]: 3:e geometriaxeln	
	Minsta avstånden respekteras vid: – Arbetsfältsbegränsning: Inga inskränkningar – Mjukvarugränsbrytare: Om ingen transformation är aktiv, eller en transformation är aktiv, vid vilken en entydig tillordning av geometriaxlarna till linjära maskinaxlar är möjlig t.ex. 5-axeltransformationer.	
<Limit>:	• <Limit> [3]: Innehåller det minsta avståndet för linjära maskinaxlar, som t.ex. på grund av en ej linjär transformation inte kan tillordnas till någon geometriaxel. Detta värde används dessutom som gränsvärde vid övervakningen av de konventionella skyddsområdena och skyddsområdena för undvikande av kollisioner.	
	• <Limit> [4]: Innehåller det minsta avståndet för roterande maskinaxlar, som t.ex. på grund av en ej linjär transformation inte kan tillordnas till någon geometriaxel.	
	Observera Detta värde blir verksamt endast vid övervakningen av mjukvarugränsbrytare för speciella transformationer.	
	Parametertyp:	Ingång
	Datatyp:	VAR REAL[5]
	Värdeområde:	-max. REAL-värde ≤ x[n] ≤ +max. REAL-värde

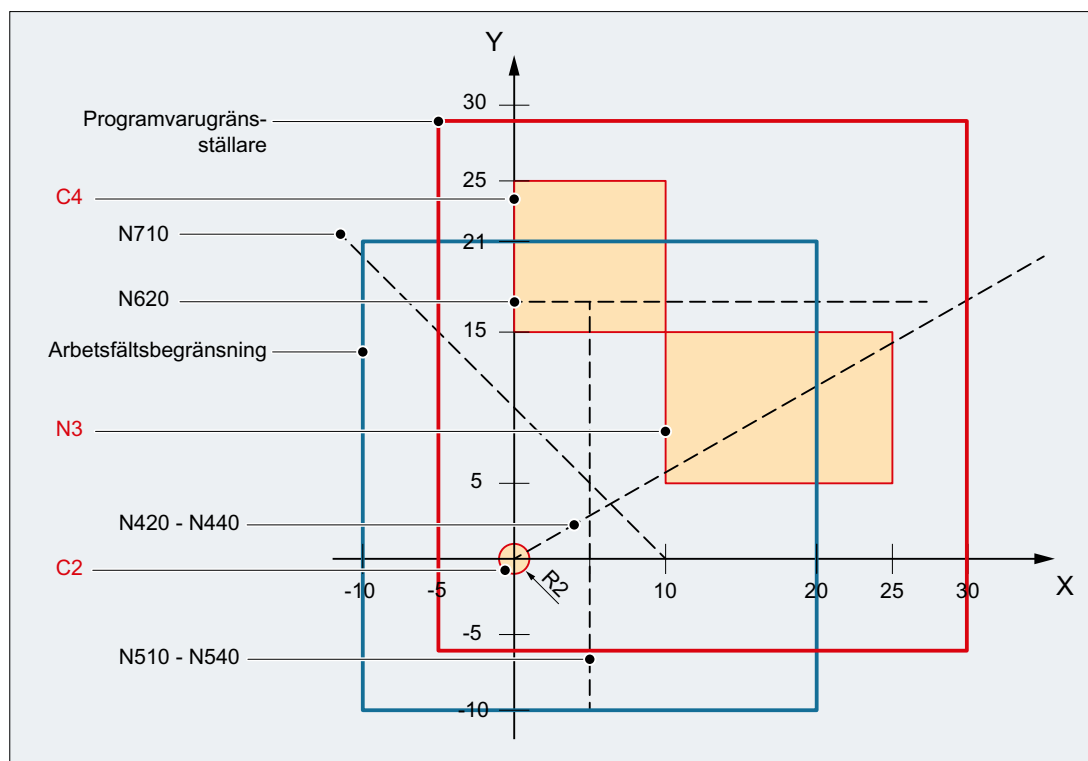
4.3 Kontroll av kränkning av skyddsområden, arbetsfältsbegränsning och mjukvarugränsbrytare (CALCPOSI)

<MaxDist>:	Referens till en vektor med den inkrementella förflyttningsvägen, vid vilken det föreskrivna minsta avståndet för en axelgräns inte underskrids av någon av de deltagande maskinaxlarna:		
	• <Dist> [0]: 1:a geometriaxeln • <Dist> [1]: 2:a geometriaxeln • <Dist> [2]: 3:e geometriaxeln		
	Är förflyttningsvägen inte inskränkt, är innehållet i denna returparameter lika med innehållet i <Dist>.		
	Vid <TestID>, bit 4 == 1: <Dist> och <MaxDist>		
	<MaxDist> och <Dist> måste som ingångsvärden innehålla vektorer som spänner upp ett rörelseplan. De båda vektorerna måste vara linjärt oberoende av varandra. Beloppet för <MaxDist> är valfritt. För beräkning av rörelseriktningen se beskrivningen till <Dist>.		
	Parametertyp:	Utgång	
	Datotyp:	VAR REAL[3]	
	Värdeområde:	-max. REAL-värde ≤ x[<n>] ≤ +max. REAL-värde	
<MeasSys>:	Måttsystem (inch / metriskt) för positions- och längduppgifter (option)		
	Datotyp:		BOOL
	Värde:	FALSE (Default)	Måttsystem motsvarande det aktuellt aktiva G-kommandot ur G-gruppen 13 (G70, G71, G700, G710). Observera Vid aktiv G70 och grundsystem metriskt eller aktiv G71 och grundsystem inch, levereras systemvariablerna \$AA_IW och \$AA_MW i grundsystemet och måste vid användning för CALCPOSI eventuellt räknas om.
		TRUE	Måttsystem enligt inställt grundsystem: MD52806 \$MN_ISO_SCALING_SYSTEM
<TestLim>:	Bitcodat val av de begränsningar som ska övervakas (option)		
	Datotyp:		INT
	Defaultvärde:		Bit 0, 1, 2, 3, 6, 7 == 1 (207)
	Bit	Decimalt	Betydelse
	0	1	Mjukvarugränsbrytare
	1	2	Arbetsfältsbegränsning
	2	4	Aktiverade konventionella skyddsområden
	3	8	Föraktiverade konventionella skyddsområden
	4	16	Vid kränkta mjukvarugränsbrytare resp. arbetsfältsbegränsningar i <Dist> returnera förflyttningsriktningen enligt Fall 1 (se ovan).
	5	32	Vid kränkta mjukvarugränsbrytare resp. arbetsfältsbegränsningar i <Dist> returnera förflyttningsriktningen enligt Fall 2 (se ovan).
	6	64	Aktiverade skyddsområden för undvikande av kollisioner
	7	128	Föraktiverade skyddsområden för undvikande av kollisioner
	8	256	Par av aktiverade och föraktiverade skyddsområden för undvikande av kollisioner

¹⁾ Har flera skyddsområden kränkts, lämnas det skyddsområde tillbaka, som leder till den största inskränkningen av den förskrivna förflyttningsvägen.

Exempel

Begränsningar



I exemplet visas de verksamma mjukvarugränsbrytarna och arbetsfältsbegränsningarna i X-Y-planet och följande tre skyddsområden:

- C2: Verktygsrelaterat kanalspecifikt skyddsområde, aktivt, cirkelformat, radie = 2 mm
- C4: Arbetsstycksrelaterat, kanalspecifikt skyddsområde, föraktiverat, kvadratisk, sidolängd = 10 mm
- N3: Maskinspecifikt skyddsområde, aktivt, fyrkantigt, sidolängd = 10 mm x 15 mm

NC-program

I NC-programmet definieras först skyddsområdena och arbetsfältsbegränsningarna. Sedan anropas funktionen `CALCPOSI()` med olika parameterinställningar.

Programkod

```

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

```


4.3 Kontroll av kränkning av skyddsområden, arbetsfältsbegränsning och mjukvarugränsbrytare (CALCPOSI)

Programkod

```

; Verktygsrelaterat skyddsområde C2
N70 CPROTDEF(2, TRUE, 0)
N80 G17 G1 X-2 Y0
N90 G3 I2 X2
N100 I-2 X-2
N110 EXECUTE(_PA)

; Arbetsstycksrelaterat skyddsområde C4
N120 CPROTDEF(4, FALSE, 0)
N130 G17 G1 X0 Y15
N140 X10
N150 Y25
N160 X0
N170 Y15
N180 EXECUTE(_PA)

; Maskinspecifikt skyddsområde N3
N190 NPROTDEF(3, FALSE, 0)
N200 G17 G1 X10 Y5
N210 X25
N220 Y15
N230 X10
N240 Y5
N250 EXECUTE(_PA)

; Aktivera resp. föraktivera skyddsområden
N260 CPROT(2, 2, 0, 0, 0)
N270 CPROT(4, 1, 0, 0, 0)
N280 NPROT(3, 2, 0, 0, 0)

; Definiera arbetsfältsbegränsningar
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.

```

4.3 Kontroll av kränkning av skyddsområden, arbetsfältsbegränsning och mjukvarugränsbrytare (CALCPOSI)

Programkod

```

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)
; Åter radera frames från N690 och N700
N720 TRANS
N730 _START[0] = 0.
N740 _START[1] = 10.
N750 _START[2] = 0.
; Vektorerna _DIST och _MAXDIST definierar rörelseplanet
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

```

Resultat från CALCPOSI()

N...	<Status>	<MaxDist>[0] Δ X	<MaxDist>[1] Δ Y	Anmärkningar
420	3123	8.040	4.594	N3 kränks.
430	1122	20.000	11.429	Ingen övervakning av skyddsområdena, Arbetsfältsbegränsning kränks.
440	1121	30.000	17.143	Endast övervakning av mjukvarugränsbrytare aktiv.
510	4213	0.000	0.000	Startpunkt kränkt C4
520	0000	0.000	-27.000	Föraktiverad C4 övervakas inte. Den föreskrivna vägen kan köras fullständigt.
540	2222	0.000	-25.000	På grund av _LIMIT[1]=2 inskränks förflyttningsvägen genom arbetsfältsbegränsningen.

4.3 Kontroll av kränkning av skyddsområden, arbetsfältsbegränsning och mjukvarugränsbrytare (CALCPOSI)

N...	<Status>	<MaxDist>[0] Δ X	<MaxDist>[1] Δ Y	Anmärkningar
620	4223	-13.000	0.000	Avstånd till C4 på grund av C2 och _LIMIT[3] totalt 4 mm. Avstånd C2 → N3 på 0,1 mm leder inte till inskränkning av förflyttningsvärden.
710	1221	0.000	21.213	Frame med translation och rotation aktiv. Den tillåtna förflyttningsvägen i _DIST gäller i förflyttat och roterat WKS.
820	102121	18.000	18.000	Mjukvarugränsbrytaren på Y-axeln kränks. Med <_TESTLIM> = 17 krävs beräkning av en riktning för fortsatt körning. Denna riktning står i <_DIST> (0.707, 0.0, 0.707). Den är giltig, eftersom etthundratusentalsplatsen är inställd i <_STATUS>.

Ytterligare informationer

Axeltillstånd "Refererat"

All av CALCPOSI () betraktade maskinaxlar måste vara referenskörda.

Cirkelrelaterade väguppgifter

Alla cirkelrelaterade väguppgifter interpreteras **alltid** som radieuppgift. Det ska speciellt vid planaxlar med aktiverad diameterprogrammering (DIAMON / DIAM90) tas hänsyn till detta.

Förflyttningsvägsreducering

Begränsas den angivna förflyttningsvägen för en axel, är i returvärdet <MaxDist> också förflyttningsvägarna för de andra axlarna reducerade andelsmässigt. Den resulterande ändpunkten ligger därigenom fortfarande på den föreskrivna banan.

Roterande axlar

Roterande axlar övervakas endast när de inte är några roterande Moduloaxlar.

Det är tillåtet att inga mjukvarugränsbrytare resp. arbetsfältsbegränsningar eller skyddsområden är definierade för en eller flera av de deltagande axlarna.

Mjukvarugränsbrytar- och arbetsfältsbegränsnings-status

Det tas endast hänsyn till mjukvarugränsbrytare och arbetsfältsbegränsningar, när dessa är aktiva vid utförandet av CALCPOSI (). Statusen kan t. ex. påverkas via:

- Maskindata: MD21020 \$MC_WORKAREA_WITH_TOOL_RADIUS
- Settingdata: \$AC_WORKAREA_CS_...
- NC/PLC-gränssnittssignaler: DB31, ... DBX12.2 / 3
- Kommandon: WALIMON / WALIMOF

Mjukvarugränsbrytare och transformationer

Vid CALCPOSI () kan vid olika kinematiska transformationer (t. ex. TRANSMIT) på grund av flertydigheten vid vissa ställen i förflyttningsområdet positionerna för maskinaxlarna (MKS) inte alltid bestämmas entydigt ur positionerna för geometriaxlarna (WKS). I normal förflyttningsdrift resulterar entydigheten som regel ur förhistorien och villkoret att en kontinuerlig rörelse i WKS

måste motsvara en kontinuerlig rörelse av maskinaxlarna i MKS. Vid övervakningen av mjukvarugränsbrytarna anlitas därför i sådana fall den vid utförandetidpunkten av CALCPOSI () föreliggande maskinpositionen för upplösning av mångtydigheten.

Märk

Fördekoderingsstopp

Vid användning av CALCPOSI () i samband med transformationer är det användarens ensamma ansvar, att för synkronisering av maskinaxelpositionerna med förkörning före CALCPOSI () programmera ett förkoderingsstopp (STOPRE).

Skyddsområdesavstånd och konventionella skyddsområden

Vid konventionella skyddsområden är **inte** garanterat, att vid en förflyttningsrörelse längs den föreskrivna förflyttningsvägen det i parametern <Limit>[3] angivna säkerhetsavståndet respekteras gentemot alla skyddsområden. Det är endast garanterat, att vid förlängning av den i <Dist> returnerade ändpunkten med säkerhetsavståndet i förflyttningsriktningen inget skyddsområde kränks. Den rätta linjen kan dock i sitt förlopp leda godtyckligt nära förbi ett skyddsområde.

Skyddsområdeavstånd och skyddsområden vid undvikande av kollision

Vid skyddsområden för undvikande av kollision är garanterat, att vid en förflyttningsrörelse längs den föreskrivna förflyttningsvägen, det i parametern <Limit>[3] angivna säkerhetsområdet respekteras gentemot alla skyddsområden.

Det i parametern <Limit>[3] angivna säkerhetsavståndet blir verksamt endast när följande gäller:

<Limit>[3] > (MD10619 \$MN_COLLISION_TOLERANCE)

Har i parametern <TestLim> bit 4 satts (beräkning av den fortsatta förflyttningsriktningen), är den i <DIST> ingående riktningsvektorn endast giltig, när etthundratusentalsplatsen har satts i returvärdet för funktionen (<Status>). Kan en sådan riktning inte fastställas, eftersom antingen skyddsområden har kränkts eller en transformation är aktiv, förblir ingångsvärdet i <DIST> oförändrat. Ett ytterligare felmeddelande kommer inte.

Speciella vägkommandon

5.1 Uppsöka kodade positioner (CAC, CIC, CDC, CACP, CACN)

Via de följande kommandona kan linjära och roterande axlar förflyttas via positionsnummer till i maskindatatabellen lagrade fasta axelpositioner. Denna typ av programmering betecknas som "Uppsökning av kodade positioner".

Syntax

CAC (<n>)
CIC (<n>)
CACP (<n>)
CACN (<n>)

Betydelse

CAC (<n>):	Uppsöka kodad position med positionsnummer n
CIC (<n>):	Uppsöka kodad position, utgående från det aktuella positionsnumret, n-positionsplatser fram (+n) eller tillbaka (-n)
CDC (<n>):	Uppsöka kodad position med positionsnummer n på den kortaste vägen (endast för roterande axlar)
CACP (<n>):	Uppsöka kodad position med positionsnummer n i positiv riktning (endast för roterande axlar)
CACN (<n>):	Uppsöka kodad position med positionsnummer n i negativ riktning (endast för roterande axlar)
<n>:	Positionsnummer inom maskindata-tabellen Värdeområde: 0, 1, ... (max. antal tabellplatser - 1)

Exempel: Uppsöka en positioneringsaxels kodade positioner

Programmeringskod	Kommentar
N10 FA[B]=300	; Matning för positioneringsaxel B
N20 POS[B]=CAC(10)	; Uppsöka kodad position med positionsnummer 10
N30 POS[B]=CIC(-4)	; Uppsöka kodad position med "aktuellt positionsnummer" - 4

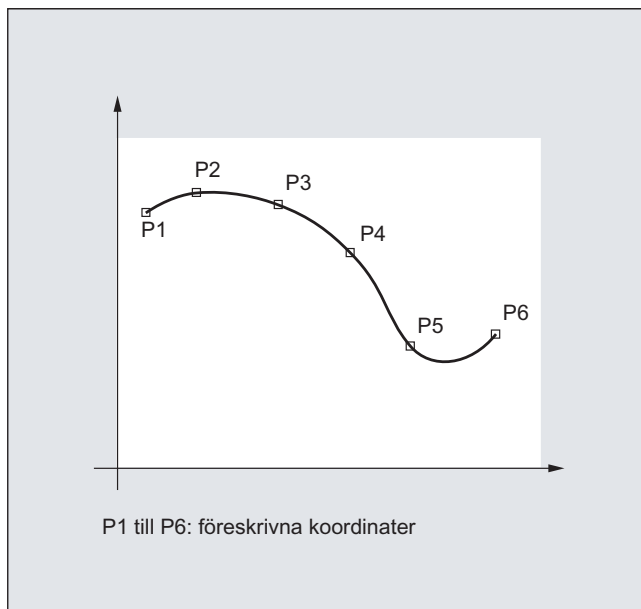
Litteratur

- Funktionshandbok Tilläggfunktioner; Delningsaxlar (T1)
- Funktionshandbok Synkronaktioner

5.2 Spline-interpolering (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Valfritt böjda konturer på arbetsstycken kan inte beskrivas analytiskt exakt. Sådana konturer närmar man sig därför genom ett begränsat antal stödpunkter t.ex. vid digitalisering av ytor. För att skapa den digitaliserade ytan på ett arbetsstycke måste stödpunkterna vara förbundna till en konturbeskrivning. Detta möjliggör spline-interpolering.

En spline definierar en kurva som är sammansatt av polynom av 2:a eller 3:e graden. Egenskaperna vid stödpunkterna till en spline kan definieras **beroende på den använda spline-typen**.



Följande spline-typer står vid SINUMERIK solution line till förfogande:

- A-spline
- B-spline
- C-spline

Syntax

Allmänt:

```
ASPLINE X... Y... Z... A... B... C...  
BSPLINE X... Y... Z... A... B... C...  
CSPLINE X... Y... Z... A... B... C...
```

Vid B-spline kan dessutom programmeras:

```
PW=<n>  
SD=2  
PL=<Wert>
```

Vid A- och C-spline kan dessutom programmeras:

```
BAUTO / BNAT / BTAN
```

5.2 Spline-interpolering (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

EAUTO / ENAT / ETAN

Betydelse

Spline-interpoleringstyp:				
ASPLINE:		Kommando för tillkoppling av A-spline-interpolering		
BSPLINE:		Kommando för tillkoppling av B-spline-interpolering		
CSPLINE:		Kommando för tillkoppling av C-spline-interpolering		
		Kommandona ASPLINE, BSPLINE och CSPLINE är modalt verksamma och hör till gruppen vägkommandon.		
Stödpunkter resp. kontrollpunkter:				
X... Y... Z... A... B... C...		Position i kartesiska koordinater		
Punktvikt (endast B-spline):				
PW:		Med kommandot PW är programmeringen av en så kallad "Punktvikt" möjlig för varje stödpunkt.		
<n>:		"Punktvikt"		
		Värdeområde:	$0 \leq n \leq 3$	
		Steglängd:	0.0001	
		Verkan:	$n > 1$	Kurvan spänns kraftigare av kontrollpunkten.
			$n < 1$	Kurvan spänns mindre kraftigt av kontrollpunkten.
Spline-grad (endast B-spline):				
SD:		Standardmässigt används en polygon av 3:e graden. Genom programmering av SD=2 kan dock också en polygon av 2:a graden användas.		
Knutavstånd (endast B-spline):				
PL:		Knutavstånden beräknas lämpligt internt. Styrningen kan dock också bearbeta föreskrivna knutavstånd, som anges som så kallad parameterintervall-längd med kommandot PL.		
<Wert>:		Parameter-intervall-längd		
		Värdeområde:	som vägmått	
Övergångsbeteende vid början av spline-kurvan (endast A- eller C-spline):				
BAUTO:		Ingen uppgift för övergångsbeteendet. Början resulterar ur läget för den första punkten.		
BNAT:		Böjning noll		
BTAN:		Tangentiell övergång till det föregående blocket (raderläge)		
Övergångsbeteende vid slutet av spline-kurvan (endast A- eller C-spline):				
EAUTO:		Ingen uppgift för övergångsbeteendet. Slutet resulterar ur läget för den sista punkten.		
ENAT:		Böjning noll		

ETAN:	Tangentiell övergång till det föregående blocket (raderläge)

Märk

Det programmerbara övergångsbeteendet har inget inflytande på B-splinen. B-spline är i start- och slutpunkt alltid tangentiell till kontrollpolygonen.

Randvillkor

- Verktygsradiekompenseringen kan användas.
- Kollisionsövervakning sker i projektionen på planet.

Exempel

Exempel 1: B-spline

Programkod 1 (alla vikter 1)

```

N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20
N40 X20 Y40
N50 X30 Y30
N60 X40 Y45
N70 X50 Y0

```


5.2 Spline-interpolering (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Programkod 2 (olika vikter)

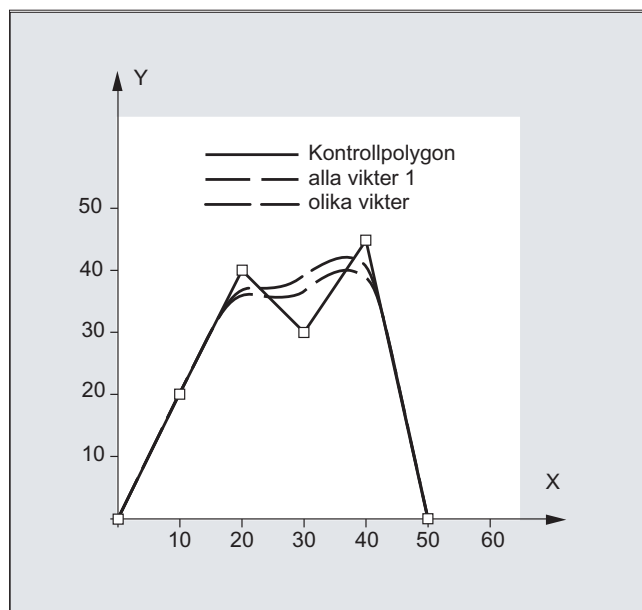
```
N10 G1 X0 Y0 F300 G64
N20 BSPLINE
N30 X10 Y20 PW=2
N40 X20 Y40
N50 X30 Y30 PW=0.5
N60 X40 Y45
N70 X50 Y0
```

Programkod 3 (kontrollpolygon)

Kommentar

```
N10 G1 X0 Y0 F300 G64
N20
N30 X10 Y20
N40 X20 Y40
N50 X30 Y30
N60 X40 Y45
N70 X50 Y0
```

; utgår



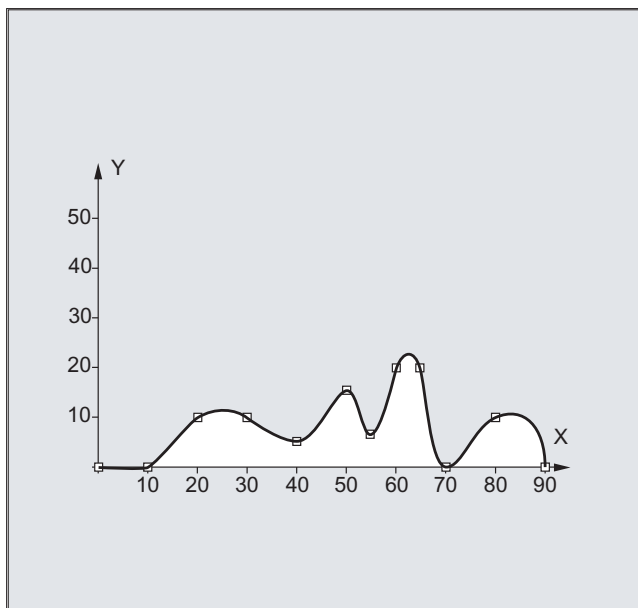
Exempel 2: C-spline, vid start och slut böjning noll

Programkod

```
N10 G1 X0 Y0 F300
N15 X10
N20 BNAT ENAT
N30 CSPLINE X20 Y10
```

Programkod

```
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
```



Exempel 3: Spline-interpolering (A-spline) och koordinattransformation (ROT)

Huvudprogram:

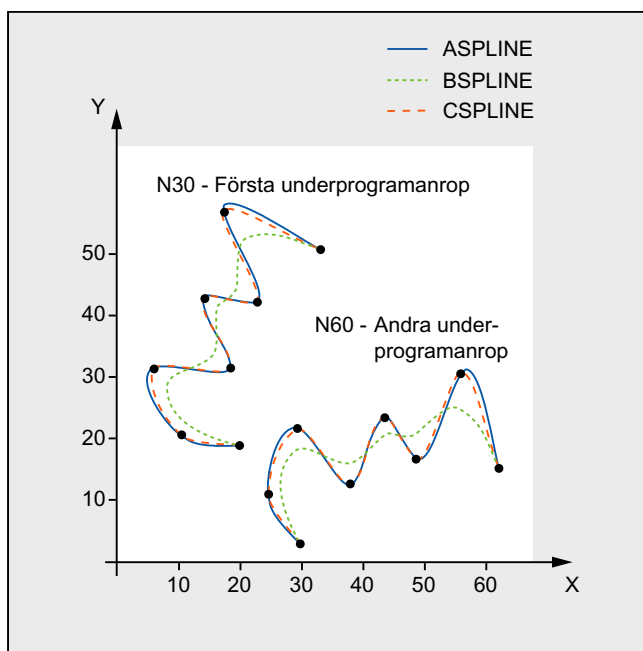
Programkod	Kommentar
N10 G00 X20 Y18 F300 G64	; Köra till startpunkt.
N20 ASPLINE	; Aktivera interpoleringstyp A-spline.
N30 KONTUR	; Första anrop av underprogrammet.
N40 ROT Z-45	; Koordinattransformation: Rotation av WKS med -45° runt Z-axeln.
N50 G00 X20 Y18	; Uppsöka konturstartpunkt.
N60 KONTUR	; Andra anrop av underprogrammet.
N70 M30	; Programslut

5.2 Spline-interpolering (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Underprogram "Kontur" (innehåller stödpunkt-koordinaterna):

Programkod
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

I den följande bilden finns förutom spline-kurvan som resulterar ur programexemplet (ASPLINE), också de spline-kurvor som skulle resultera vid aktivering av en B- eller C-spline-interpolering (BSPLINE, CSPLINE):



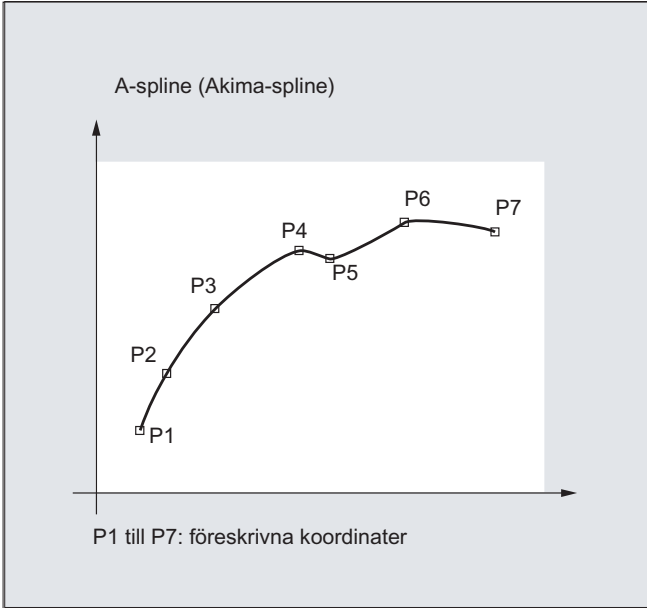
Ytterligare informationer

Fördelar med spline-interpoleringen

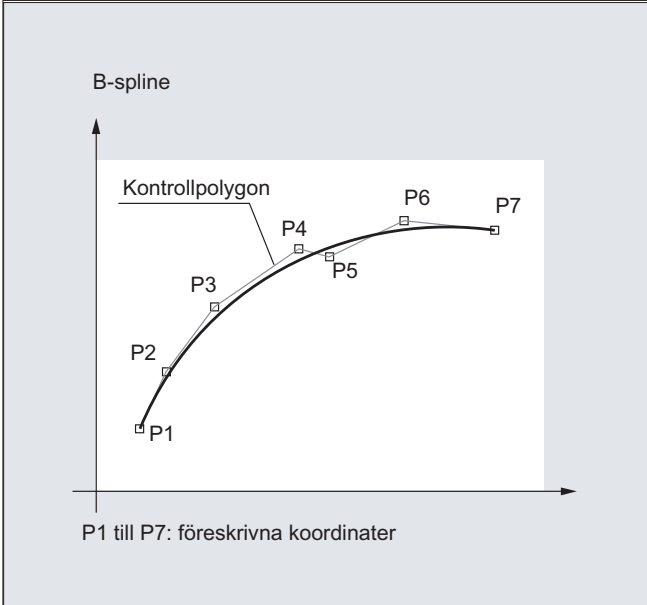
Genom användning av spline-interpoleringen låter sig i motsats till användning av linjeblock G01, följande fördelar uppnås:

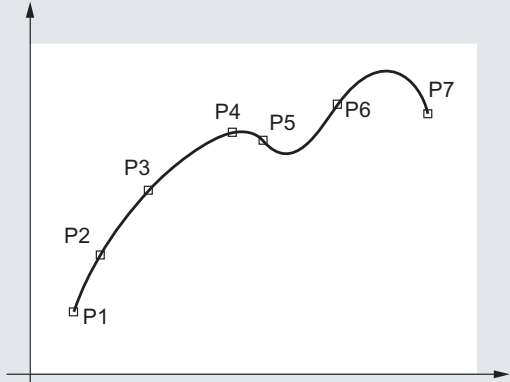
- Reducering av antalet nödvändiga detaljprogramblock för beskrivning av konturen
- Mjukare, mekaniskkonande kurvförlopp vid övergångar mellan detaljprogramblocken

Egenskaper och användning av olika spline-typer

Spline-typ	Egenskaper och användning
A-spline	A-spline (Akima-spline)  P1 till P7: föreskrivna koordinater Egenskaper: • Förlöper exakt genom de föreskrivna stödpunkterna. • Kurvförloppet är tangent- men inte böjningskontinuerligt. • Skapar knappast oönskade svängningar. • Det område som påverkas av stödpunktsändringar är lokalt dvs. förändring av en stödpunkt inverkar endast på upp till max. 6 närliggande stödpunkter. Användning: A-spline lämpar sig framför allt för interpolering av kurvförlopp med stora stigningsändringar (t.ex. trappformade kurvförlopp).

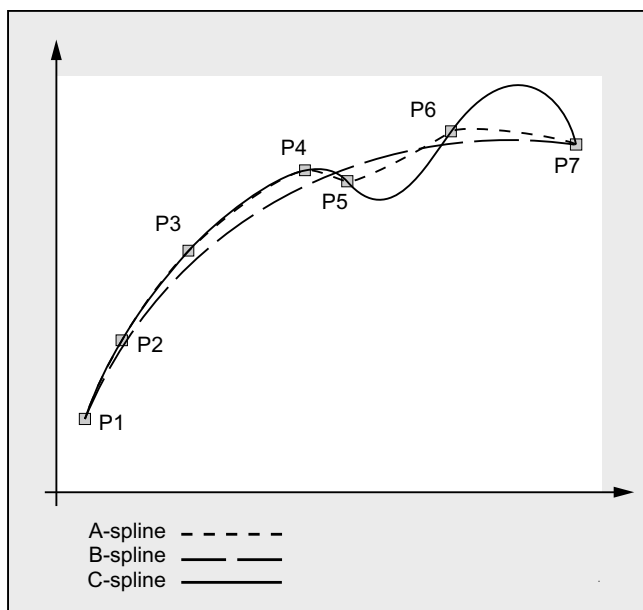
5.2 Spline-interpolering (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Spline-typ	Egenskaper och användning
B-spline	 Egenskaper: • Förlöper inte genom de föreskrivna stödpunkterna utan endast i deras närhet. Kurvan spänns genom stödpunkterna. Genom viktning av stödpunkterna med en faktor kan kurvförloppet dessutom påverkas. • Kurvförloppet är tangent- och böjningskontinuerligt. • Skapar inga oönskade svängningar. • Det område som påverkas av stödpunktsändringar är lokalt dvs. förändring av en stödpunkt inverkar endast på upp till max. 6 närliggande stödpunkter. Användning: B-spline är primärt tänkt som gränssnitt till CAD-systemen.

Spline-typ	Egenskaper och användning
C-spline	C-spline (kubisk spline)  P1 till P7: föreskrivna koordinater Egenskaper: • Förlöper exakt genom de föreskrivna stödpunkterna. • Kurvförloppet är tangent- och böjningskontinuerligt. • Skapar ofta oönskade svängningar, speciellt på ställen med stora stigningsändringar. • Det område som påverkas av stödpunktsändringar är globalt dvs. förändring av en stödpunkt inverkar på hela kurvförloppet. Användning: C-spline kan väl användas när stödpunkterna ligger på en analytiskt känd kurva (cirkel, parabel, hyperbel)

5.2 Spline-interpolering (ASPLINE, BSPLINE, CSPLINE, BAUTO, BNAT, BTAN, EAUTO, ENAT, ETAN, PW, SD, PL)

Jämförelse av de tre spline-typerna vid lika stödpunkter



Minsta antal spline-block

G-kommandona `ASPLINE`, `BSPLINE` och `CSPLINE` förbinder blockslutpunkter med splines. Därtill måste under förkörningen en rad block (slutpunkter) beräknas samtidigt. Buffertens storlek för bearbetningen uppgår standardmässigt till 10 block. Inte varje blockinformation är en spline-slutpunkt. Styrningen behöver dock av 10 block ett visst antal spline-slutpunkt-block:

Spline-typ	Minsta antal spline-block
A-spline:	Av 10 block måste minst 4 vara spline-block. Kommentarblock och parameterräkningar räknas inte härvid.
B-spline:	Av 10 block måste minst 6 vara spline-block. Kommentarblock och parameterräkningar räknas inte härvid.
C-spline:	Det nödvändiga minsta antalet spline-block resulterar ur följande summa: Värde från MD20160 <code>\$MC_CUBIC_SPLINE_BLOCKS + 1</code> I MD20160 förs det antal punkter in via vilka spline-avsnittet beräknas. Standardinställningen uppgår till 8. Av 10 block måste därför i standardfall minst 9 vara spline-block.

Märk

När det tolererbara värdet underskrids matas ett larm ut likaså när en i spline deltagande axel programmeras som positioneringsaxel.

Sammanfattning av korta spline-block

Vid spline-interpoleringen kan korta spline-block uppstå, som leder till en onödig reducering av banhastigheten. Med funktionen "Sammanfattning av korta spline-block" kan dessa block

sammanfattas så att den resulterande blocklängden är tillräckligt stor och inte leder till en minskning av banhastigheten.

Funktionen aktiveras via det kanalspecifika maskindatumet:

MD20488 \$MC_SPLINE_MODE (inställning för spline-interpolering)

Litteratur:

Funktionshandbok Grundfunktioner; Bansstyrningsdrift, precisionsstopp, LookAhead (B1),
Kapitel: Sammanfattning av korta spline-block

5.3 Spline-förband (SPLINEPATH)

De axlar i spline-förbandet som ska interpoleras väljs med kommandot `SPLINEPATH`. Upp till åtta banaxlar är möjligt vid spline-interpoleringen.

Märk

Programmeras inte `SPLINEPATH` explicit, så förflyttas de tre första axlarna i kanalen som spline-förband.

Syntax

Fastläggandet av spline-förbandet görs i ett åtskilt block:

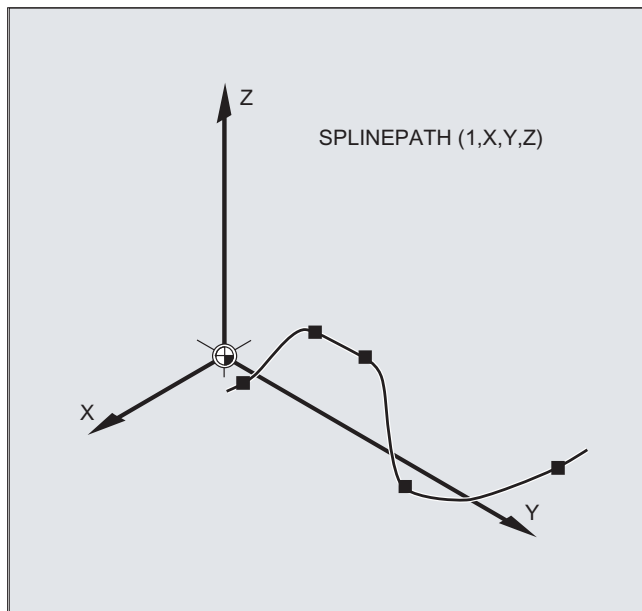
```
SPLINEPATH (n, X, Y, Z, ...)
```

Betydelse

<code>SPLINEPATH:</code>	Kommando för fastläggande av ett spline-förband
<code>n:</code>	=1 (fast värde)
<code>X, Y, Z, ... :</code>	Beteckning för de banaxlar i spline-förbandet som ska interpoleras

Exempel: Spline-förband med tre banaxlar

Programkod	Kommentar
N10 G1 X10 Y20 Z30 A40 B50 F350	
N11 SPLINEPATH(1,X,Y,Z)	; Spline-förband
N13 CSPLINE BAUTO EAUTO X20 Y30 Z40 A50 B60	; C-spline:
N14 X30 Y40 Z50 A60 B70	; Stödpunkter
...	
N100 G1 X... Y...	; Bortval spline-interpolering



5.4 Koppla till/från NC-block-kompression (COMPON, COMPCURV, COMPCAD, COMPSURF, COMPOF)

Funktionerna för kompression av linjärblock (och beroende av parametreringen även cirkel- och/eller snabbgångsblock) kopplas till/från med G-kommandona till G-gruppen 30. Kommandona är modalt verksamma.

Syntax

```
COMPON / COMPCURV / COMPCAD / COMPSURF
...
COMPOF
```

Betydelse

COMPON:	Tillkoppling av kompressor-funktion COMPON.
COMPCURV:	Tillkoppling av kompressor-funktion COMPCURV.
COMPCAD:	Tillkoppling av kompressor-funktion COMPCAD.
COMPSURF:	Tillkoppling av kompressor-funktion COMPSURF.
COMPOF:	Frånkoppling av den aktuellt aktiva kompressor-funktionen

Märk

För ytterligare förbättring av ytkvaliteten kan översläpningsfunktionen G642 och jerkbegränsningen SOFT användas. Dessa kommandon ska skrivas i programmets början.

Exempel: COMPCAD

Programkod	Kommentar
N10 G00 X30 Y6 Z40	
N20 G1 F10000 G642	; Tillkoppling: Översläpningsfunktion G642
N30 SOFT	; Tillkoppling: Jerkbegränsning SOFT
N40 COMPCAD	; Tillkoppling: Kompressor-funktion COMPCAD
N50 STOPFIFO	
N24050 Z32.499	; 1:a Förflyttningsblock
N24051 X41.365 Z32.500	; 2:a Förflyttningsblock
...	
N99999 X... Z...	; sista förflyttningsblocket
COMPOF	; Kompressor-funktion från.
...	

5.5 Polynom-interpolering (POLY, POLYPATH, PO, PL)

I egentlig mening handlar det vid polynom-interpoleringen (POLY) inte om en spline-interpoleringstyp. Den är i första hand tänkt som gränssnitt för programmeringen av externt skapade spline-kurvor. Härvid kan spline-avsnitten programmeras direkt.

Denna interpoleringstyp avlastar NC från beräkningen av polynom-koefficienterna. Den kan användas optimalt när koefficienterna kommer direkt från ett CAD-system eller postprocessor.

Syntax

Polynom 3:e graden.

```
POLY PO[X]=(xe,a2,a3) PO[Y]=(ye,b2,b3) PO[Z]=(ze,c2,c3) PL=n
```

Polynom 5:e graden och ny polynomsyntax:

```
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")
```

Märk

Summan av de i ett NC-block programmerade polynom-koefficienterna och axlarna får inte överskrida det maximalt tillåtna axelantalet per block.

Betydelse

POLY :	Tillkoppling av polynom-interpoleringen med ett block med POLY.
POLYPATH :	Polynom-interpolering selekterbar för de båda axelgrupperna AXIS eller VECT
PO[Achsbezeichner/Variable] :	Slutpunkter och polynom-koefficienter
X, Y, Z :	Axelbeteckning
xe, ye, ze :	Uppgift för slutpositionen för respektive axel; värdeområde som vägmått
a2, a3, a4, a5 :	Koefficienterna a_2 , a_3 , a_4 och a_5 skrivs med sina värden; värdeområde som vägmått. Den var gång sista koefficienten kan utgå när den har värdet noll.
PL :	Längd på det parameterintervall på vilket polynomen är definierade (definitionsområde för funktionen $f(p)$). Intervallet börjar alltid vid 0, p kan anta värden från 0 till PL. Teoretiskt värdeområde för PL: 0,0001 ... 99 999,9999 Observera: PL-värdet gäller för det block i vilket det står. Är ingen PL programmerad, verkar PL=1.

Till-/frånkoppling av polynom-interpoleringen

Polynom-interpoleringen kopplas i detaljprogrammet till med G-kommandot `POLY`.

G-kommandot `POLY` hör tillsammans med `G0`, `G1`, `G2`, `G3`, `ASPLINE`, `BSPLINE` och `CSPLINE` till 1:a G-gruppen.

Axlar som är programmerade med endast namn och slutpunkt (t.ex. `X10`), förflyttas linjärt. Är alla axlar i ett NC-block så programmerade förhåller sig styrningen som vid `G1`.

Polynom-interpoleringen kopplas genom programmeringen av ett annat kommando i den 1:a G-gruppen (t.ex. `G0`, `G1`) implicit åter från.

Polynomkoefficient

`PO`-värdet (`PO[ ]=`) resp. `...=PO(...)` anger alla polynom-koefficienter för en axel. I enlighet med polynomets grad anges flera värden skilda åt med komma. Inom ett block är olika polynomgrader möjliga för olika axlar.

Underprogram POLYPATH

Med `POLYPATH(...)` kan polynom-interpoleringen frigges selektivt för vissa axelgrupper:

Endast banaxlar och extraaxlar: `POLYPATH("AXES")`

Endast orienteringsaxlar: `POLYPATH("VECT")`

(vid förflyttning med orienterings-transformation)

De ej frigivna axlarna förflyttas linjärt.

Standardmässigt är polynom-interpoleringen frigiven för båda axelgrupperna.

Genom programmering utan parameter `POLYPATH( )` inaktiveras polynom-interpoleringen för alla axlar.

Exempel

Programkod	Kommentar
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	; Polynom-interpolering till
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	; blandade uppgifter för axlarna
N27 PO[X]=(10,2.5) PO[Y]=(2.3)	; ingen PL programmerad; PL=1 verkar
N30 G1 X... Y... Z.	; polynom-interpolering från
...	

Exempel: Ny polynomsyntax

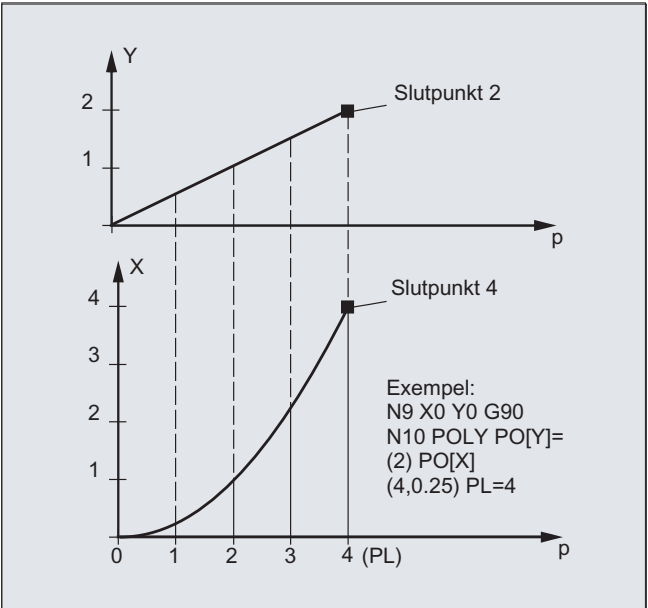
Fortfarande giltig polynomsyntax	Ny polynomsyntax
PO[Achsbezeichner]=(.. , ..)	Achsbezeichner=PO(.. , ..)
PO[PHI]=(.. , ..)	PHI=PO(.. , ..)
PO[PSI]=(.. , ..)	PSI=PO(.. , ..)
PO[THT]=(.. , ..)	THT=PO(.. , ..)
PO[]=(.. , ..)	PO(.. , ..)
PO[variable]=IC(.. , ..)	variable=PO IC(.. , ..)

Exempel: Kurva i X/Y-planet

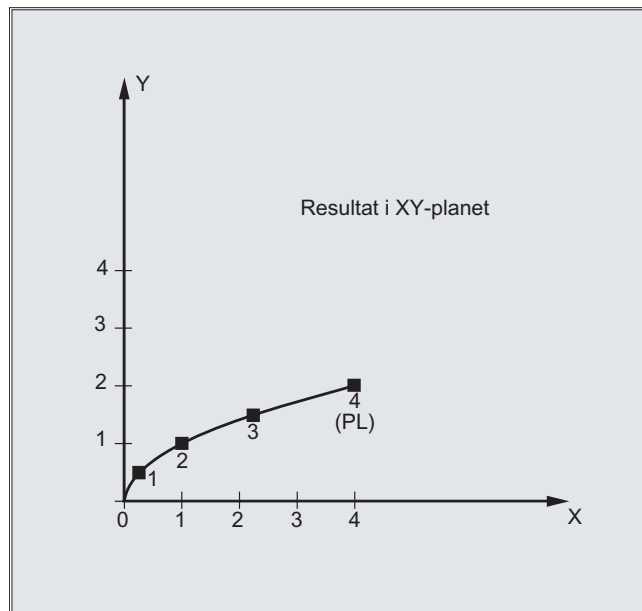
Programmering

Programkod
N9 X0 Y0 G90 F100
N10 POLY PO[Y]=(2) PO[X]=(4,0.25) PL=4

Förlopp för kurvorna X(p) och Y(p)



Förlopp för kurvan i XY-planet



Beskrivning

Den allmänna formen för polynom-funktionen lyder:

$$f(p) = a_0 + a_1p + a_2p^2 + \dots + a_np^n$$

med: a_i : konstanta koefficienter ($i = 0, 1, \dots, n$)

p : parametrar

I styrningen kan maximalt polynom av 5:e graden programmeras:

$$f(p) = a_0 + a_1p + a_2p^2 + a_3p^3 + a_4p^4 + a_5p^5$$

Genom att belägga koefficienterna med konkreta värden kan olika kurvförlopp som linjer, parabler och potensfunktioner skapas.

En rät linje skapas av $a_2 = a_3 = a_4 = a_5 = 0$:

$$f(p) = a_0 + a_1p$$

Vidare gäller:

a_0 : axelposition i slutet av föregående blocket

$p = PL$

$$a_1 = (x_E - a_0 - a_2 \cdot p^2 - a_3 \cdot p^3) / p$$

Det är möjligt att programmera polynom **utan** att polynom-interpoleringen aktiverades med G-kommandot `POLY`. I detta fall interpoleras inte de programmerade polynomen, utan de programmerade ändpunkterna på axlarna uppsöks linjärt (G1). Först efter explicit aktivering av polynom-interpoleringen i detaljprogrammet (`POLY`) körs de programmerade polynomen också som sådana.

Egenskap: Nämnar-polynom

För geometriaxlarna kan med `PO[] = (...)` utan angivande av ett axelnamn också ett gemensamt nämnar-polynom programmeras dvs. rörelsen hos geometriaxlarna interpoleras som kvot av två polynom.

Därmed låter sig t.ex. koniska snitt (cirkel, ellips, parabel, hyperbel) framställas exakt.

Exempel:

Programkod	Kommentar
POLY G90 X10 Y0 F100	; Geometriaxlarna förflyttar sig linjärt till positionen X10 Y0.
PO[X]=(0,-10) PO[Y]=(10) PO[]=(2,1)	; Geometriaxlarna förflyttar sig med fjärdedels cirkel till X0 Y10.

Den konstanta koefficienten (a_0) i nämnarpolynomet anges alltid med 1. Den programmerade slutpunkten är oberoende av G90 / G91.

Ur de programmerade värdena beräknas $X(p)$ och $Y(p)$ till:

$$X(p) = (10 - 10 * p^2) / (1 + p^2)$$

$$Y(p) = 20 * p / (1 + p^2)$$

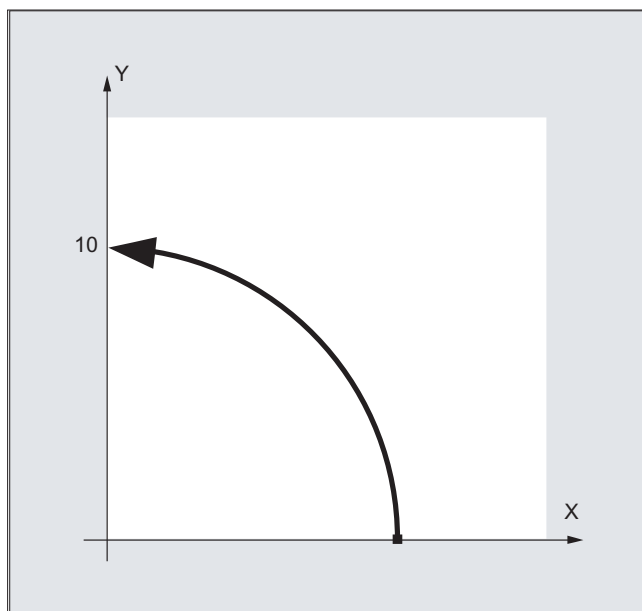
$$\text{med } 0 \leq p \leq 1$$

De programmerade begynnelsepunkterna, slutpunkterna, koefficient a_2 och $PL=1$ leder till följande mellanresultat:

$$\text{Täljare (X)} = 10 + 0 * p - 10 * p^2$$

$$\text{Täljare (Y)} = 0 + 20 * p + 0 * p^2$$

$$\text{Nämnare} = 1 + p^2$$



Vid tillkopplad polynom-interpolering avvisas programmeringen av ett nämnarpolynom med nollställetn inom intervallet $[0, PL]$ med ett larm. På rörelsen hos extraaxlar har nämnarpolynomet inget inflytande.

Märk

En verktygsradiekompensering kan vid polynom-interpoleringen med G41, G42 kopplas till och användas som för linjär eller cirkelinterpolering.

5.6 Inställbar banreferens (SPATH, UPATH)

Vid polynominterpolering (POLY, ASPLINE, BSPLINE, CSPLINE, COMCON, COMPCURV) förskrivs positionerna för banaxlarna i av polynom $p_i(U)$. Kurvparametern U flyttar sig därvid inom ett NC-block från 0 till 1.

Med FGROUП väljs de axlar (FGROUП-axlar) till vilka banmatningen F hänför sig. En interpolering med konstant hastighet på banvägen S för FGROUП-axlarna betyder under polynominterpoleringen dock som regel en ej konstant ändring av kurvparametern U . För de axlar som inte ingår i FGROUП kan därför väljas mellan två möjligheter, hur dessa ska följa FGROUП-axlarna:

- Synkront till banvägen S (SPATH)
- Synkront till kurvparametern U (UPATH)

Syntax

SPATH
UPATH

Betydelse

SPATH:	De ej i FGROUП ingående axlarna flyttas relaterade till banvägen S
UPATH:	De ej i FGROUП ingående axlarna flyttas relaterade till kurvparametern U

Märk

UPATH och SPATH bestämmer också sambandet mellan F -ord-polynomet (FPOLY, FCUB, FLIN) och banrörelsen.

Randvillkor

SPATH resp. UPATH har ingen betydelse vid:

- Linjärinterpolering ($G1$)
- Cirkelinterpolering ($G2, G3$)
- Gängblock ($G33, G34, G35, G33x, G63$)
- Alla banaxlar ingår i FGROUП

Exempel

Det följande exemplet visar skillnaden mellan de båda typerna av rörelsestyrning.

Programkod

```
N10 FGROUП (X,Y,Z)
N15 G1 X0 A0 F1000 SPATH ; SPATH
N20 POLY PO[X]=(10,10) A10
```

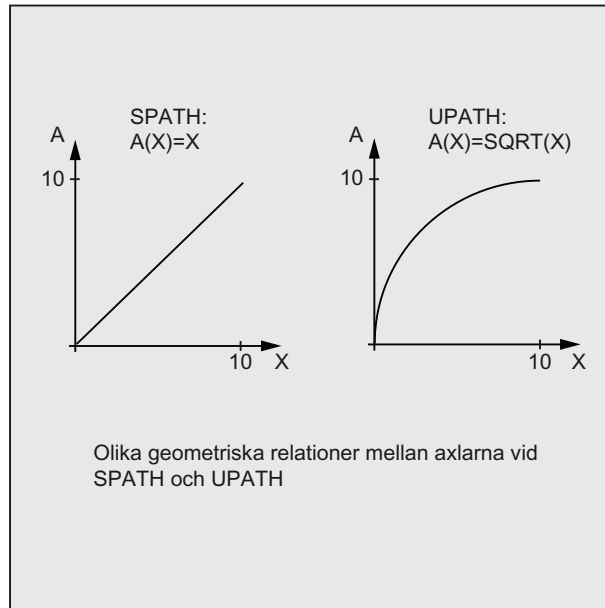
Programkod

```

N10 FGROUP(X,Y,Z)
N15 G1 X0 A0 F1000 UPATH          ; UPATH
N20 POLY PO[X]=(10,10) A10

```

I båda programavsnitten beror i N20 vägen S för FGROUP-axlarna på kvadraten för kurvparametern U. Därför resulterar längs vägen för X olika positioner för synkronaxeln A, beroende på om SPATH eller UPATH är aktiv.

**Ytterligare informationer****Styrningsbeteende vid Reset och maskin-/optionsdata**

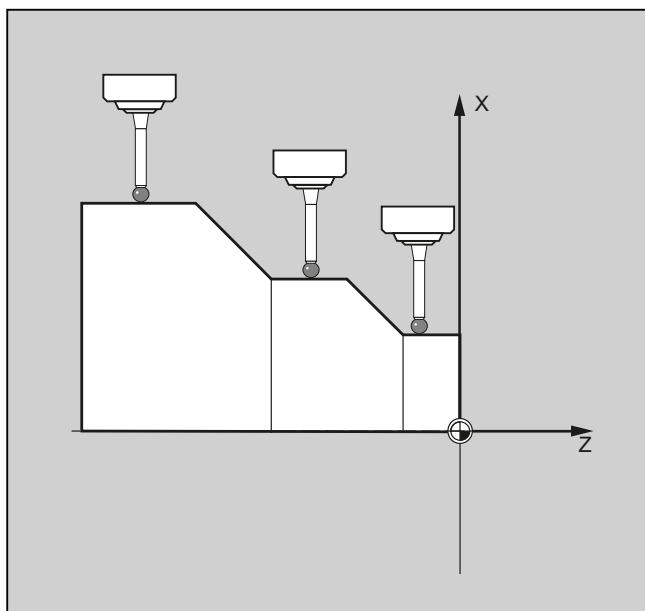
Efter Reset är det genom MD20150 \$MC_GCODE_RESET_VALUES[44] bestämda G-kommandot verksamt (45:e G-gruppen).

Grundinställningsvärdet för typen av översläpning fastläggs med MD20150 \$MC_GCODE_RESET_VALUES[9] (10:e . G-gruppen).

Det axelspecifika maskindatumet MD33100 \$MA_COMPRESS_POS_TOL[<n>] har en utvidgad betydelse: det innehåller toleranser för kompressorfunktionen och för översläpningen med G642.

5.7 Mäta med kopplande mätprobe (MEAS, MEAW)

Med funktionen "Mäta med kopplande mätprobe" uppsöks ärpositioner på arbetsstycket och vid mätprobens kopplingsflank mäts positionerna för alla i mätblocket programmerade axlar och skrivs för varje axel i den tillhörande minnescellen.



För programmeringen av funktionen står de båda följande fasta adresserna till förfogande:

- MEAS
Med MEAS raderas restvägen mellan är- och börposition.
- MEAW
MEAW används för mätuppgifter vid vilka i varje fall den programmerade positionen ska uppsökas.

MEAS och MEAW är blockvis verksamma och programmeras tillsammans med rörelseanvisningar. Matning och interpoleringstyp (G0, G1, ...), likaväl som antalet axlar måste därvid vara anpassade till respektive mätproblem.

Syntax

```
MEAS=<TE> G... X... Y... Z...
MEAW=<TE> G... X... Y... Z...
```

Betydelse

MEAS:	Kommando: Mäta med restvägsradering	
	Verkan:	blockvis
MEAW:	Kommando: Mäta utan restvägsradering	
	Verkan:	blockvis

<TE>:	Trigger-händelse för utlösning av mätning		
	Typ:	INT	
	Värdeområde:	-2, -1, 1, 2	
	Betydelse:		
	(+)1	stigande flank på mätprobe 1 (på mätingång 1)	
	-1	fallande flank på mätprobe 1 (på mätingång 1)	
	(+)2	stigande flank på mätprobe 2 (på mätingång 2)	
	-2	fallande flank på mätprobe 2 (på mätingång 2)	
	Observera: Det existerar maximalt 2 mätprober (beroende på utbyggnadsnivå).		
G . . . :	Interpoleringstyp, t.ex. G0, G1, G2 eller G3		
X . . . Y . . . Z . . . :	Slutpunkter i kartesiska koordinater		

Exempel

Programkod	Kommentar
N10 MEAS=1 G1 F1000 X100 Y730 Z40	; Mätblock med mätprobe till den första mätingången och linjärinterpolering.
...	Fördekoderingsstopp skapas automatiskt.

Ytterligare informationer

Mätuppdagsstatus

Är i programmet en utvärdering erforderlig om mätproben har kopplat eller inte kan tillståndsvariabeln \$AC_MEA[<n>] (<n> = nummer på mätproben) avrögas:

Värde	Betydelse
0	Mätuppdag inte fullfört
1	Mätuppdag avslutas med framgång (mätprobe har kopplat)

Märk

Blir mätproben inkopplad i programmet sätts variabeln på 1. Vid start av ett mätblock sätts variabeln automatiskt på mätprobens begynnelsestillstånd.

Mätvärdesupptagande

Positionerna för alla förflyttningsbara ban- och positioneringsaxlar i blocket (maximalt antal axlar beroende på styrningskonfiguration) registreras. Vid MEAS bromsas rörelsen definierat efter kopplingen av mätproben.

Märk

Är i ett mätblock en geometriaxel programmerad lagras mätvärdena för alla aktuella geometriaxlar.

Är i ett mätblock en i en transformation deltagande axel programmerad lagras mätvärdena för alla axlar som deltar i denna transformation.

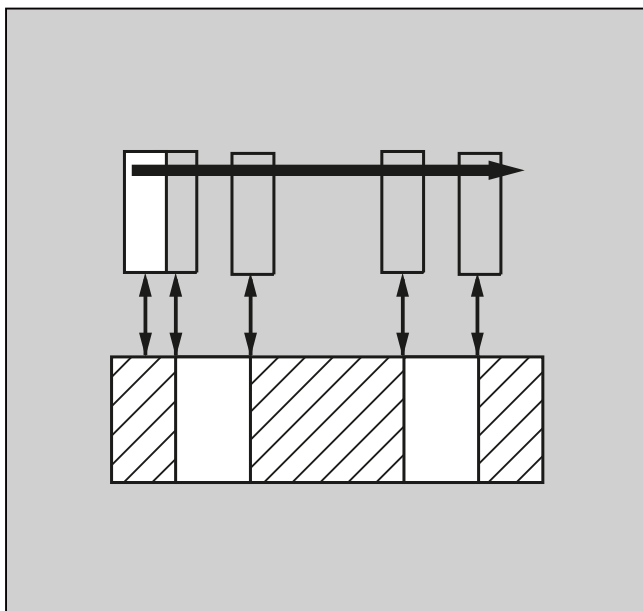
Läsa mätresultat

Mätresultaten för de med mätproben registrerade axlarna kan läsas via de följande systemvariablerna:

- \$AA_MM[<Achse>]
Mätresultat i maskinkoordinatsystemet
- \$AA_MW[<Achse>]
Mätresultat i arbetsstyckskoordinatsystemet

5.8 Axelspecifik mätning (MEASA, MEAWA, MEAC) (option)

Vid axelspecifik mätning kan flera mätprober och flera mätsystem användas.



För programmeringen av funktionen står nyckelorden `MEASA`, `MEAWA` och `MEAC` till förfogande.

Med `MEASA` resp. `MEAWA` registreras för den programmerade axeln upp till fyra mätvärden per mätning und lagras passande till trigger-händelsen i systemvariabler.

Kontinuerliga mätuppdrag kan genomföras med `MEAC`. I detta fall lagras mätresultaten i FIFO-variabler.

Syntax

```
MEASA [<Achse>] = (<Modus>, <TE1>, ..., <TE4>)
```

```
MEAWA [<Achse>] = (<Modus>, <TE1>, ..., <TE4>)
```

```
MEAC [<Achse>] = (<Modus>, <Messspeicher>, <TE1>, ..., <TE4>)
```

Märk

`MEASA` och `MEAWA` är blockvis verksamma och kan programmeras tillsammans i ett block. Programmeras däremot `MEASA/MEAWA` tillsammans med `MEAS/MEAW` i ett block leder det till ett felmeddelande.

Betydelse

MEASA:	Nyckelord: Axelspecifik mätning med restvägsradering	
	Verkan:	blockvis
MEAWA:	Nyckelord: Axelspecifik mätning utan restvägsradering	
	Verkan:	blockvis

MEAC:	Nyckelord: Axelspecifik kontinuerlig mätning utan restvägsradering	
	Verkan:	blockvis
<Achse>:	Namn på den till mätningen använda kanalaxeln	
<Modus>:	Tvåsiffrigt tal för angivande av driftmode (mätmode och mätsystem)	
	Endekad (mätmode):	
	0	Avbryta mätuppdrag.
	1	Upp till 4 olika samtidigt aktiverbara triggerhändelser.
	2	Upp till 4 efter varandra aktiverbara triggerhändelser.
	3	Upp till 4 efter varandra aktiverbara triggerhändelser dock ingen övervakning av triggerhändelse 1 vid Start (larm 21700/21703 undertrycks).
	Observera: Detta mode är inte möjligt vid MEAC.	
	Tiodekad (mätsystem):	
	0 (eller ingen uppgift)	aktivt mätsystem
	1	mätsystem 1
<TE>:	2	mätsystem 2
	3	båda mätsystemen
	Trigger-händelse för utlösning av mätning	
	Typ:	INT
	Värdeområde:	-2, -1, 1, 2
	Betydelse:	
	(+)1	stigande flank på mätprobe 1
	-1	fallande flank på mätprobe 1
<Messspeicher>:	(+)2	stigande flank på mätprobe 2
	-2	fallande flank på mätprobe 2
Nummer för FIFO (tempominne)		

Exempel

Exempel 1: Axelspecifik mätning med restvägsradering i mode 1 (utvärdering i kronologisk ordningsföljd)

a) med 1 mätsystem

Programkod	Kommentar
...	
N100 MEASA[X]=(1,1,-1) G01 X100 F100	; Mäta i mode 1 med aktivt mätsystem. Vänta på mätsignal med stigande/fallande flank på mätprobe 1 på förflyttningsvägen till X=100.
N110 IF \$AC_MEA[1]==FALSE GOTO ENDE	; Kontrollera framgång i mätningen.
N120 R10=\$AA_MM1[X]	; Spara till den första programmerade triggerhändelsen (stigande flank) hörande mätvärde.

5.8 Axelspecifik mätning (MEASA, MEAWA, MEAC) (option)

Programkod	Kommentar
N130 R11=\$AA_MM2[X]	; Spara till den andra programmerade triggerhändelsen (fallande flank) hörande mätvärde.
N140 ENDE:	

b) med 2 mätsystem

Programkod	Kommentar
...	
N200 MEASA[X]=(31,1,-1) G01 X100 F100	; Mäta i mode 1 med båda mätsystemen. Vänta på mätsignal med stigande/fallande flank på mätprobe 1 på förflyttningsvägen till X=100.
N210 IF \$AC_MEA[1]==FALSE GOTOF ENDE	; Kontrollera framgång i mätningen.
N220 R10=\$AA_MM1[X]	; Spara mätvärde från mätsystem 1 vid stigande flank.
N230 R11=\$AA_MM2[X]	; Spara mätvärde från mätsystem 2 vid stigande flank.
N240 R12=\$AA_MM3[X]	; Spara mätvärde från mätsystem 1 vid fallande flank.
N250 R13=\$AA_MM4[X]	; Spara mätvärde från mätsystem 2 vid fallande flank.
N260 ENDE:	

Exempel 2: Axelspecifik mätning med restvägsradering i mode 2 (utvärdering i programmerad ordningsföljd)

Programkod	Kommentar
...	
N100 MEASA[X]=(2,1,-1,2,-2) G01 X100 F100	; Mäta i mode 2 med aktivt mätsystem. Vänta på mätsignal i ordningsföljden stigande flank på mätprobe 1, fallande flank mätprobe 1, stigande flank på mätprobe 2, fallande flank mätprobe 2 på förflyttningsvägen till X=100.
N110 IF \$AC_MEA[1]==FALSE GOTOF MESSTASTER2	; Kontrollera framgång i mätningen med mätprobe 1.
N120 R10=\$AA_MM1[X]	; Spara till den första programmerade triggerhändelsen (stigande flank mätprobe 1) hörande mätvärde.
N130 R11=\$AA_MM2[X]	; Spara till den andra programmerade triggerhändelsen (stigande flank mätprobe 1) hörande mätvärde.
N140 MESSTASTER2:	
N150 IF \$AC_MEA[2]==FALSE GOTOF ENDE	; Kontrollera framgång i mätningen med mätprobe 2.

5.8 Axelspecifik mätning (MEASA, MEAWA, MEAC) (option)

Programkod	Kommentar
N160 R12=\$AA_MM3[X]	; Spara till den tredje programmerade triggerhändelsen (stigande flank mätprobe 2) hörande mätvärde.
N170 R13=\$AA_MM4[X]	; Spara till den fjärde programmerade triggerhändelsen (stigande flank mätprobe 2) hörande mätvärde.
N180 ENDE:	

Exempel 3: Axelspecifik kontinuerlig mätning i mode 1 (utvärdering i kronologisk ordningsföljd)**a) Mätning av upp till 100 mätvärden**

Programkod	Kommentar
...	
N110 DEF REAL MESSWERT[100]	
N120 DEF INT Schleife=0	
N130 MEAC[X]=(1,1,-1) G01 X1000 F100	; Mäta i mode 1 med aktivt mätsystem, spara mätvärdena under \$AC_FIFO1, vänta på mätsignal med fallande flank på mätprobe 1 på förflyttningsvägen till X=1000.
N135 STOPRE	
N140 MEAC[X]=(0)	; Avbryta mätning efter uppnådd axelposition.
N150 R1=\$AC_FIFO1[4]	; Spara antal utmatade mätvärden i parameter R1.
N160 FOR Schleife=0 TO R1-1	
N170 MESSWERT[Schleife]=\$AC_FIFO1[0]	; Läs ut mätvärden från \$AC_FIFO1 och spara.
N180 ENDFOR	

b) Mätning med restvägsradering efter 10 mätvärden

Programkod	Kommentar
...	
N10 WHEN \$AC_FIFO1[4]>=10 DO MEAC[x]=(0) DELDTG(x)	; Radera restväg
N20 MEAC[x]=(1,1,1,-1) G01 X100 F500	
N30 MEAC[X]=(0)	
N40 R1=\$AC_FIFO1[4]	; Antal mätvärden.
...	

c) Mäta en fallande/stigande kuggflank med 2 mätprober

Programkod	Kommentar
...	
N110 DEF REAL MESSWERT[16]	

5.8 Axelspecifik mätning (MEASA, MEAWA, MEAC) (option)

Programkod	Kommentar
N120 DEF INT Schleife=0	
N130 MEAC[X]=(1,1,-1,2) G01 X100 F100	; Mäta i mode 1 med aktivt mätsystem, spara mätvärdena under \$AC_FIFO1, vänta på mätsignal i ordningsföljden fallande flank på mätprobe 1, stigande flank mätprobe 2, på förflyttningsvägen till X=100.
N140 STOPRE	; Fördekoderingsstopp
N150 MEAC[X]=(0)	; Avbryta mätning efter uppnådd axelposition.
N160 R1=\$AC_FIFO1[4]	; Spara antal utmatade mätvärden i parameter R1.
N170 FOR Schleife=0 TO R1-1	
N180 MESSWERT[Schleife]=\$AC_FIFO1[0]	; Läs ut mätvärden från \$AC_FIFO1 och spara.
N190 ENDFOR	

Ytterligare informationer

Mätuppdrag

Programmeringen av ett mätuppdrag kan göras i detaljprogrammet eller utifrån en synkronaktion (se Kapitel "Synkronaktioner (Sida 597)"). Per axel kan därvid endast ett mätuppdrag vara aktivt vid en och samma tidpunkt.

Märk

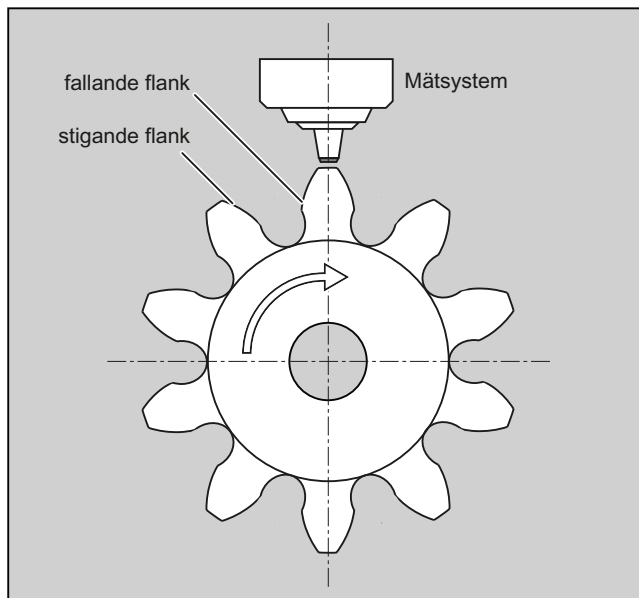
Matningen ska anpassas till respektive mätproblem.

Vid MEASA och MEAWA kan korrekta resultat garanteras endast vid matningar vid vilka inte mer än ett lika och inte mer än 4 olika triggerhändelser per lägesregleringstakt inträffar.

Vid kontinuerlig mätning med MEAC får förhållandet mellan interpolatortakt och lägesregleringstakt inte bli större än 1:8.

Trigger-händelse

En triggerhändelse är sammansatt av numret för mätproben och utlösningsskriteriet (stigande eller fallande flank) för mätsignalen.



För varje mätning kan var gång upp till 4 triggerhändelser bearbetas av de aktiverade mätproberna alltså upp till två mätprober med var och en två mätflanker. Ordningsföljden för bearbetningen samt det maximala antalet triggerhändelser är därvid beroende av det valda modet.

Märk

För mätmode 1 gäller: En lika triggerhändelse får programmeras endast en gång i ett mätuppdrag!

Vid MEAC kan antalet mätvärden per triggerhändelse ökas genom användning av PROFIBUS-telegram 395 på totalt 8 mätvärden vid stigande och 8 vid fallande flank per triggerhändelse och lägesregleringstakt.

- En mätprobe: 8 mätvärden vid stigande och 8 vid fallande flank
- Två mätprober: 4 mätvärden vid stigande och 4 vid fallande flank per mätprobe

Därigenom låter sig högre matningar och varvtal realiseras vid användning av PROFIBUS-telegram 395.

Litteratur:

Funktionshandbok Tilläggfunktioner; Mätning (M5), Kapitel: Axial mätning

Driftmode

Med den första siffran (tiodekad) i driftmodet väljs det önskade mätsystemet. Finns endast ett mätsystem, dock ett andra programmerat, används automatiskt det som finns.

Med den andra siffran (endekad) väljs det önskade mätmetod. Därmed anpassas mätförloppet till möjligheterna för respektive styrning:

- **Mode 1**
Utvärderingen av triggerhändelser görs i kronologisk ordningsföljd för uppträdandet. I detta mode kan vid användning av sexaxelkomponenter endast en triggerhändelse programmeras resp. det kopplas automatiskt om till mode 2 (utan meddelande) när flera triggerhändelser anges.
- **Mode 2**
Utvärderingen av triggerhändelserna görs i den programmerade ordningsföljden.
- **Mode 3**
Utvärderingen av triggerhändelserna görs i den programmerade ordningsföljden dock ingen övervakning av triggerhändelse 1 vid START.

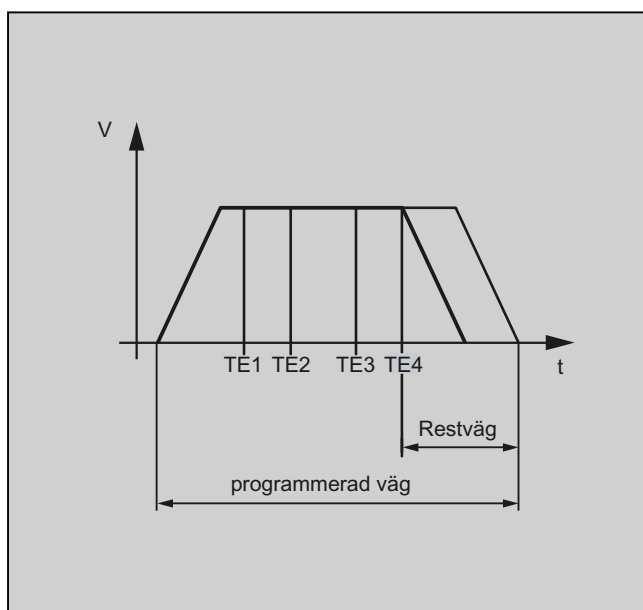
Märk

Vid användning av 2 mätsystem är endast två triggerhändelser programmerbara.

Mätning med och utan restvägsradering

Vid programmeringen av MEASA görs restvägsraderingen först efter registreringen av alla mätvärden som kräv.

För speciella mätuppgifter, vid vilka i varje fall den programmerade positionen ska uppsökas, används MEAWA.



Märk

MEASA är inte programmerbar i synkronaktioner. Ersättningsvis kan MEAWA plus restvägsradering programmeras som synkronaktion.

Startas mätuppdraget med MEAWA utifrån synkronaktionerna är mätvärdena disponibla endast i maskinkoordinatsystemet.

Märresultat för MEASA, MEAWA

Mätresultaten står under följande systemvariabler till förfogande:

- i maskinkoordinatsystemet:

\$AA_MM1 [<Achse>]	Mätvärde från det programmerade mätsystemet vid triggerhändelse 1
...	...
\$AA_MM4 [<Achse>]	Mätvärde från det programmerade mätsystemet vid triggerhändelse 4

- i arbetsstyckskoordinatsystemet:

\$AA_MW1 [<axel>]	Mätvärde från det programmerade mätsystemet vid triggerhändelse 1
...	...
\$AA_MW4 [<axel>]	Mätvärde från det programmerade mätsystemet vid triggerhändelse 4

Geometriaxlar / transformationer

Ska den axiala mätningen för en geometriaxel startas måste samma mätuppdrag programmeras explicit för alla resterande geometriaxlar. Det samma gäller för axlar som deltar i en transformation.

Exempel:

```
N10 MEASA[Z]=(1,1) MEASA[Y]=(1,1) MEASA[X]=(1,1) G0 Z100
```

eller

```
N10 MEASA[Z]=(1,1) POS[Z]=100
```

Mätuppdrag med 2 mätsystem

Görs ett mätuppdrag med två mätsystem registreras varje av de båda möjliga triggerhändelserna av båda mätsystemen för respektive axel. Beläggningen av de reserverade variablerna är därmed föreskriven:

\$AA_MM1 [<Achse>]	resp.	\$AA_MW1 [<Achse>]	Mätvärde från mätsystem 1 vid triggerhändelse 1
\$AA_MM2 [<Achse>]	resp.	\$AA_MW2 [<Achse>]	Mätvärde från mätsystem 2 vid triggerhändelse 1
\$AA_MM3 [<Achse>]	resp.	\$AA_MW3 [<Achse>]	Mätvärde från mätsystem 1 vid triggerhändelse 2
\$AA_MM4 [<Achse>]	resp.	\$AA_MW4 [<Achse>]	Mätvärde från mätsystem 2 vid triggerhändelse 2

Systemvariabler

Mätprobestatusen står under följande systemvariabler till förfogande:

\$A_PROBE[<n>]

Värde	Betydelse
1	Mätprobe inkopplad
0	Mätprobe inte inkopplad

Mätprobebegränsningen står under följande systemvariabler till förfogande:

\$A_PROBE_LIMITED[<n>]

Värde	Betydelse
1	Mätprobebegränsning aktiv
0	Mätprobebegränsning inaktiv

<n> = mätprobe

Litteratur:

Listhandbok Systemvariabler

Mätuppdagsstatus vid MEASA, MEAWA

Är en utvärdering erforderlig i programmet så kan mätuppdagsstatus avfrågas via \$AC_MEA[<n>], med <n> = numret på mätproben. Så snart alla i ett block programmerade triggerhändelser för mätproben har skett, levererar denna variabel värdet 1. Annars är värdet 0.

Märk

Starta mätning från synkronaktioner aktualiseras inte \$AC_MEA mer. I detta fall ska den nya PLC-gränssnittssignalen DB31, ... DBX62.3 resp. den likvärdiga variabeln \$AA_MEAAC[<Achse>] avfrågas.

Betydelse:

\$AA_MEAAC==1: mätning aktiv

\$AA_MEAAC==0: mätning inte aktiv

Kontinuerlig mätning (MEAC)

Mätvärdena föreligger vid MEAC i maskinkoordinatsystemet och lagras i det angivna FIFO[n]-minnet (tempominne). Är två mätprober projekterade för mätningen lagras mätvärdena från den andra mätproben åtskilt i det extra för detta projekterade (vid MD inställbart) FIFO[n+1]-minnet.

FIFO-minnet är ett tempominne, i vilket mätvärdena förs in i \$AC_FIFO-variablerna enligt omsättningsprincipen (se Kapitel "Synkronaktioner (Sida 597)").

Märk

FIFO-innehållet kan endast läsas ut en gång från tempominnet. För flerfaldig användning av mässdata måste dessa mellanlagras i användardata.

Överskrider antalet mätvärden för FIFO-minnet det i maskindatum fastlagda högsta antalet så avslutas mätningen automatiskt.

Ändlös mätning låter sig realiserats genom cykliska utläsning av mätvärden. Utläsningen måste därvid göras minst med samma frekvens som ingången av nya mätvärden.

Litteratur:

- Funktionshandbok Synkronaktioner; Utförlig beskrivning, Kapitel: Parameter (\$AC_FIFO)
- Funktionshandbok Tilläggsfunktioner; Mätning (M5), Kapitel: Axial mätning

Skydd mot felprogrammeringar

Följande felprogrammeringar identifieras och indikeras som ett fel:

- MEASA/MEAWA programmerade tillsammans med MEAS/MEAW i ett block
Exempel:
N01 MEAS=1 MEASA[X]=(1,1) G01 F100 POS[X]=100
- MEASA/MEAWA med parameterantal <2 eller >5
Exempel:
N01 MEAWA[X]=(1) G01 F100 POS[X]=100
- MEASA/MEAWA med triggerhändelse skild från 1/ -1/ 2/ -2
Exempel:
N01 MEASA[B]=(1,1,3) B100
- MEASA/MEAWA med felaktig mode
Exempel:
N01 MEAWA[B]=(4,1) B100
- MEASA/MEAWA med dubbelt programmerad triggerhändelse
Exempel:
N01 MEASA[B]=(1,1,-1,2,-1) B100
- MEASA/MEAWA och geometriaxel som saknas
Exempel:
N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) G01 X50 Y50 Z50 F100 ;GEO-Achse X/Y/Z
- Ej enhetligt mätuppsdrag vid geometriaxlar
Exempel:
N01 MEASA[X]=(1,1) MEASA[Y]=(1,1) MEASA[Z]=(1,1,2) G01 X50 Y50 Z50 F100

5.9 Speciella funktioner för OEM-användaren (OMA1 ... OMA5, OEMIPO1, OEMIPO2, G810 ... G829)

OEM-adresser

Betydelsen hos OEM-adresser bestämmer OEM-användaren. Funktionaliteten inbringas via Compile-cykler. 5 OEM-adresser är reserverade (OMA1 ... OMA5). Adressbeteckningarna kan ställas in. OEM-adresser är tillåtna i varje block.

Reserverade kommandoanrop

För OEM-användaren är följande G-kommandoanrop reserverade:

- OEMIPO1, OEMIPO2 (från G-gruppen 1)
- G810 ... G819 (G-gruppen 31)
- G820 ... G829 (G-gruppen 32)

Funktionaliteten inbringas via Compile-cykler.

Funktioner och underprogram

Dessutom kan OEM-användare också lägga till fördefinierade funktioner och underprogram med parameteröverföring.

Märk

Arbetsstyckssimulering

Till SW 4.4 stöds inga, från SW 4.4 endast utsökta Compile-cykler (CC) vid arbetsstyckssimuleringen.

Språkkommandon i detaljprogrammet från ej understödda Compile-cykler (OMA1 ... OMA5, OEMIPO1/2, G810 ... G829, egna procedurer och funktioner) leder därför utan individuell behandling till larmmeddelande och till avbrott av simuleringen.

Lösning: Behandla de CC-specifika språkelement som saknas i detaljprogrammet individuellt (\$P_SIM-avfrågning).

Exempel:

```
N1 G01 X200 F500
IF (1==$P_SIM)
N5 X300 ;bei Simulation CC nicht aktiv
ELSE
N5 X300 OMA1=10
ENDIF
```

5.10 Matningsreducering med hörnfördröjning (FENDNORM, G62, G621)

Vid den automatiska hörnfördröjningen sänks matningen klockformigt kort före det aktuella hörnet. Dessutom kan arbetsmättet för det för bearbetningen relevanta verktygsbeteendet ställas in med parametrar via settingdata. Dessa är:

- Början och slut på matningsreduceringen
- Override, med vilken matningen reduceras
- Identifikation av det relevanta hörnet

Som relevanta hörn tas det hänsyn till de hörn, vars innervinkel är mindre än det via settingdatum parameterinställda hörnet.

Med defaultvärdet `FENDNORM` kopplas funktionen för den automatiska hörnoverriden från.

Litteratur:

/FBFA/ Funktionsbeskrivning ISO-dialekter

Syntax

`FENDNORM`

`G62 G41`

`G621`

Betydelse

<code>FENDNORM:</code>	Automatisk hörnfördröjning från
<code>G62:</code>	Hörnfördröjning vid innerhörn vid aktiv verktygsradiekompensering
<code>G621:</code>	Hörnfördröjning vid alla hörn vid aktiv verktygsradiekompensering

G62 verkar endat vid innerhörn med

- aktiv verktygsradiekompensering `G41`, `G42` och
- aktiv banstyrningsdrift `G64`, `G641`

Med nedsänkt matning körs fram till det motsvarande hörnet som resulterar ur:

$F * (\text{override till matningsreducering}) * \text{matningsoverride}$

Den maximalt möjliga matningsnedsänkningen uppnås precis när verktyget, relaterat till medelpunktsbanan, skall göra riktningsväxel vid det motsvarande hörnet.

G621 verkar analogt till G62 vid varje hörn, för de av `FGROUP` fastlagda axlarna.

5.11 Programmerbart rörelseslutkriterium (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)

Liknande blockväxelkriteriet vid baninterpolering (G601, G602 und G603) kan rörelseslutkriteriet vid enkelaxelinterpolering i ett detaljprogram resp. i synkronaktioner för kommando-/PLC-axlar programmeras.

Beroende på vilket rörelseslutkriterium som är inställt avslutas detaljprogramblock resp. teknologicykelblock med enkelaxelrörelser olika snabbt. Samma gäller för PLC via FC15/16/18.

Syntax

```

FINEA [<Achse>]
COARSEA [<Achse>]
IPOENDA [<Achse>]
IPOBRKA (<Achse> [, <Zeitpunkt>])
ADISPOSA (<Achse> [, <Modus>, <Fenstergröße>])

```

Betydelse

FINEA:	Rörelseslutkriterium: "Fint precisionsstopp"	
	Verkan:	modal
COARSEA:	Rörelseslutkriterium: "Grovt precisionsstopp"	
	Verkan:	modal
IPOENDA:	Rörelseslutkriterium: "Interpolator-stopp"	
	Verkan:	modal
IPOBRKA:	Blockväxelkriterium: Bromsramp	
	Verkan:	modal
ADISPOSA:	Toleransfönster för rörelseslutkriterium	
	Verkan:	modal
<Achse>:	Kanalaxelnamn (X, Y, ...)	
<Tidpunkt>:	Tidpunkten för blockväxlingen relaterad till bromsrampen i %:	
	100% = Början på bromsrampen 0% = Slut på bromsrampen, betyder samma som IPOENDA	
	Typ:	REAL
<Modus>:	Referens för toleransfönstret	
	Värdeområde:	0 Toleransfönster inte aktivt
		1 Toleransfönster beträffande börposition
		2 Toleransfönster beträffande ärposition
	Typ:	INT
<Fenstergröße>:	Storlek på toleransfönstret	
	Typ:	REAL

Exempel

Exempel 1: Rörelseslutkriterium: "Interpolator-stopp"

Programkod
<pre>; Kör positioneringsaxel X till 100, hastighet 200 m/ min, acceleration 90%, ; Rörelseslutkriterium: Interpolator-stopp N110 G01 POS[X]=100 FA[X]=200 ACC[X]=90 IPOENDA[X] ; Synkronaktion: ; ALLTID NÄR: Ingång 1 är inställd ; DÅ Kör positioneringsaxel X till 50, hastighet 200 m/ min, acceleration 140%, ; Rörelseslutkriterium: Interpolator-stopp N120 EVERY \$A_IN[1] DO POS[X]=50 FA[X]=200 ACC[X]=140 IPOENDA[X]</pre>

Exempel 2: Blockväxelkriterium: "Bromsramp"

Programkod	Kommentar
	; Defaultinställning verksam
N40 POS[X]=100	; Positioneringsrörelse från X till position 100 Blockväxelkriterium: Fint precisionsstopp
N20 IPOBRKA(X,100)	; Blockbyteskriterium "Bromsramp", 100% = Början på bromsrampen
N30 POS[X]=200	; Blockväxel följer så snart axel X börjar bromsa
N40 POS[X]=250	; Axel X bromsar inte vidare på position 200, utan kör vidare till position 250. Så snart som axeln börjar bromsa sker blockväxeln.
N50 POS[X]=0	; Axel X bromsar och kör tillbaka på position 0 Blockväxling vid position 0 och "Fint precisionsstopp"
N60 X10 F100	; Axel X kör som banaxel till position 10

Ytterligare informationer

Systemvariabel för rörelseslutkriterium

Det verksamma rörelseslutkriteriet kan läsas via systemvariabeln \$AA_MOTEND.

Litteratur: /LIS2sl/ Lifthandbok, bok 2

Blockväxelkriterium: "Bromsramp" (IPOBRKA)

Har vid aktiveringen av blockväxelkriteriet "Bromsramp" ett värde programmerats för den optionala blockväxeltidpunkten, blir detta verksamt för nästa positioneringsrörelse och skrivs huvudkörningssynkront i settingdatumet. Har inget värde angivits för blockväxeltidpunkten, blir det aktuella värdet i settingdatumet verksamt.

5.11 Programmerbart rörelseslutkriterium (FINEA, COARSEA, IPOENDA, IPOBRKA, ADISPOSA)

SD43600 \$SA_IPOBRAKE_BLOCK_EXCHANGE

Med nästa programmering av ett axialt rörelseslutkriterium (FINEA, COARSEA, IPOENDA) inaktiveras IPOBRKA för den motsvarande axeln.

Extra blockväxelkriterium: "Toleransfönster" (ADISPOSA)

Med ADISPOSA kan ett toleransfönster definieras runt blockslutpunkten (valfritt är- eller börposition) som extra blockväxelkriterium. För blockväxlingen måste då båda villkoren vara uppfyllda:

- Blockväxelkriterium: "Bromsramp"
- Blockväxelkriterium: "Toleransfönster"

Litteratur

Ytterligare informationer över blockväxelkriterium för positioneringsaxlar se:

- Funktionshandbok Tilläggsfunktioner; Positioneringsaxlar (P2)
- Programmeringshandbok Grunder; Kapitel "Matningsreglering"

Koordinattransformationer (frames)

6.1 Koordinattransformation via framevariabel

Förutom de i programmeringshandbok Grunder, Kapitel "Koordinattransformationer (Frames)" beskrivna kommandona som t.ex. `ROT`, `AROT`, `SCALE`, etc. kan arbetsstyckskoordinatsystemet (WKS) också transformeras via Framevariablerna `$P_...FR` (datahållningsframes) och `$P_...FRAME` (aktiva Frames).

Den följande bilden ger en överblick över struktureringen av Framevariablerna:

- Datahållningsframes
- Aktiva frames
- Aktiv totalframe: Sammanlänkning av alla aktiva frames
- NCU-globala frames
- Kanalspecifika frames

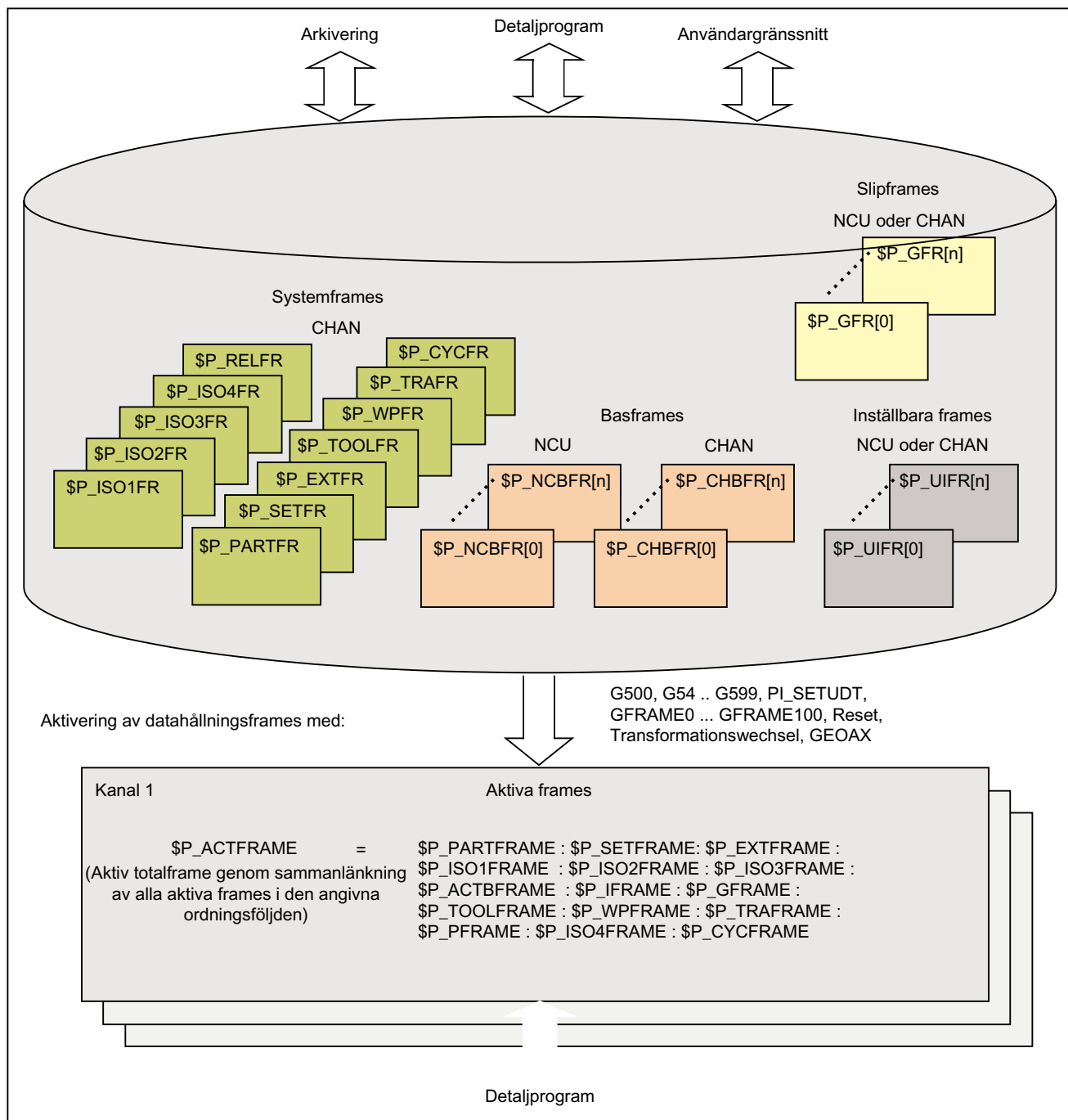


Bild 6-1 Översikt av Framevariablerna

6.1.1 Fördefinierad framevariabel (\$P_CHBFRAME, \$P_IFRAME, \$P_PFRAME, \$P_ACTFRAME)

Aktiv: Kanalspecifika basframes \$P_CHBFRAME[<n>] (\$P_BFRAME)

Märk

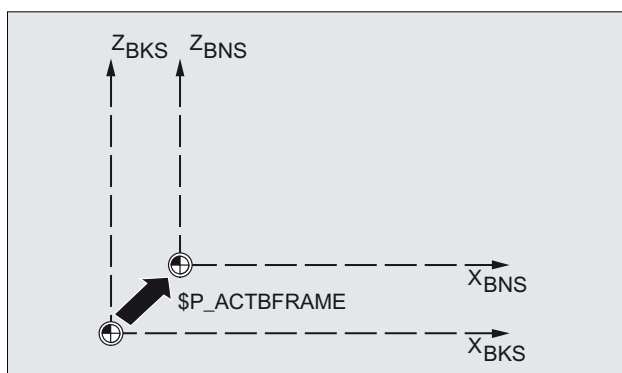
De för aktuell basframe \$P_BFRAME och datahållnings-basframe \$P_UBFR bibehålls av kompatibilitetsskäl.

- $\$P_BFRAME \sqsubseteq \$P_CHBFRAME[0]$
- $\$P_UBFR \sqsubseteq \$P_CHBFR[0]$.

Framevariablerna \$P_CHBFRAME[<n>] definierar relationen mellan baskoordinatsystemet (BKS) och basnollpunktsystemet (BNS).

Ska den aktuella kanalspecifika basframen \$P_CHBFRAME[<n>] genast bli verksam i NC-programmet, står följande möjligheter till förfogande

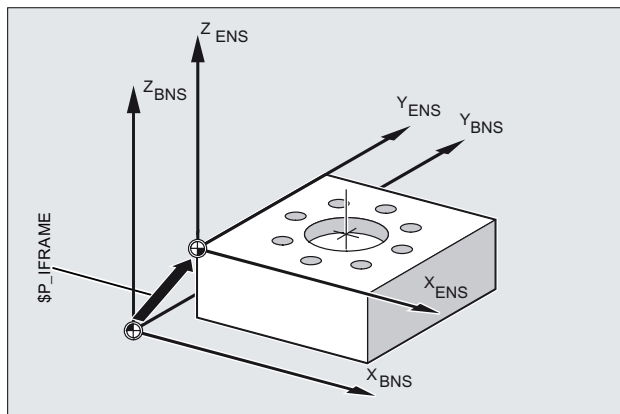
- Kommandon:
 - G500 (Koppla från alla inställbara frames, basframes förblir aktiva)
 - G54 ... G599 (inställbara nollpunktsförflyttningar)
- Tillordning av en kanalspecifik basframe för datahållningen till en aktuell kanalspecifik basframe:
 $\$P_CHBFRAME[<n>] = \$P_CHBFR[<m>]$



Aktiv: kanalspecifik inställbar frame \$P_IFRAME

Framevariabeln \$P_IFRAME, definierar relationen mellan basnollpunktsystemet (BNS) och det inställbara nollpunktsystemet (ENS).

- $\$P_IFRAME$ motsvarar $\$P_UIFR[\$P_IFRNUM]$
- $\$P_IFRAME$ innehåller efter programmering av t.ex. G54 den genom G54 definierade translationen, rotationen, skalningen eller speglingen.

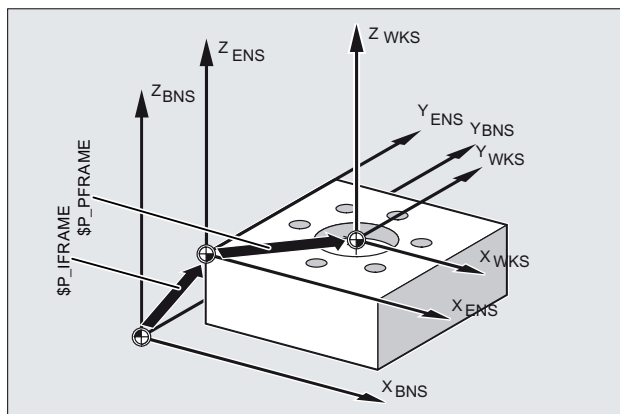


Aktiv: kanalspecifik programmerbar frame \$P_PFRAME

Framevariabeln \$P_PFRAME definierar relationen mellan det inställbara nollpunktsystemet (ENS) och arbetsstyckskoordinatsystemet (WKS).

\$P_PFRAME innehåller den resulterande ramen som resulterar

- ur programmeringen av TRANS/ATRANS, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR resp.
- ur tillordningen av CTRANS, CROT, CMIRROR, CSCALE i den programmerbara FRAME



Aktiv: Totalframe \$P_ACTFRAME

Den i kanalen verksamma totalramen, resulterar ur sammanlänknigen av alla i kanalen verksamma frames.

```
$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 beskriver nollpunkten för det aktuella arbetsstyckskoordinatsystemet.

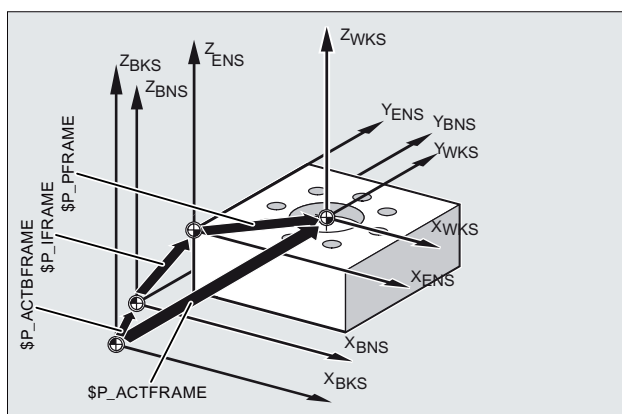
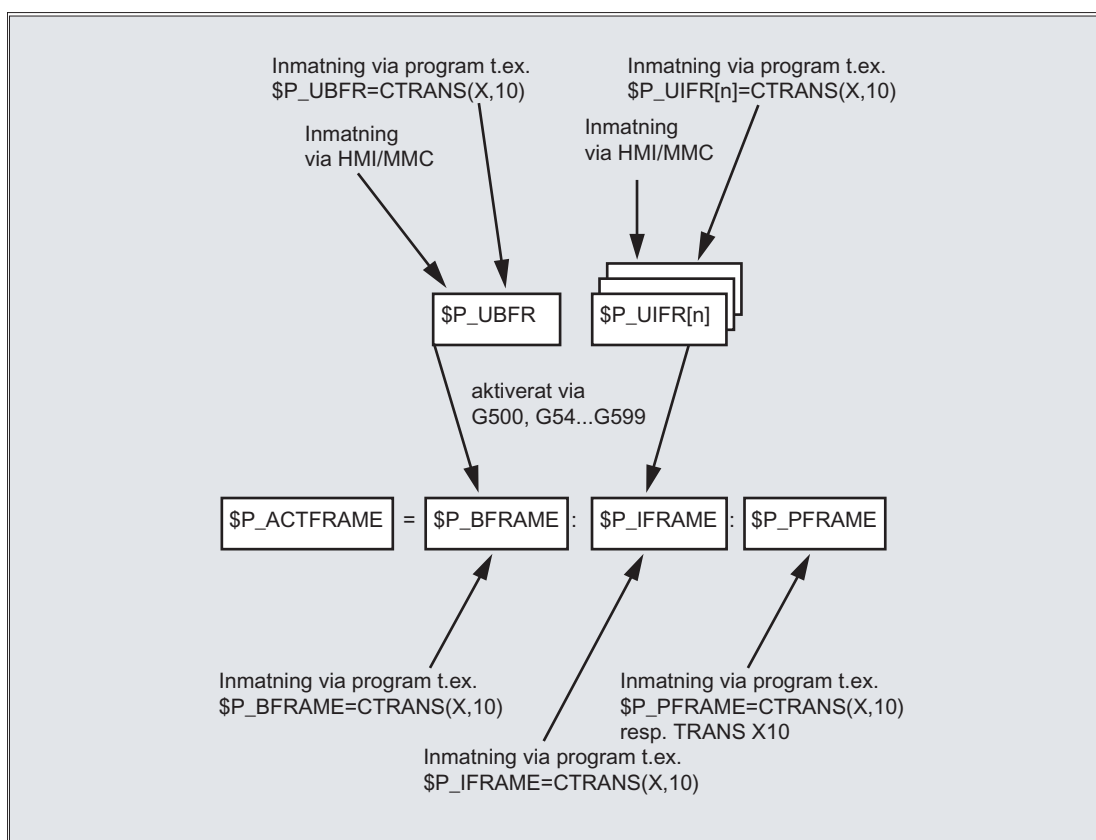


Bild 6-2 Framevariabel \$P_ACTFRAME

Förändras en av de följande frames \$P_BFRAME / \$P_CHBFRAME[<n>], \$P_IFRAME eller \$P_PFRAME beräknas den aktuella totalframe \$P_ACTFRAME på nytt.



Basframe och inställbar frame verkar efter Reset, när MD 20110 RESET_MODE_MASK är inställt på följande sätt:

Bit0=1, Bit14=1 --> \$P_UBFR (basframe) verkar

Bit0=1, Bit5=1 --> \$P_UIFR[\$P_UIFRNUM] (inst. frame) verkar

Datahållning: Kanalspecifika basframes \$P_CHBFR[<n>]

Via framevariabeln \$P_CHBFR[<n>] läses / skrivs basframes i datahållningen.

Datahållningsramen blir genom skrivningen inte genast aktiv i kanalen. Aktiveringen av den skrivna ramen sker vid:

- Kanal-Reset och MD20110 \$MC_RESET_MODE_MASK, Bit0 == 1 och Bit14 == 1
- Kommando G500, G54 ... G57, G505 ... G599 (till/frånkoppling av basframes med påföljande nyberäkning av den aktuella totalramen)

Datahållning: Kanalspecifika inställbara frames \$P_UIFR[<n>]

Via framevariablerna \$P_UIFR[<n>] läses / skrivs de inställbara frames i datahållningen.

Framen blir genom skrivningen inte genast aktiv i kanalen. Inberäkningen av den skrivna ramen i kanalen sker vid:

- Kommando G500 (frånkoppling av alla inställbara frames resp. nollpunktsförflyttningar)
- Kommando G54 ... G57, G505 ... G599 (tillkoppling av en inställbar frame resp. nollpunktsförflyttning)

Aktiv inställbar frame	Datahållningsframe	(motsvarar kommando)
\$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

6.2 Värdetillordningar till frames

6.2.1 Tillordna direkta värden (axelvärde, vinkel, skala)

I NC-programmet kan du direkt belägga frames eller framevariabler med värden.

Syntax

Syntax

```
$P_PFRAME = CTRANS(X, <Verschiebungswert>, Y, <Verschiebungswert>, Z, <Verschiebungswert>, ...)
```

```
$P_PFRAME = ROT(X, <Winkel>, Y, <Winkel>, Z, <Winkel>, ...)
```

```
$P_UIFR[..] = CROT(X, <Winkel>, Y, <Winkel>, Z, <Winkel>, ...)
```

```
$P_PFRAME = CSCALE(X, <Maßstab>, Y, <Maßstab>, Z, <Maßstab>, ...)
```

```
$P_PFRAME = CMIRROR(X, Y, Z)
```

Syntaxen för \$P_CHBFRAME[<n>] är identisk med \$P_PFRAME.

Betydelse

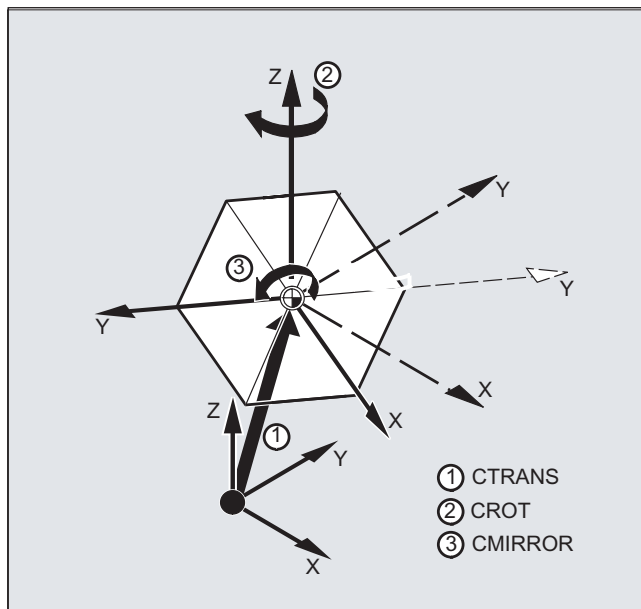
CTTRANS:	Förflyttning i de angivna axlarna
CROT:	Rotation runt de angivna axlarna
CSCALE:	Skalförändring i de angivna axlarna
CMIRROR:	Riktningsbyte för den angivna axeln
X, Y, Z:	Förflyttningsvärde i den angivna geometriaxelns riktning
<Verschiebungswert>:	Förflyttningsvärde
<Winkel>:	Vinkel runt vilken vrids
<Maßstab>:	Skaluppgift

Exempel

Värdetillordningar till framekomponenter för den aktuella programmerbara ramen

Värdetillordningen till framekomponenterna translation, rotation och spegling för den aktuella programmerbara ramen.

```
$P_PFRAME = CTRANS(X,10,Y,20,Z,5) : CROT(Z,45) : CMIRROR(Y)
```



Skriva rotationskomponenter för en frame

Tillordning av värden till alla tre axlarna i vridkomponenten till den inställbara datahållningsramen \$P_UIFR med CROT :

```
$P_UIFR[5] = CROT(X, 0, Y, 0, Z, 0)
```

Alternativt därtill den direkta tillordningen av enskilda värden direkt till respektive axel i vridkomponenten till datahållningsramen:

```
$P_UIFR[5, Y, RT]=0
$P_UIFR[5, X, RT]=0
$P_UIFR[5, Z, RT]=0
```

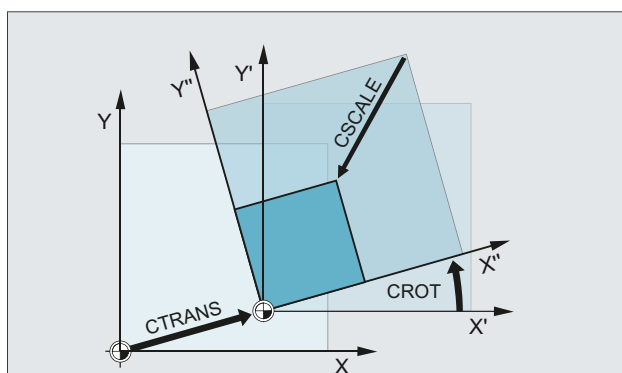
Beskrivning

Flera operationer på en frame kan genom länkningsoperatören : förbindas med varandra. Operationerna utförs sedan efter varandra från vänster till höger.

Exempel

Sammanlänkade operationer på \$P_PFRAME med förflyttning, rotation och skalning:

```
$P_PFRAME = CTRANS(...) : CROT(...) : CSCALE...
```



6.2.2 Läs och förändra framekomponenter (TR, FI, RT, SC, MI)

Du har möjlighet av åtkomst till **enskilda** data i en frame, t.ex. till ett bestämt förflyttningsvärde eller rotationsvinkel. Dessa värden kan du förändra eller tillordna en annan variabel.

Syntax

`R10=$P_UIFR[$P_UIFNUM,X,RT]`

Rotationsvinkeln RT runt X-axeln från den aktuellt giltiga nollpunktsförflyttningen \$P_UIFRNUM ska tillordnas variabeln R10.

`R12=$P_UIFR[25,Z,TR]`

Förflyttningsvärdet TR i Z från datablocket för den inställda ramen nr 25 ska tillordnas variabeln R12.

`R15=$P_PFRAME[Y,TR]`

Förflyttningsvärdet TR i Y för de aktuella programmerbara ramen ska tillordnas variabeln R15.

`$P_PFRAME[X,TR]=25`

Förflyttningsvärdet TR i X för den aktuella programmerbara ramen ska förändras. X25 gäller genast.

Betydelse

\$P_UIFRNUM:	Med denna variabel upprättas automatiskt relationen till den aktuellt giltiga inställbara nollpunktsförflyttningen.
P_UIFR[n,...,...]:	Genom angivande av framenumret n har du åtkomst till den inställbara ramen nr n.
	Angivande av den komponent som ska läsas eller förändras:
TR:	TR Translation
FI:	FI Translation Fine
RT:	RT Rotation
SC:	SC Scale skalförändring
MI:	MI Spegling
X, Y, Z:	Dessutom (se exempel) anges den motsvarande axeln X, Y, Z.

Värdeområde för rotationen RT

Vridning runt 1:a geometriaxel:	-180° till +180°
Vridning runt 2:a geometriaxel:	-90° till +90°
Vridning runt 3:e geometriaxel:	-180° till +180°

Beskrivning

Anropa frame

Genom angivande av systemvariabeln `$P_UIFRNUM` har du direkt åtkomst till den med `$P_UIFR` resp. G54, G55, ... aktuellt inställda nollpunktsförflyttningen (`$P_UIFRNUM` innehåller numret för den aktuellt inställda ramen).

Alla andra sparade inställbara frames `$P_UIFR` anropar du genom angivande av det motsvarande numret `$P_UIFR[n]`.

För fördefinierad framevariabel och egendefinierade frames anger du namnet t.ex. `$P_IFRAME`.

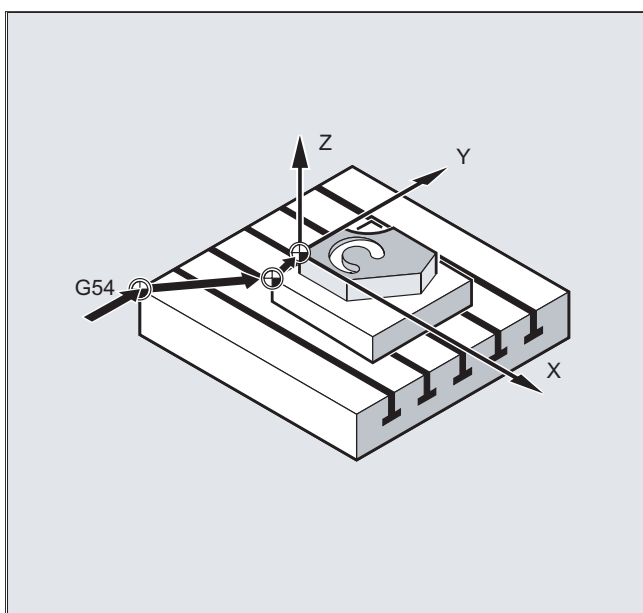
Anropa data

Inom hakparentesen står axelnamn och framekomponent för det värde till vilket du önskar åtkomst eller att förändra, t.ex. `[X, RT]` eller `[Z, MI]`.

6.2.3 Räkna med frames

I NC-programmet kan en frame tillordnas en annan frame eller frames förbindas med varandra.

Framelänkar lämpar sig t.ex. för beskrivningen av flera arbetsstycken som är anordnade på en palett och som ska bearbetas i ett tillverkningsförlopp.



För beskrivningen av palettuppgifter skulle framekomponenterna t.ex. kunna innehålla endast vissa detaljvärden, genom vars sammanlänkning olika arbetsstycksnollpunkter genereras.

Exempel

Tillordningar

Programkod	Kommentar
DEF FRAME EINSTELLUNG_1	; Definition av en lokal framevariabel
EINSTELLUNG_1 = CTRANS(X,10)	; Tillordning av resultatet för en funktion på framevariabeln
\$P_PFRAME = EINSTELLUNG_1	; Tillordning av framevariablerna till den aktuella ramen
DEF FRAME EINSTELLUNG_4	; Definition av en lokal framevariabel
EINSTELLUNG_4 = \$P_PFRAME	; Mellanlagring av den aktuella ramen i framevariablerna
...	
\$P_PFRAME = EINSTELLUNG_4	; Returläsning av den aktuella ramen från framevariablerna

Sammanlänknings

Frames blir av operatoren : i den programmerade ordningsföljden förbundna med varandra. framekomponenterna som t.ex. förflyttningar, rotationer osv. utförs additivt efter varandra.

Programkod	Kommentar
\$P_IFRAME = \$P_UIFR[15] : \$P_UIFR[16]	; Tillordning av resultat-ramen från sammanlänknings- ningen av de ; båda inställbara datahållningsframes med den ak- tiva ; inställbara total-ramen. Användningsexempel: ; \$P_UIFR[15]: Förflyttning ; \$P_UIFR[16]: Vridning
\$P_UIFR[3] = \$P_UIFR[4] : \$P_UIFR[5]	; Tillordning av resultat-ramen från sammanlänknings- ningen av de ; båda inställbara datahållningsframes med den ak- tiva ; andra inställbara datahållningsramen

6.2.4 Definition av framevariabler (DEF FRAME)

Förutom de fördefinierade framevariablerna kan också egna framevariabler definieras. De självdefinierade framevariablerna är användarvariabler av typ FRAME. Namnet på ramen kan inom ramen för användarvariablernas regler tilldelas fritt.

Med funktionerna CTRANS, CROT, CSCALE, CMIRROR kan de självdefinierade framevariablerna tilldelas värden.

Syntax

```
DEF FRAME <Name>
```

Betydelse

DEF FRAME:	Definiera användarvariabler av typ FRAME.
<Namn>:	Namn för framevariabler

Exempel

Definition av en framevariabel "PALETTE" och tillordning av förflyttnings- och rotationsvärden:

Programkod	Kommentar
DEF FRAME PALETTE	; Definiera framevariabeln PALETTE
PALETTE = CTRANS(...) : CROT(...)	; Tillordning av resultatframes från sammanlänknings- ningen av ; förskjutning och rotation till framevariabeln PALETTE

6.3 Grov- och finförflyttning (CTrans, CFine)

Finförflyttning

En finförflyttning `CFINE ( . . . )` kan användas på följande frames:

- Inställbara frames: `$P_UIFR` resp. `$P_IFRAME`
- Basframes: `$P_NCBFR[<n>]`, `$P_CHBFR[<n>]` resp. `$P_CHBFRAMES[<n>]` eller `$P_ACTBFRAME`
- Programmerbar frame: `$P_PFRAME`

Finförflyttningen av en frame programmeras med kommandot `CFINE ( . . . )`.

Grovförflyttning

Grovförflyttningen `CTrans ( . . . )` kan användas på alla frames.

Total förskjutning

Den totala förskjutningen erhålls genom addition av grov- och finförflyttning.

Maskindata

Frigivning av finförflyttningen

Finförflyttningen frigges med maskindatum:

MD18600 \$MN_MM_FRAME_FINE_TRANS = 1

Syntax

Finförflyttning

- Totalframe
 - `<Frame> = CFINE (<K_1>, <Wert>)`
 - `<Frame> = CFINE (<K_1>, <Wert>, <K_2>, <Wert>)`
 - `<Frame> = CFINE (<K_1>, <Wert>, <K_2>, <Wert>, <K_3>, <Wert>)`
- Frame-komponent
 - `<Frame>[<n>, <K_1>, FI] = <Wert>`

Grovförflyttning

- Totalframe
 - `<Frame> = CTRANS (<K_1>, <Wert>)`
 - `<Frame> = CTRANS (<K_1>, <Wert>, <K_2>, <Wert>)`
 - `<Frame> = CTRANS (<K_1>, <Wert>, <K_2>, <Wert>, <K_3>, <Wert>)`
- Frame-komponent
 - `<Frame>[<n>, <K_1>, TR] = <Wert>`

6.3 Grov- och finförflyttning (CTrans, CFine)

Speciell för den programmerbara ramen \$P_PFRAME:

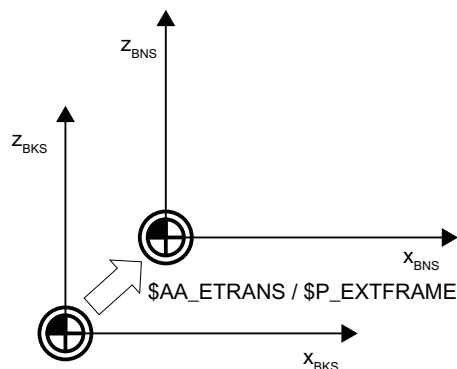
- TRANS <K_1> <Wert>
- TRANS <K_1> <Wert> <K_2> <Wert>
- TRANS <K_1> <Wert> <K_2> <Wert> <K_3> <Wert>

Betydelse

<Frame>:	Frame t.ex. Inställbar frame för datahållningen \$P_UIFR[<n>]
CFine:	Finförflyttning, additiv förflyttning.
CTrans:	Grovförflyttning, absolut förflyttning
TRANS:	Endast programmerbar frame: Grovförflyttning, absolut förflyttning
<K_n>:	Koordinataxlar X, Y, Z
<Wert>:	Förflyttningsvärde

6.4 Extern nollpunktsförflyttning (\$AA_ETRANS)

Den externa nollpunktsförflyttningen är en linjär förflyttning mellan baskoordinatsystemet (BKS) och basnollpunktsystemet (BNS).



Den externa nollpunktsförflyttningen med hjälp av \$AA_ETRANS verkar, beroende på maskindata-parametreringen, på två sätt:

1. Systemvariabeln \$AA_ETRANS verkar efter aktiveringen genom NC/PLC-gränssnittssignalen direkt som förflyttningsvärde
2. Värdet för systemvariabeln \$AA_ETRANS övertas efter aktiveringen genom NC/PLC-gränssnittssignalen i de aktiva systemframes \$P:EXTFRAME och datahållningsramen \$P_EXTFR. Sedan beräknas den aktiva totalramen \$P_ACTFRAME på nytt.

Maskindata

I samband med systemvariablerna \$AA_ETRANS ska skiljas mellan två tillvägagångssätt, som väljs via det följande maskindatamet:

MD28082 \$MC_MM_SYSTEM_FRAME_MASK, Bit1 = <Wert>

<Wert>	Betydelse
0	Funktion: Direkt skrivande av \$AA_ETRANS[<Achse>] av PLC, HMI eller NC-program. Frigivning för utkörning av nollpunktsförflyttningen för \$AA_ETRANS[<Achse>] i nästa möjliga förflyttningsblock: DB31, ... DBX3.0
1	Funktion: Aktivering av den aktiva systemramen \$P:EXTFRAME och datahållningsramen \$P_EXTFR Frigivning för utkörning av nollpunktsförflyttningen för \$AA_ETRANS[<Achse>] av: DB31, ... DBX3.0. Därpå följer i kanalen: • Stopp för alla förflyttningsrörelser i kanalen (utom kommando- och PLC-axlar) • Fördekoderingsstopp med anslutande reorganisering (STOPRE) • Grovförflyttning aktiv frame \$P_EXTFRAME[<Achse>] = \$AA_ETRANS[<Achse>] • Grovförflyttning datahållningsframe \$P_EXTFR[<Achse>] = \$AA_ETRANS[<Achse>] • Ny beräkning av den aktiva totalramen \$P_ACTFRAME • Utkörning av förflyttningen i de programmerade axlarna. • Fortsättning av den stoppade förflyttningsrörelsen resp. NC-programmet

Programmering

- Syntax
\$AA_ETRANS[<Achse>] = <Wert>
- Betydelse

\$AA_ETRANS:	Systemvariabel för mellanlagring av den externa nollpunktsförflyttningen
<Achse>:	Kanalaxel
<Wert>:	Förflytningsvärde

NC/PLC-gränssnittssignal

DB31, ... DBX3.0 = 0 → 1 ⇒ \$P_EXTFRAME[<Achse>] = \$P_EXTFR[<Achse>] = \$AA_ETRANS[<Achse>]

6.5 Ärvärdesinställning med förlust av referensstatus (PRESETON)

Proceduren `PRESETON()` sätter för en eller flera axlar ett nytt ärvärde i maskinkoordinatsystemet (MKS). Detta motsvarar en nollpunktsförflyttning i MKS för axeln. Axeln förflyttas därigenom inte.

Med `PRESETON` utlöses ett fördekoderingsstopp med synkronisering. Ärpositionen tilldelas axeln först när den står stilla.

Är axeln inte tillordnad kanalen vid `PRESETON` är det fortsatta tillvägagångssättet beroende av den axelspecifika projekteringen av axelbytesbeteendet:

MD30552 \$MA_AUTO_GET_TYPE

Referensstatus

Med inställningen av ett nytt ärvärde i maskinkoordinatsystemet nollställs referensstatusen för maskinaxeln:

DB31, ... DBX60.4 / .5 = 0 (referenskört / synkroniserat mätsystem 1 / 2)

Det rekommenderas att använda `PRESETON` endast vid axlar utan referenspunktsplikt.

För att återupprätta det ursprungliga maskinkoordinatsystemet måste mätsystemet till maskinaxeln, t.ex. med referenspunktkörning från detaljprogrammet (`G74`) på nytt referensköras.

 SE UPP
Förlust av referensstatus
Med inställningen av ett nytt ärvärde i maskinkoordinatsystemet med <code>PRESETON</code> återställs referensstatusen för maskinaxeln till "ej referenskörd / synkroniserad".

Programmering

Syntax

`PRESETON(<Achse_1>, <Wert_1> [, <Achse_2>, <Wert_2>, ... <Achse_8>, <Wert_8>])`

Betydelse

PRESETON:	Ärvärdesinställning med förlust av referensstatus	
	Fördekoderingsstopp:	ja
	Ensam i blocket:	ja
<Achse_x>:	Maskinaxelnamn	
	Typ:	AXIS
	Värdeområde:	alla i kanalen definierade maskinaxelnamn
<Wert_x>:	Nytt ärvärde för maskinaxeln i maskinkoordinatsystemet (MKS) Inmatningen görs i det aktuellt giltiga måttssystemet (inch / metriskt) Det tas hänsyn till en aktiv diameterprogrammering (<code>DIAMON</code>)	
	Typ:	REAL

Litteratur

PRESETONS i NC-program

En utförlig beskrivning av PRESETON i NC-program finns i:

Funktionshandbok Grundfunktioner; kapitel "K2: Axlar, koordinatsystem, frames" > "Koordinatsystem" > "Maskinkoordinatsystem (MKS)" > "Ärvärdesinställning med förlust av referensstatus (PRESETON)"

PRESETONS i synkronaktioner

En utförlig beskrivning av PRESETON i synkronaktioner finns i:

Funktionshandbok Synkronaktioner; Kapitel: "Utförlig beskrivning" > "Aktioner i synkronaktioner" > "Ärvärdesinställning med förlust av referensstatus (PRESETON)"

6.6 Ärvärdesinställning utan förlust av referensstatus (PRESETON)

Proceduren `PRESETONS ( )` sätter för en eller flera axlar ett nytt ärvärde i maskinkoordinatsystemet (MKS). Detta motsvarar en nollpunktsförflyttning i MKS för axeln. Axeln förflyttas därigenom inte.

Med `PRESETONS` utlöses ett fördekoderingsstopp med synkronisering. Ärpositionen tilldelas axeln först när den står stilla.

Är axeln inte tillordnad kanalen vid `PRESETONS` är det fortsatta tillvägagångssättet beroende av den axelspecifika projekteringen av axelbytesbeteendet:

MD30552 \$MA_AUTO_GET_TYPE

Referensstatus

Med inställningen av ett nytt ärvärde i maskinkoordinatsystemet (MKS) med `PRESETONS` förändras **inte** referensstatusen för maskinaxeln.

Förutsättningar

- **Givartyp**

`PRESETONS` är möjligt endast vid följande givartyper för det aktiva mätsystemet:

- MD30240 \$MA_ENC_TYPE[<Messsystem>] = 0 (simulerad givare)
- MD30240 \$MA_ENC_TYPE[<Messsystem>] = 1 (råsignalgivare)

- **Referenskörar-mode**

`PRESETONS` är möjligt endast vid följande referenskörar-modes för det aktiva mätsystemet:

- MD34200 \$MA_ENC_REFP_MODE[<Messsystem>] = 0 (ingen referenspunktkörning möjlig)
- MD34200 \$MA_ENC_REFP_MODE[<Messsystem>] = 1 (referenskörning av inkrementella, roterande eller linjära mätsystem: Nollimpuls på givarspåret)

Programmering

Syntax

`PRESETONS (<Achse_1>, <Wert_1> [, <Achse_2>, <Wert_2>, ... <Achse_8>, <Wert_8>])`

Betydelse

PRESETONS:	Ärvärdesinställning utan förlust av referensstatus	
	Fördekoderingsstopp:	ja
	Ensam i blocket:	ja
<Achse_x>:	Maskinaxelnamn	
	Typ:	AXIS
	Värdeområde:	alla i kanalen definierade maskinaxelnamn
<Wert_x>:	Nytt aktuellt ärvärde för maskinaxeln i maskinkoordinatsystemet (MKS)	
	Inmatningen görs i det aktiva måttssystemet (inch / metriskt)	
	Det tas hänsyn till en aktiv diameterprogrammering (DIAMON)	
	Typ:	REAL

Litteratur

PRESETONS i NC-program

En utförlig beskrivning av PRESETONS i NC-program finns i:

Funktionshandbok Grundfunktioner; kapitel "K2: Axlar, koordinatsystem, frames" > "Koordinatsystem" > "Maskinkoordinatsystem (MKS)" > "Ärvärdesinställning utan förlust av referensstatus (PRESETONS)"

PRESETONS i synkronaktioner

En utförlig beskrivning av PRESETONS i synkronaktioner finns i:

Funktionshandbok Synkronaktioner; Kapitel: "Utförlig beskrivning" > "Aktioner i synkronaktioner" > "Ärvärdesinställning utan förlust av referensstatus (PRESETONS)"

6.7 Frame-beräkning ur 3 mätpunkter i rymden (MEAFRAME)

Funktionen MEAFRAME används som stöd för mätcyklerna. Den beräknar framen ur tre ideala och de tillhörande uppmätta punkterna.

Positioneras ett arbetsstycke för bearbetningen är dess position relativ till det kartesiska maskinkoordinatsystemet beträffande sin idealposition i allmänhet både förflyttad och även roterad. För exakt bearbetning eller mätning är antingen en dyrbar fysikalisk justering eller ändring av rörelserna i detaljprogrammet nödvändigt.

En frame kan fastläggas genom avsökning av tre punkter i rymden vars idealpositioner är kända. Avsökningen görs med en berörings- eller optisk sensor, som vidrör speciella på bärarplattan precisa fixerade hål eller mätkulor.

Syntax

```
MEAFRAME (<Ideal-Punkte>, <Messpunkte>, <Qualität>)
```

Betydelse

MEAFRAME:	Funktionsanrop		
<Ideal-Punkte>:	2-dim. REAL-fält som innehåller de tre koordinaterna för idealpunkterna		
<Messpunkte>:	2-dim. REAL-fält som innehåller de tre koordinaterna för de uppmätta punkterna		
<Qualität>:	Variabel med vilken informationer över kvaliteten för FRAME-beräkningen returneras		
	Typ:	VAR REAL	
	Värde:	-1	De ideala punkterna ligger nästan på en rät linje: Framen kunde inte beräknas. Den returnerade FRAME-variabeln innehåller en neutral frame.
		-2	Mätpunkterna ligger nästan på en rät linje: Framen kunde inte beräknas. Den returnerade FRAME-variabeln innehåller en neutral frame.
		-4	Beräkningen av rotationsmatrix slår fel av en annan orsak.
		≥ 0.0	Summan av distorsionerna (avstånd mellan punkterna) som behövs för överföring av den uppmätta triangeln till en som är kongruent med den ideala triangeln.

Märk

Mätningens kvalitet

För att de uppmätta ska kunna tillordnas de ideala koordinaterna med en kombinerad rotation/translation måste den av mätpunkterna bildade triangeln vara kongruent med den ideala triangeln. Detta blir verkställt av en kompensationsalgorithm, som minimerar summan av kvadraterna på avvikelserna, vilken överför den uppmätta till den ideala triangeln.

Den effektivt nödvändiga distorsionen av mätpunkterna kan tjäna som indikator för kvaliteten på mätningen och utges därför som extra variabel till MEAFRAME.

Märk

Den av MEAFRAME skapade framen kan genom funktionen ADDFRAME transformeras till en annan frame i frame-kedjan (se exempel "Länkning med ADDFRAME").

Exempel**Exempel 1:****Detaljprogram 1:****Programkod**

```
...
DEF FRAME CORR_FRAME
```

Inställning av mätpunkter:**Programkod****Kommentar**

```
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]=                ; För Test.
SET(10.1,0.2,-0.2,-0.2,10.2,0.1,-0.2,0.2,9.8)

DEF REAL FIT_QUALITY=0
DEF REAL ROT_FRAME_LIMIT=5                ; Tillåter max. 5 grader förvrid-
                                           ning av detaljpositionen.
DEF REAL FIT_QUALITY_LIMIT=3              ; Tillåter max. 3 mm förflyttning
                                           mellan
                                           den ideala och den uppmätta triang-
                                           eln.

DEF REAL SHOW_MCS_POS1[3]
DEF REAL SHOW_MCS_POS2[3]
DEF REAL SHOW_MCS_POS3[3]
```

Programkod**Kommentar**

```
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)
GOTO NO_FRAME
ENDIF
```

6.7 Frame-beräkning ur 3 mätpunkter i rymden (MEAFRAME)

Programkod	Kommentar
N240 IF FIT_QUALITY > FIT_QUALITY_LIMIT	
SETAL(65010)	
GOTOF NO_FRAME	
ENDIF	
N250 IF CORR_FRAME[X,RT] > ROT_FRAME_LIMIT	; Begränsning av den 1:a RPY-vinkeln.
SETAL(65020)	
GOTOF NO_FRAME	
ENDIF	
N260 IF CORR_FRAME[Y,RT] > ROT_FRAME_LIMIT	; Begränsning av den 2:a RPY-vinkeln.
SETAL(65021)	
GOTOF NO_FRAME	
ENDIF	
N270 IF CORR_FRAME[Z,RT] > ROT_FRAME_LIMIT	; Begränsning av den 3:e RPY-vinkeln.
SETAL(65022)	
GOTOF NO_FRAME	
ENDIF	
N300 \$P_IFRAME=CORR_FRAME	; Aktivera avsökningsframe med en inställningsbar frame. ; Kontrollera frame genom positionering av geometriaxlarna på de ideala punkterna.
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	; Inaktivera inställbar frame, eftersom förbelagd med nollframe (inget värde infört).
No_FRAME	; Inaktivera inställbar frame, eftersom förbelagd med nollframe (inget värde infört).
M0	
M30	

Exempel 2: Sammanlänkning av frames

Länkning av MEAFRAME för korrigeringar

Funktionen MEAFRAME levererar en korrigeringsframe. Sammanlänkas denna korrigeringsframe med den inställbara framen \$P_UIFR[1], som var aktiv vid anropet av funktionen (t.ex. G54), så erhåller man en inställbar frame för ytterligare omräkningar för förflyttning eller bearbetning.

Länkning med ADDFRAME

Ska denna korrigeringsframe i framelänken verka på ett annat ställe eller är före den inställbara framen också andra frames aktiva då kan funktionen ADDFRAME användas för inlänkning i en av kanal-basframes eller en systemframe.

I frames får härvid ej vara aktiva:

- Spegling med MIRROR
- Skalning med SCALE

Ingångsparametrarna för bör- och ärvärden är arbetsstyckskoordinaterna. I grundsystemet till styrningen ska dessa koordinater alltid anges metriskt eller i inch (G71/G70) och som radierelaterat (DIAMOF) mått.

Litteratur:

Ytterligare informationer till ADDFRAME se:

Funktionshandbok Grundfunktioner; K2: Axlär, koordinatsystem, frames

6.8 NCU-globala frames

NCU-globala frames finns det per NCU endast en gång för alla kanaler. NCU-globala frames kan skrivas och läsas utifrån alla kanaler. Aktiveringen av NCU-globala frames sker i respektive kanal.

Genom globala frames kan **kanalaxlar och maskinaxlar** skalas och speglas med förflyttningar.

Geometriskt sammanhang och framelänkar

Vid globala frames existerar inget geometriskt sammanhang mellan axlarna. Därför kan inga rotationer och ingen programmering av geometri-axelbeteckningar göras.

- På globala frames låter sig inga rotationer användas. Programmeringen av en rotation blir med larmet: "18310 Kanal %1 Block %2 Frame: Rotation otillåten", avvisat.
- Sammanlänkningen av globala frames och kanalspecifika frames är möjlig. Den resulterande framen innehåller alla frameandelar inklusive rotationen för alla axlar. Tillordningen av en frame med rotationsandelar till en global frame blir med larmet "Frame: Rotation otillåten", avvisat.

NCU-globala frames

NCU-globala basframes \$P_NCBFR[n]

Upp till 8 NCU-globala basframes kan projekteras:

Samtidigt kan det finnas kanalspecifika basframes.

Globala frames kan skrivas och läsas utifrån alla kanaler i en NCU. Vid skrivning av globala frames ska användaren sörja för en kanalkoordinering. Detta kan realiseras t. ex. med Wait-märken (WAITMC).

Maskintillverkare

Antalet globala basframes projekteras via maskindaten.

Litteratur:

Funktionshandbok Grundfunktioner; Axlar, koordinatsystem, frames (K2)

NCU-globala inställbara frames \$P_UIFR[n]

Alla inställbara frames G500, G54...G599 kan projekteras antingen NCU-globalt eller kanalspecifikt.

Maskintillverkare

Alla inställbara frames kan med hjälp av maskindatumet MD18601 \$MN_MM_NUM_GLOBAL_USER_FRAMES omprojekteras till globala frames.

Som axelbeteckningar vid frame-programkommandon kan kanalaxelbeteckningar och maskinaxelbeteckningar användas. Programmeringen av geometriaxelbeteckningar avvisas med ett larm.

6.8.1 Kanalspecifika frames (\$P_CHBFR, \$P_UBFR)

Inställbara frames eller basframes kan via detaljprogrammet och via BTSS skrivas och läsas av manövreringen och av PLC.

Finförflyttningen är möjlig också för globala frames. Undertryckningen av globala frames görs lika som för kanalspecifika frames med G53, G153, SUPA och G500.

Maskintillverkare

Via maskindatumet MD28081 \$MC_MM_NUM_BASE_FRAMES kan antalet basframes projekteras i kanalen. Standardkonfigurationen är så konstruerad att det finns minst en basframe per kanal. Maximalt är 8 basframes per kanal möjliga. Förutom de 8 basframes i kanalen kan det också finnas 8 NCU-globala basframes.

Kanalspecifika frames

\$P_CHBFR[n]

Via systemvariabeln \$P_CHBFR[n] kan basframes läsas och skrivas. Vid skrivningen av en basframe aktiveras inte den sammanlänkade totala basframen utan aktiveringen sker först med utförandet av en G500, G54... G599-anvisning. Variabeln tjänar företrädesvis som minne för skrivförlopp på basframen av HMI eller PLC. Dessa frame-variabler sparas via datasäkringen.

Första basframe i kanalen

Ett skrivande på den fördefinierade variabeln \$P_UBFR aktiverar basframen med fältindex 0 inte samtidigt utan aktiveringen sker först med utförandet av en G500, G54... G599-anvisning. Variabeln kan också skrivas och läsas i programmet.

\$P_UBFR

\$P_UBFR är identisk med \$P_CHBFR[0]. Standardmässigt finns det alltid en basframe i kanalen så att systemvariabeln är kompatibel med äldre versioner. Finns det ingen kanalspecifik basframe blir vid skrivning eller läsning larmet "Frame: Anvisning otillåten" utmatad.

6.8.2 I kanalen verksamma frames

I kanalen verksamma frames matas in av detaljprogrammet via de beträffande systemvariablerna för dessa frames. Härtill hör också systemframes. Via dessa systemvariabler kan den aktuella systemframen läsas och skrivas i detaljprogrammet.

Aktuella i kanalen verksamma frames

Översikt

Aktuella systemframes

\$P_PARTFRAME

\$P_SETFRAME

\$P_EXTFRAME

för:

TCARR och PAROT

Ärvärdesinställning och kantsökning

Extern nollpunktsförflyttning

\$P_NCBFRAME[n]	Aktuella NCU-globala basframes
\$P_CHBFRAME[n]	Aktuella kanal-basframes
\$P_BFRAME	Aktuell 1:a basframe i kanalen
\$P_ACTBFRAME	Total-basframe
\$P_CHBFRMASK och \$P_NCBFRMASK	Total-basframe
\$P_IFFRAME	Aktuell inställbar frame
Aktuella systemframes	för:
\$P_TOOLFRAME	TOROT och TOFRAME
\$P_WPFRAME	Arbetsstycksreferenspunkter
\$P_TRAFRAME	Transformationer
\$P_PFRAME	Aktuell programmerbar frame
Aktuell systemframe	för:
\$P_CYCFRAME	Cykler
P_ACTFRAME	Aktuell totalframe
FRAME-länkning	Aktuell frame är sammansatt av den totala-basframen

\$P_NCBFRAME[n] Aktuella NCU-globala basframes

Via systemvariabeln **\$P_NCBFRAME[n]** kan de aktuella globala basframe-fältelementen läsas och skrivas. Den resulterande total-basframen räknas genom skrivförloppet in i kanalen.

Den ändrade framen blir aktiv endast i den kanal i vilken framen programmerades. Ska framen ändras för alla kanaler i en NCU måste samtidigt **\$P_NCBFR[n]** och **\$P_NCBFRAME[n]** skrivas. De andra kanalerna måste sedan fortfarande aktivera framen med t.ex. G54. Vid skrivandet av en basframe beräknas den totala basframen på nytt.

\$P_CHBFRAME[n] Aktuella kanal-basframes

Via systemvariabeln **\$P_CHBFRAME[n]** kan de aktuella kanal-basframe-fältelementen läsas och skrivas. Den resulterande total-basframen räknas genom skrivförloppet in i kanalen. Vid skrivandet av en basframe beräknas den totala basframen på nytt.

\$P_BFRAME Aktuell 1:a basframe i kanalen

Via den fördefinierade framevariabeln **\$P_BFRAME** kan den aktuella basframen med fältindex 0, som är giltig i kanalen, läsas och skrivas i detaljprogrammet. Den skrivna basframen räknas genast in.

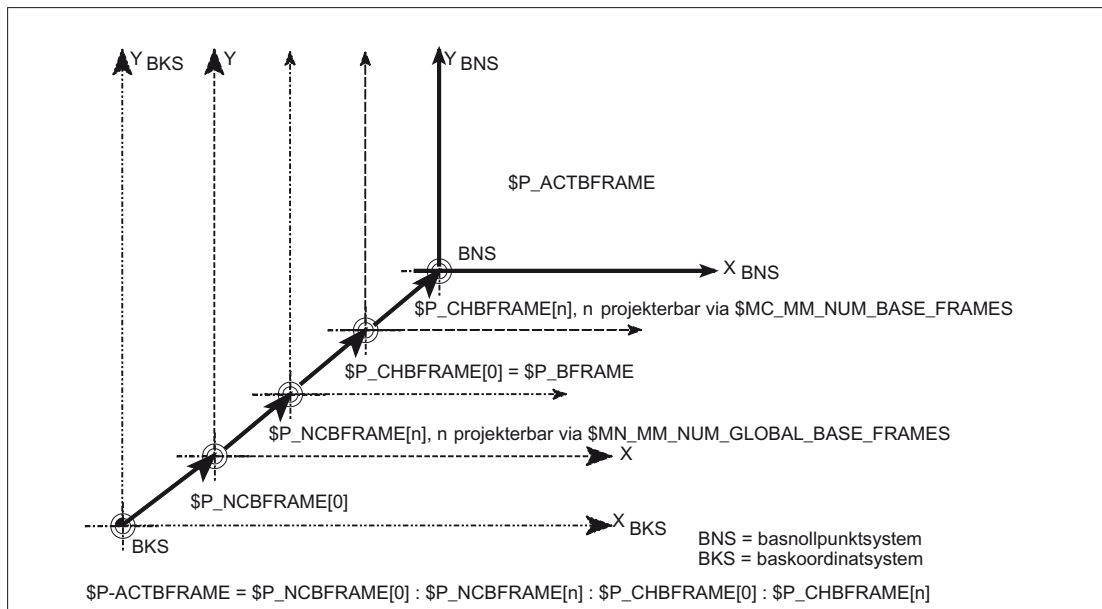
\$P_BFRAME är identisk med **\$P_CHBFRAME[0]**. Systemvariabeln har standardmässigt alltid ett giltigt värde. Finns det ingen kanalspecifik basframe blir vid skrivning eller läsning larmet "Frame: Anvisning otillåten" utmatad.

\$P_ACTBFRAME Total-basframe

Variabeln **\$P_ACTFRAME** registrerar den sammanlänkade total-basframen. Variabeln kan nu läsas.

\$P_ACTFRAME motsvarar:

\$P_NCBFRAME[0] : ... : \$P_NCBFRAME[n] : \$P_CHBFRAME[0] : ... : \$P_CHBFRAME[n].



\$P_CHBFRMASK och \$P_NCBFRMASK Total bas-frame

Via systemvariabeln **\$P_CHBFRMASK** och **\$P_NCBFRMASK** kan användaren välja vilka basframes han vill integrera i beräkningen av den "totala"-basframen. Variablerna kan programmeras endast i programmet och läsas via BTSS. Värdet för variablerna interpreteras som bitmask och anger vilket basframe-fältelement från **\$P_ACTFRAME** som flyter in i beräkningen.

Med **\$P_CHBFRMASK** kan föreskrivas vilka kanalspecifika basframes och med **\$P_NCBFRMASK** vilka NCU-globala basframes som räknas in.

Med programmeringen av variablerna beräknas den totala basframen och den totala framen på nytt. Efter Reset och i grundinställningen är värdet av **\$P_CHBFRMASK** och **\$P_NCBFRMASK** enligt följande:

\$P_CHBFRMASK = **\$MC_CHBFRAME_RESET_MASK**

\$P_NCBFRMASK = **\$MC_CHBFRAME_RESET_MASK**

Exempel:

```
$P_NCBFRMASK = 'H81' ;$P_NCBFRAME[0] : $P_NCBFRAME[7]
```

```
$P_CHBFRMASK = 'H11' ;$P_CHBFRAME[0] : $P_CHBFRAME[4]
```

\$P_IFRAME Aktuell inställbar frame

Via den fördefinierade framevariabeln **\$P_IFRAME** kan den aktuella inställbara framen vilken är giltig i kanalen, läsas och skrivs i detaljprogrammet. Den skrivna inställbara framen räknas genast in.

Vid NCU-globala inställbara frames verkar den ändrade framen endast i den kanal i vilken framen programmerades. Ska framen ändras för alla kanaler i en NCU måste samtidigt **\$P_UIFR[n]** und **\$P_IFRAME** skrivas. De andra kanalerna måste sedan fortfarande aktivera den motsvarande framen med t.ex. G54.

\$P_PFRAME Aktuell programmerbar frame

\$P_PFRAME är den programmerbara ramen som resulterar ur programmeringen av TRANS/ATRANS, G58/G59, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR resp. ur tillordningen av CTRANS, CROT, CMIRROR, CSCALE till den programmerbara FRAMEn.

Aktuell, programmerbar framevariabel som upprättar relationen mellan det inställbara nollpunktsystemet (ENS) och arbetsstyckskoordinatsystemet (WKS).

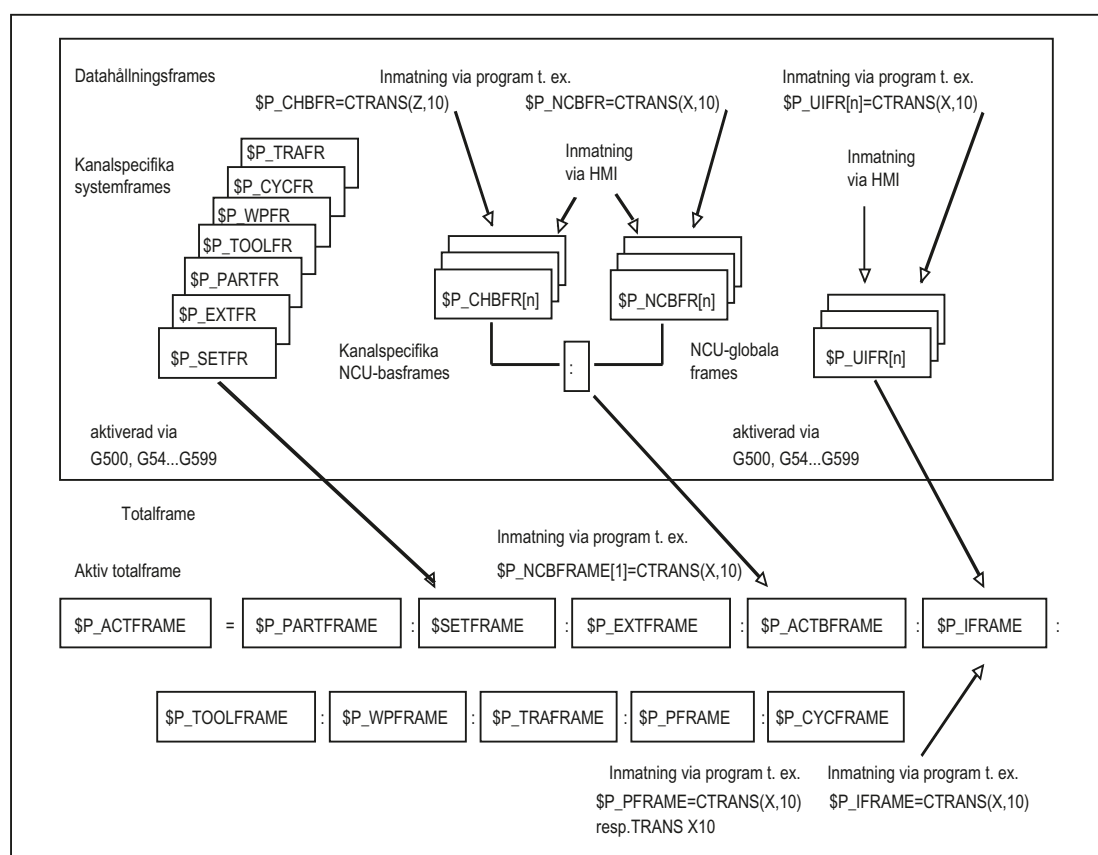
P_ACTFRAME Aktuell totalframe

Den aktuella resulterande totalramen \$P_ACTFRAME följer nu som sammanlänkning av alla basframes, den aktuella inställbara ramen och den programmerbara ramen. Den aktuella ramen aktualiseras alltid när en frameandel ändrar sig.

\$P_ACTFRAME motsvarar:

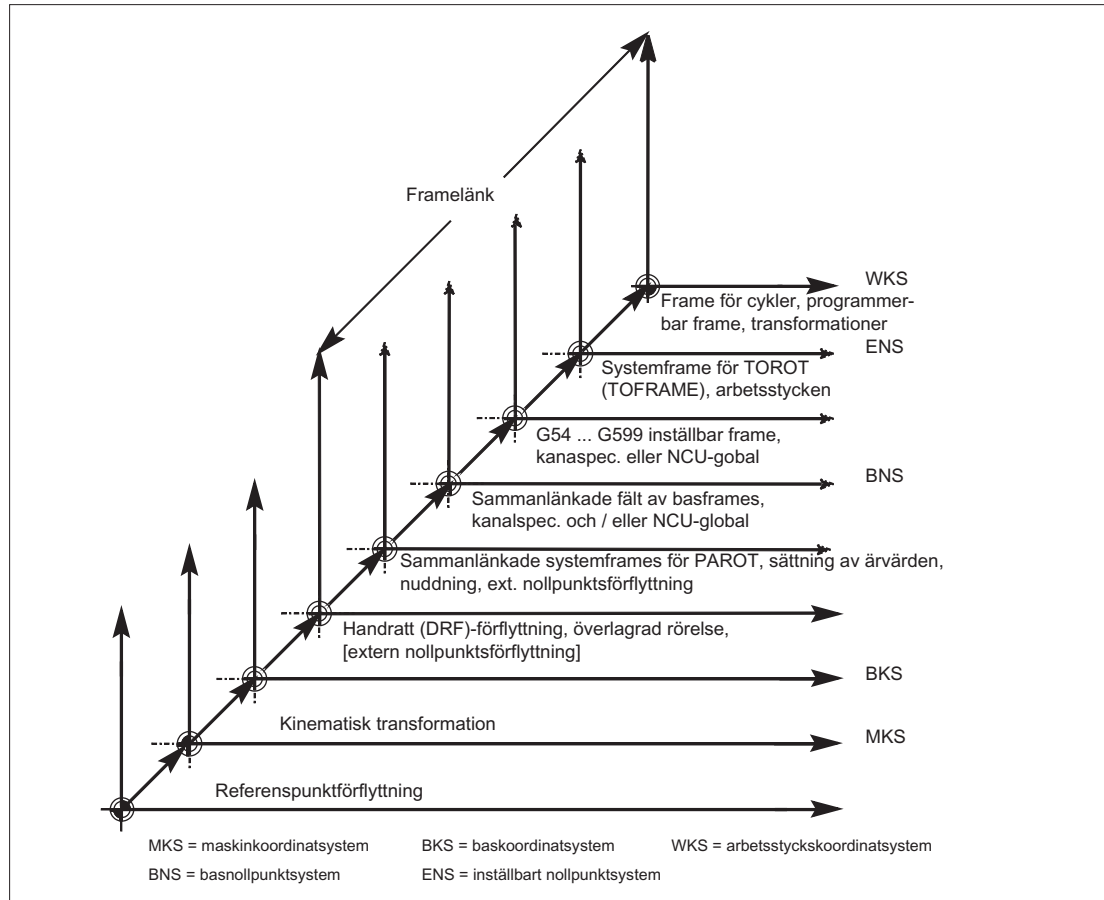
\$P_PARTFRAME : \$P_SETFRAME : \$P_EXTFRAME : \$P_ACTBFRAME : \$P_IFRAME :

\$P_TOOLFRAME : \$P_WPFRAME : \$P_TRAFRAME : \$P_PFRAME : \$P_CYCFRAME



Frame-länkning

Den aktuella ramen är sammansatt av den totala basramen, den inställbara ramen, systemramen och den programmerbara ramen enligt den ovan angivna aktuella totalramen.



Transformationer

7.1 Allmän programmering av transformationstyperna

Allmän funktion

För anpassning av styrningen till olika maskinkinematiker finns urvalet transformationstyper med lämpliga parametrar att programmera. Via dessa parametrar kan för den valda transformationen både orienteringen av verktyget i rymden och även orienteringsrörelserna för de roterande axlarna göras en motsvarande överenskommelse.

Vid tre-, fyr- och femaxel transformationerna hänför sig de programmerade positionsuppgifterna alltid till verktygets spets som följdstyrs ortogonalt till den i rymden förefintliga bearbetningsytan. De kartesiska koordinaterna räknas om från baskoordinatsystemet till maskinkoordinatsystemet och hänför sig till geometriaxlarna. Dessa beskriver arbetspunkten. Virtuella roterande axlar beskriver verktygets orientering i rymden och programmeras med TRAORI.

Vid den kinematiska transformationen kan positioner programmeras i det kartesiska koordinatsystemet. Styrningen transformerar de med TRANSMIT, TRACYL och TRAANG programmerade förflyttningsrörelserna i det kartesiska koordinatsystemet till förflyttningsrörelserna för de reala maskinaxlarna.

Programmering

Tre-, fyr- och fem-axel-transformationer TRAORI

Den överenskomna orienteringstransformation aktiveras med kommandot TRAORI och de tre möjliga parametrarna för trafonummer, orienteringsvektor och roterande axeloffsets.

```
TRAORI(trafonummer, orienteringsvektor, roterande axeloffsets)
```

Kinematiska transformationer

Till de kinematiska transformationerna hör de överenskomna transformationerna

```
TRANSMIT(trafonummer)
```

```
TRACYL(arbetsdiameter, trafonummer)
```

```
TRAANG(vinkel för den snett stående axeln, trafonummer)
```

Koppla från aktiv transformation

Med TRAFOOF kan den för tillfället aktiva transformationen kopplas från.

Orienteringstransformation

Tre-, fyr- och fem-axel-transformationer TRAORI

För optimal bearbetning av i rymden formade ytor i maskinens arbetsrum behöver verktygsmaskiner förutom de tre linjära axlarna X, Y och Z också extra axlar. De extra axlarna

7.1 Allmän programmering av transformationstyperna

beskriver orienteringen i rymden och nämns i det följande orienteringsaxlar. De står som vridaxlar hos fyra maskintyper med olika kinematik till förfogande.

1. Tvåaxel-vridhuvud t.ex. kardanskt verktygshuvud med en roterande axel parallellt till en linjär axel vid fast verktygsbord.
2. Tvåaxel-vridbord t.ex. fast vridhuvud med verktygsbord vridbart runt två axlar.
3. Enaxel-vridhuvud och enaxel-vridbord t.ex. ett vridbart vridhuvud med vridet verktyg vid verktygsbord vridbart runt en axel.
4. Tvåaxel-vridhuvud och enaxel-vridbord t.ex. vid verktygsbord vridbart runt en axel och ett vridbart vridhuvud med verktyg vridbart runt sig självt.

3- och 4-axel-transformationerna är specialformer av 5-axel-transformationen och programmeras analogt till 5-axel-transformationerna.

Den "**Generiska 3-/4-/5-/6-axel-transformationen**" räcker med sin omfattning av funktioner till för rätvinkligt anordnade roterande axlar samt transformationerna för det kardanska fräshuvudet och kan som vilken annan orienteringstransformation som helst också aktiveras för dessa fyra maskintyper med TRAORI. Vid den generiska 5/6-axel-transformationen har verktygsorienteringen en ytterligare tredje frihetsgrad vid vilken för valfri verktygsriktning i rymden verktyget kan vridas runt sin egen axel.

Litteratur: /FB3/ Funktionshandbok Specialfunktioner; 3- till 5-axeltransformation (F2)

Kinematikoberoende grundläge för verktygsorienteringen

ORIRESET

Är med TRAORI en orienteringstransformation aktiv, då kan med ORIRESET grundlägena för upp till 3 orienteringsaxlar med optimala parametrar A, B, C anges. Tillordningen av ordningsföljden för de programmerade parametrarna till de roterande axlarna sker enligt den genom transformationen fastlagda ordningsföljden för orienteringsaxlarna. Programmeringen av ORIRESET(A, B, C) förorsakar att orienteringsaxlarna går linjärt och synkront från sin momentana position till den angivna grundlägespositionen.

Kinematiska transformationer

TRANSMIT och TRACYL

Vid fräsbearbetningen på svarvmaskiner kan för den överenskomna transformationen antingen

1. en bearbetning i ändplanet i vriduppspänning med TRANSMIT eller
2. en bearbetning från valfritt löpande spår på cylindriska kroppar med TRACYL

programmeras.

TRAANG

Ska ansättningsaxeln t.ex. för teknologin slipa också kunna ansättas snett så kan för den överenskomna transformationen en parameterbar vinkel programmeras med TRAANG.

Kartesisk PTP-körning

Till den kinematiska transformationen hör också den "kartesiska PTP-körningen" vid vilken upp till 8 olika länkpositioner STAT= kan programmeras. Positionerna programmeras i det kartesiska koordinatsystemet varvid maskinens rörelse sker i maskinkoordinaterna.

Litteratur:

/FB2/ Funktionshandbok Tilläggfunktioner; Kinematisk transformation (M1)

Sammanlänkade transformationer

Det kan alltid kopplas två transformationer efter varandra. Vid den härigenom sammanlänkade andra transformationen övertas rörelseandelarna för axlarna från den första transformationen.

Som första transformation är möjlig:

- Orienteringstransformation TRAORI
- Polartransformation TRANSMIT
- Cylindertransformation TRACYL
- Transformation lutande axel TRAANG

Den andra transformationen måste vara lutande axel TRAANG

7.1.1 Orienteringsrörelser vid transformationerna**Förflyttningsrörelser och orienteringsrörelser**

Förflyttningsrörelserna för de programmerade orienteringarna beror primärt på maskintypen. Vid tre-, fyr- och fem-axel-transformationen med TRAORI beskriver de roterande axlarna eller de svängbara linjäraxlarna verktygets orienteringsrörelser.

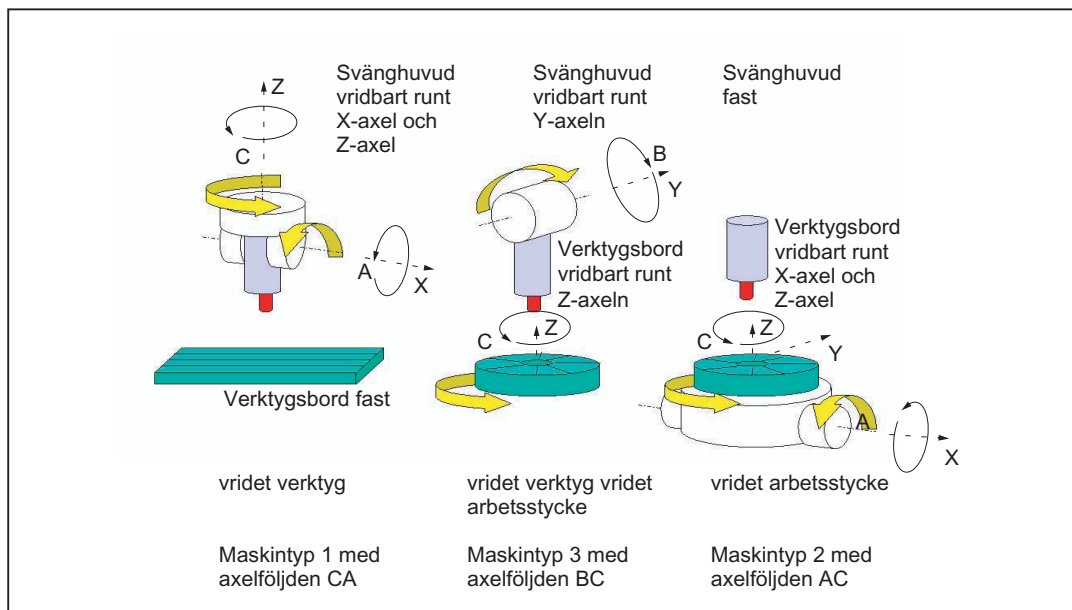
Ändringar i positionerna för de i orienteringstransformationen deltagande roterande axlarna leder till utjämningsrörelser för de övriga maskinaxlarna. Positionen för verktygsspetsen förblir därvid oförändrad.

Orienteringsrörelser för verktyget kan via de roterande axelbeteckningarna A..., B..., C... för de virtuella axlarna allt efter användning programmeras antingen genom angivande av Euler- resp. RPY-vinkeln eller riktnings- resp. ytnormalvektorer, normerade vektorer för vridaxlen för en kon eller för den intermediära orienteringen på en manteryta för kon.

Vid den kinematiska transformationen med TRANSMIT, TRACYL och TRAANG transformerar styrningen de programmerade förflyttningsrörelserna i det kartesiska koordinatsystemet till förflyttningsrörelserna för de reala maskinaxlarna.

Maskinkinematik vid tre-, fyr- och fem-axel-transformation TRAORI

Antingen kan verktyget eller verktygsbordet med upp till två roterande axlar vara vridbara. En kombination av var och en enligt vridhuvud och vridbord är också möjligt.



Maskintyp	Programmering av orienteringen
Tre-axel-transformation maskintyper 1 och 2	Programmering av verktygsorienteringen endast i det plan som är vinkelrätt mot den roterande axeln. Det existerar två translatoriska axlar (linjäraaxlar) och en roterande axel.
Fyr-axel-transformation maskintyper 1 och 2	Programmering av verktygsorienteringen endast i det plan som är vinkelrätt mot den roterande axeln. Det existerar tre translatoriska axlar (linjäraaxlar) och en roterande axel.
Fem-axel-transformation maskintyper 3 Enaxel-vridhuvud och enaxel-vridbord	Programmering av orienteringstransformationen. Kinematik med tre linjäraaxlar och två ortogonala roterande axlar. De roterande axlarna är parallella till två av de tre linjäraaxlarna. Den första roterande axeln flyttas av två kartesiska linjäraaxlar. Den roterar den tredje linjäraaxeln med verktyget. Den andra roterande axeln vridar arbetsstycket.

Generiska 5/6-axeltransformationer

Maskintyp	Programmering av orienteringstransformationen
Generisk fem-/sex-axel transformation maskintyper 4 Tvåaxel-vridhuvud med verktyg vridbart runt sig självt och enaxel-vridbord	Programmering av orienteringstransformationen. Kinematik med tre linjäraaxlar och tre ortogonala roterande axlar. De roterande axlarna är parallella till två av de tre linjäraaxlarna. Den första roterande axeln flyttas av två kartesiska linjäraaxlar. Den roterar den tredje linjäraaxeln med verktyget. Den andra roterande axeln vridar arbetsstycket. Grundorienteringen för verktyget kan genom en extra vridning runt sig självt programmeras med vridvinkeln THETA.

Vid anrop av "Generisk tre-, fyr- och fem-/sex-axel transformation" kan dessutom grundorienteringen för verktyget överföras. Inskränkningarna beträffande riktningen för de

roterande axlarna gäller inte längre. När de roterande axlarna inte står exakt vinkelrätt mot varandra eller förefintliga roterande axlar inte står exakt parallellt till de linjära axlarna kan "Generisk fem-/sex-axel transformation" leverera bättre resultat för vektysorienteringen.

Kinematiska transformationer TRANSMIT, TRACYL och TRAANG

För fräsbearbetningar på svarvmaskiner eller en snett ansättningsbar axel vid slipning gäller beroende på transformationen i standardfall följande axelanordningar:

TRANSMIT	Aktivering av polar-transformationen
Ändplansbearbetning i vriduppspänning	en roterande axel en ansättningsaxel vinkelrätt mot vridaxeln en längsaxel parallell till vridaxeln

TRACYL	Aktivering av cylindermanteltransformationen
Bearbetning av valfritt löpan- de spår på den cylindriska kroppen	en roterande axel en ansättningsaxel vinkelrätt mot vridaxeln en längsaxel parallell till vridaxeln

TRAANG	Aktivering av transformationen lutande axel
Bearbetning med lutande an- sättningsaxel	en roterande axel en ansättningsaxel med parameterbar vinkel en längsaxel parallell till vridaxeln

Kartesisk PTP-körning

Maskinens rörelser sker i maskinkoordinaterna och programmeras med:

TRAORI	Aktivering av transformationen
PTP punkt-till-punkt-körning	Uppsöka position i det kartesiska koordinatsystemet (MKS)
CP	Banrörelse för de kartesiska axlarna i (BKS)
STAT	Länkarnas position är beroende av transformationen
TU	Med vilken vinkel axlarna flyttar sig längs den kortaste vägen

PTP-körning vid generisk 5/6-axel transformation

Maskinens rörelse sker i maskinkoordinaterna och verktygsorienteringen kan programmeras både med roterande axelpositionerna och även med de av kinematiken oberoende vektorerna Euler resp. RPY-vinkel eller riktningsektorerna.

Därvid är roterande axelinterpolering, vektorinterpolering med storcirkelinterpolering eller interpolering av orienteringsvektorn på en mantelyta för kon möjlig.

Exempel tre-, till fem-axel-transformation vid ett kardanskt fräshuvud

Verktygsmaskinen har minst 5 axlar därav

- Tre translatoriska axlar för rätlinjiga rörelser, som flyttar arbetspunkten till varje valfri position i arbetsrummet.
- Två roterande vridaxlar som är anordnade med en projekterbar vinkel (för det mesta 45 grader) möjliggör för verktyget att inta orienteringar i rymden som vid 45 graders placering är begränsade till en halvkula.

7.1.2 Översikt av orienteringstransformationen TRAORI

Möjliga programmeringstyper i samband med TRAORI

Maskintyp	Programmering vid aktiv transformation TRAORI
Maskintyper 1, 2 eller 3 två-axel-vridhuvud eller tvåaxel-vridbord eller en kombination av enaxel-vridhuvud och enaxel-vridbord.	Axelföljd för orienteringsaxlarna och verktygets orienteringsriktning är antingen maskinrelaterad projekterbar via maskindata beroende av maskinkinematiken eller arbetsstycksrelaterad med programmerbar orientering oberoende av maskinkinematiken Vridriktningarna för orienteringsaxlarna i referenssystemet programmeras med: - ORIMKS referenssystem = maskinkoordinatsystem - ORIWKS referenssystem = arbetsstyckskoordinatsystem Grundinställningen är ORIWKS. Programmering av orienteringsaxlarna med: A, B, C för maskinpositionerna direkt A2, B2, C2 vinkelprogrammering för virtuella axlar med - ORIEULER via Euler-vinkel (standard) - ORIRPY via RPY-vinkel - ORIVIRT1 via virtuella orienteringsaxlar 1:a definitionen - ORIVIRT2 via virtuella orienteringsaxlar 2:a definitionen med åtskillnad av interpoleringstyp: Linjär interpolering - ORIAxes av orienteringsaxlar eller maskinaxlar Storcirkelinterpolering (interpolering av orienteringsvektorn) - ORIVECT av orienteringsaxlar Programmering av orienteringsaxlarna genom angivande av A3, B3, C3 för vektorkomponenterna (riktning-/ytnormal) Programmering av den resulterande verktygsorienteringen A4, B4, C4 för ytnormalvektorn i blockbörjan A5, B5, C5 för ytnormalvektorn i blockslutet LEAD försprångsvinkel för verktygsorienteringen TILT sidledsvinkel för verktygsorienteringen
	Interpolering av orienteringsvektorn på en mantelyta för kon Orienteringsändringar på en valfritt i rymden förefintlig mantelyta för kon genom interpolering: - ORIPLANE i planet (storcirkelinterpolering) - ORICONCW på en mantelyta för kon medurs - ORICONCCW på en mantelyta för kon moturs A6, B6, C6 för riktningsvektorn (konens vridaxel) -OICONIO interpolering på en mantelyta för kon med: A7, B7, C7 intermediära vektorer (start- och slutorientering) eller - ORICONTO på en mantelyta för kon tangentiell övergång Orienteringsändringar relaterade till en bana med - ORICURVE föreskrift för rörelsen av två kontaktpunkter via $PO[XH]=(x_e, x_2, x_3, x_4, x_5)$ orienteringspolynom till 5:e graden $PO[YH]=(y_e, y_2, y_3, y_4, y_5)$ orienteringspolynom till 5:e graden $PO[ZH]=(z_e, z_2, z_3, z_4, z_5)$ orienteringspolynom till 5:e graden - ORIPATHS glättning av orienteringsförloppet med A8, B8, C8 omorienteringsfas för verktyget motsvarar: Riktning och väglängd för verktyget vid lyftningsrörelsen

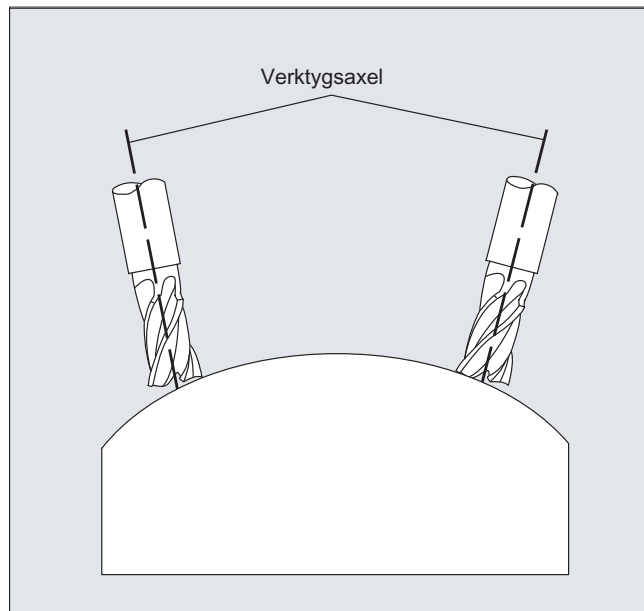
7.1 Allmän programmering av transformationstyperna

Maskintyp	Programmering vid aktiv transformation TRAORI
Maskintyper 1 och 3	Programmering av vridningarna för verktygsorienteringen med LEAD försprångsvinkel vinkel relativ till ytnormalvektorn PO[PHI] programmering av ett polynom upp till 5:e graden TILT sidledsvinkel vridning runt bantangenten (Z-riktning PO[PSI] programmering av ett polynom upp till 5:e graden THETA vridvinkel (vridning runt verktygsriktningen i Z) THETA= värde som uppnås vid blockslutet THETA=AC(...) koppla om blockvis till absolut måttuppgift THETA=IC(...) koppla om blockvis till kedjemåttuppgift THETA=Θ _e interpolera programmerad vinkel G90/G91 PO[THT]=(...) programmering av ett polynom upp till 5:e graden
Ytterligare maskintyper med extra vridning av verktyget runt sig självt kräver en 3:e roterande axel	
Orienteringstransformation, som t.ex. generisk 6-axel-transformation. Vridningar av orienteringsvektorn.	Programmering av vridvektorn - ORIOTA vridning absolut - ORIOTR relativ vridvektor - ORIOTT tangentiell vridvektor
Banrelativ orientering för orienteringsändringar relativa till banan eller vridning av vridvektor tangentiellt till banan	Orienteringsändringar relativa till banan med - ORIPATH verktygsorientering relaterad till banan - ORIPATHS extra vid en brytning i orienteringsförloppet Programmering av vridvektorn - ORIOTC tangentiell vridvektor, vridning till bantangenten

7.2 Tre-, fyr- och fem-axel-transformation (TRAORI)

7.2.1 Allmänna sammanhang kardanskt verktygshuvud

För att uppnå optimala skärvillkor vid bearbetningen av i rymden böjda ytor måste verktygets ansättningsvinkel kunna förändras.

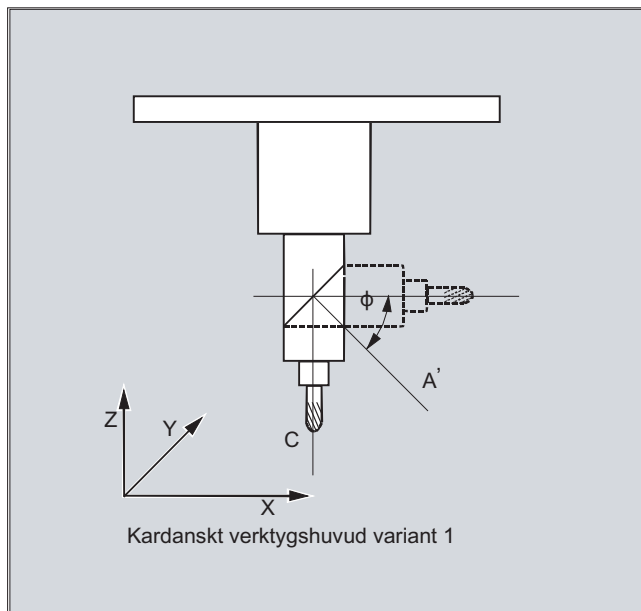


Med vilken maskinkonstruktion detta kan uppnås finns lagrat i axeldata.

5-axel-transformation

Kardaniskt verktygshuvud

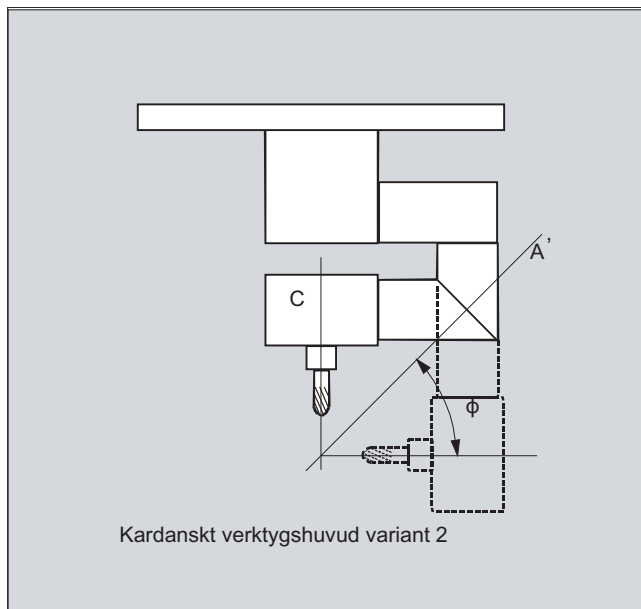
Här bestämmer tre linjärsaxlar (X, Y, Z) och två orienteringsaxlar (C, A) ansättningsvinkeln och arbetspunkten för verktyget. En av de båda orienteringsaxlarna är anlagd som lutande axel, här i exemplet A' - i mångs fall som 45°-anordning.



I de här visade exemplen ser du anordningarna i exemplet med det kardanska verktygshuvudet med maskinkinematiken CA!

Maskintillverkare

Axelföljden för orienteringsaxlarna och orienteringsriktningen för verktyget kan beroende på maskinkinematiken ställas in via maskindata.



I detta exempel ligger A' med vinkeln ϕ till X-axeln

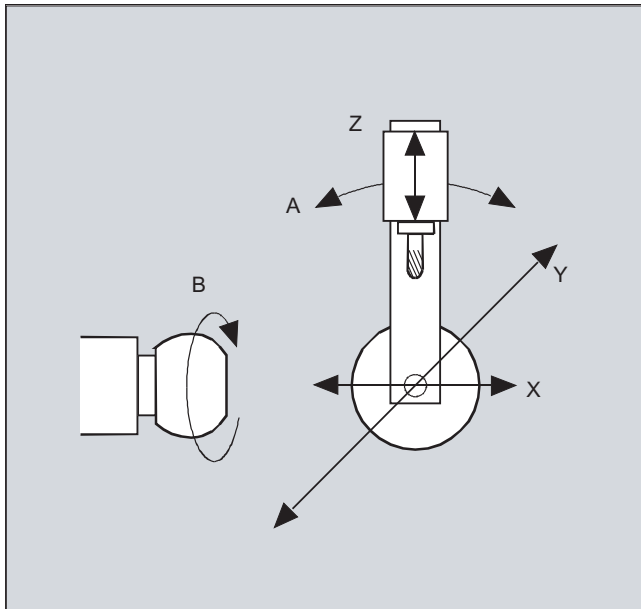
Allmänt gäller följande möjliga sammanhang:

A' ligger med vinkeln ϕ till	X-axeln
B' ligger med vinkeln ϕ till	Y-axeln
C' ligger med vinkeln ϕ till	Z-axeln

Vinkeln ϕ kan projekteras inom området 0° till $+89^\circ$ via maskindata.

Med vridbar linjärxel

Härvid rör det sig om en anordning med rörligt arbetsstycke och rörligt verktyg. Kinematiken är sammansatt av tre linjäraxlar (X, Y, Z) och två rätvinkligt anordnade vridaxlar. Den första roterande axeln flyttas t.ex. via en korsslid av två linjäraxlar, verktyget står parallellt till den tredje linjärxelen. Den andra roterande axeln vrider arbetsstycket. Den tredje linjärxelen (vridaxeln) ligger i korsslidens plan.



Axelföljden för de roterande axlarna och orienteringsriktningen för verktyget kan beroende på maskinkinematiken ställas in via maskindata.

Det gäller följande möjliga sammanhang:

Axlar:	Axelföljder:
1:a roterande axel	A A B B C C
2:a roterande axel	B C A C A B
Svängd linjärxel	Z Y Z X Y X

Ytterligare förklaringar till konfigurerbara axelföljder för verktygets orienteringsriktning se

Literatur: /FB3/ Funktionshandbok Specialfunktioner; 3- till 5-axel-transformationer (F2), Kapitel Kardaniskt fräshuvud, "Parametrering".

7.2.2 Tre-, fyr- och fem-axel-transformation (TRAORI)

Användaren kan projektera två resp. tre translatoriska axlar och en roterande axel. Transformationerna utgår ifrån att den roterande axeln står ortogonalt till orienteringsplanet.

Orienteringen av verktyget är möjlig endast i det plan som är vinkelrätt till den roterande axeln. Transformationen stöder maskintyperna med rörligt verktyg och rörligt arbetsstycke.

Projekteringen och programmeringen av tre- och fyr-axel-transformationerna är analoga till fem-axel-transformationerna.

Litteratur:

Funktionshandbok Specialfunktioner; Fleraxeltransformationer (F2)

Syntax

TRAORI (<n>)

TRAORI (<n>, <X>, <Y>, <Z>, <A>,)

TRAFOOF

Betydelse

TRAORI:	Aktiverar den första överenskomna orienteringstransformationen	
TRAORI (<n>):	Aktiverar den med n överenskomna orienteringstransformationen	
<n>:	Nummer på transformationen	
	Värde:	1 eller 2
	Exempel: TRAORI(1) aktiverar orienteringstransformation 1	
<X>, <Y>, <Z>:	Komponent av orienteringsvektorn i vilken verktyget pekar.	
<A>, :	Programmerbar offset för de roterande axlarna	
TRAFOOF:	Koppla från transformation	

Verktygsorientering

Beroende på den valda orienteringsriktningen för verktyget måste i NC-programmet det aktiva arbetsplanet (G17, G18, G19) ställas in så att verktygslängdskompenseringen verkar i verktygsorienteringens riktning.

Märk

Efter tillkopplingen av transformationen hänför sig positionsuppgifter (X, Y, Z) alltid till verktygets spets. Ändring av positionerna för de i transformationen deltagande roterande axlarna leder till utjämningsrörelser för de övriga maskinaxlarna varigenom positionen för verktygsspetsen förblir oförändrad.

Orienteringstransformationen är alltid riktad från verktygsspetsen till verktygsinfästningen.

Offset för orienteringsaxlar

Vid aktivering av orienteringstransformationen kan en extra offset för orienteringsaxlarna programmeras direkt.

Parametrar får utelämnas när den riktiga ordningsföljden respekteras vid programmeringen.

Exempel:

TRAORI (, , , , A, B) ; när bara en enda offset ska matas in

Alternativt till den direkta programmeringen kan den extra offseten för orienteringsaxlar också övertas automatiskt från den momentant aktiva nollpunktsförflyttningen. Överföringen projekteras via maskindata.

Exempel

TRAORI (1,0,0,1)	; Grundorienteringen för verktyget pekar i Z-riktning
TRAORI (1,0,1,0)	; Grundorienteringen för verktyget pekar i Z-riktning
TRAORI (1,0,1,1)	; Grundorienteringen för verktyget pekar i Y/Z-riktning (motsvarar läget -45°)

7.2.3 Varianter för orienteringsprogrammering och grundläge (ORIRESET)

Orienteringsprogrammering av verktygsorienteringen vid TRAORI

I förbindelse med en programmerbar orienteringstransformation TRAORI kan förutom till de linjära axlarna X, Y, Z också via de roterande axelbeteckningarna A..., B..., C... axelpositioner eller virtuella axlar med vinklar eller vektorkomponenter programmeras. För orienterings- och maskinaxlar är olika interpoleringstyper möjliga. Oberoende av vilka orienteringspolynom PO[vinkel] och axelpolynom PO[axel] som momentant är aktiva kan flera olika polynomtyper som t.ex. G1, G2, G3, CIP eller POLY vara programmerade.

Ändringen av verktygets orientering kan också programmeras via orienteringsvektorer. Härvid kan slutorienteringen för varje block antingen ske genom direkt programmering av vektorn eller genom programmering av de roterande axlarnas positioner.

Varianter för orienteringsprogrammering vid tre- till fem-axel-transformationer

Följande varianter av orienteringsprogrammering utesluter inte varandra.

A, B, C	Direkt angivande av positionerna för de roterande axlarna
A2, B2, C2	vinkelprogrammering virtuella axlar via Eulervinkel eller RPY-vinkel.
A3, B3, C3	Angivande av vektorkomponenterna
LEAD, TILT	Angivande av försprågs- och sidledsvinkel relaterade till banan och ytan
A4, B4, C4 A5, B5, C5	Ytnormalvektor i blockbörjan och i blockslutet
A6, B6, C6 A7, B7, C7	Interpolering av orienteringsvektorn på en konisk mantelyta
A8, B8, C8	Orientering av verktyget, riktning och väglängd för lyftningsrörelsen

Uppsökning av grundläge för verktygsorienteringen (ORIRESET)

Genom `ORIRESET ( . . . )` förflyttas orienteringsaxlarna för respektive maskinkinematik linjärt och synkront från sin aktuella position till den programmerade grundlägespositionen. Om ingen grundlägesposition har programmerats för en axel, då används positionen från det tillhörande maskindatumet `$MC_TRAFO5_ROT_AX_OFFSET_1/2`.

Det tas ingen hänsyn till aktiva frames för de roterande axlarna.

Exempel för maskinkinematik CA (kanalaxelnamn C, A)

Kommando	Beskrivning
<code>ORIRESET(90, 45)</code>	Axel C: 90° Axel A: 45°
<code>ORIRESET(, 30)</code>	Axel C: <code>\$MC_TRAFO5_ROT_AX_OFFSET_1/2[0]</code> Axel A: 30°
<code>ORIRESET()</code>	Axel C: <code>\$MC_TRAFO5_ROT_AX_OFFSET_1/2[0]</code> Axel A: <code>\$MC_TRAFO5_ROT_AX_OFFSET_1/2[1]</code>

Exempel för maskinkinematik CAC (kanalaxelnamn C, A, B)

Kommando	Beskrivning
<code>ORIRESET(90, 45, 90)</code>	Axel C: 90° Axel A: 45° Axel B: 90°
<code>ORIRESET()</code>	Axel C: <code>\$MC_TRAFO5_ROT_AX_OFFSET_1/2[0]</code> Axel A: <code>\$MC_TRAFO5_ROT_AX_OFFSET_1/2[1]</code> Axel B: <code>\$MC_TRAFO5_ROT_AX_OFFSET_1/2[2]</code>

Märk

Uppsökningen av grundläget för verktygsorienteringen med `ORIRESET ( . . . )` får ske endast vid aktiv orienteringstransformation `TRAORI ( . . . )`.

Programmering av vridningarna LEAD, TILT och THETA

Försprångsvinkel LEAD och sidledsvinkel TILT

Vridningarna för verktygsorienteringen programmeras vid tre- till fem-axel-transformationerna med försprångsvinkeln LEAD och sidledsvinkeln TILT.

Vridvinkel THETA

Vid en transformation med en tredje roterande axel kan både för orienteringen med vektorkomponenter och även med programmeringen av vinkeln LEAD, TILT rotationen av verktyget runt sig självt med vridvinkeln THETA programmeras.

7.2.4 Programmering av verktygsorienteringen (A..., B..., C..., LEAD, TILT)

För programmeringen av orienteringen av verktyget finns det följande möjligheter:

1. Direkt programmering av rörelserna för de roterande axlarna. Orienteringsändringen sker alltid i bas- resp. maskinkoordinatsystemet. Orienteringsaxlarna förflyttas som synkronaxlar.
2. Programmering i Euler- eller RPY-vinklar enligt vinkeldefinition via A2, B2, C2.
3. Programmering av riktningsvektorn via A3, B3, C3. Riktningsvektorn pekar från verktygsspetsen i riktning mot verktygsinfästningen.
4. Programmering av ytnormalvektorn vid blockbörjan med A4, B4, C4 och vid blockslutet med A5, B5, C5 (ändplansfräsning).
5. Programmering via försprångsvinkel LEAD och sidledsvinkel TILT
6. Programmering av vridaxeln för konen som normerad vektor via A6, B6, C6 eller den intermediära orienteringen på mantelytan för konen via A7, B7, C7, se Kapitel "Orienteringsprogrammering längs en konisk mantelyta (ORIPLANE, ORICONxx)".
7. Programmering av omorienteringen, riktning och väglängd för verktyget under lyftningsrörelsen via A8, B8, C8, se Kapitel "Glättning av orienteringsförloppet (ORIPATHS A8=, B8=, C8=)"

Märk

I alla fall är orienteringsprogrammeringen endast tillåten när en orienteringstransformation är tillkopplad.

Fördel: Dessa program kan överföras på varje maskinkinematik.

Definition av verktygsorienteringen via G-kommando

Märk**Maskintillverkare**

Vid maskindatum kan kopplas om mellan Euler- eller RPY-vinklar. Vid motsvarande maskindatainställningar är en omkoppling både beroende och även oberoende av det aktiva G-kommandot i grupp 50 möjlig. Det kan väljas mellan följande inställningsmöjligheter:

1. När både maskindata för definitionen av orienteringsaxlarna och definition av orienteringsvinkeln via G-kommando är satta på noll är:
De med A2, B2, C2 programmerade vinklarna blir **beroende av maskindata** vinkeldefinition för orienteringsprogrammering interpreterade antingen som Euler- eller RPY-vinklar.
2. När maskindatum för definitionen av orienteringsaxlarna via G-kommando är satt på ett sker omkoppling
beroende på det aktiva G-kommandot i grupp 50:
De med A2, B2, C2 programmerade vinklarna blir interpreterade enligt ett av de aktiva G-kommandona ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2, ORIAXPOS och ORIPY2. De med orienteringsaxlarna programmerade värdena interpreteras enligt det aktiva G-kommandot i grupp 50 också som orienteringsvinklar.
3. När maskindatum för definitionen av orienteringsvinklarna via G-kommando är satt på ett och maskindatum för definitionen av orienteringsaxlarna via G-kommando är satt på noll sker omkoppling
oberoende av det aktiva G-kommandot i grupp 50:
De med A2, B2, C2 programmerade vinklarna interpreteras enligt ett av de aktiva G-kommandona ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2, ORIAXPOS och ORIPY2. De med orienteringsaxlarna programmerade värdena interpreteras oberoende av det aktiva G-kommandot i grupp 50 alltid som roterande axelpositioner.

Syntax

Roterande axelpositioner

```
G1 X<Wert> Y<Wert> Z<Wert> A<Wert> B<Wert> C<Wert>
```

Eulervinkel

```
G1 X<Wert> Y<Wert> Z<Wert> A2=<Wert> B2=<Wert> C2=<Wert>
```

Riktningsvektor

```
G1 X<Wert> Y<Wert> Z<Wert> A3=<Wert> B3=<Wert> C3=<Wert>
```

Ytnormalvektorn i blockbörjan

```
G1 X<Wert> Y<Wert> Z<Wert> A4=<Wert> B4=<Wert> C4=<Wert>
```

Ytnormalvektorn i blockslutet

```
G1 X<Wert> Y<Wert> Z<Wert> A5=<Wert> B5=<Wert> C5=<Wert>
```

Försprångsvinkel

LEAD=<Wert>

Sidledsvinkel

TILT=<Wert>

Betydelse

G1:	Linjärinterpolering
X, Y, Z:	Linjärxelpositioner
A, B, C:	Roterande axelpositioner
A2=, B2=, C2=:	Vinkelprogrammering (Euler- eller RPY-vinkel)
A3=, B3=, C3=:	Riktningsvektor i X, Y och Z-koordinater i WKS
A4=, B4=, C4=:	Ytnormalvektorn i blockbörjan i X, Y och Z-koordinater i WKS
A5=, B5=, C5=:	Ytnormalvektor i blockslutet i X, Y och Z-koordinater i WKS
LEAD= :	Försprångsvinkel ¹⁾
TILT= :	Sidledsvinkel ¹⁾
1) Interpreteringen av vinkeluppgifterna är beroende av inställningen i MD21094 \$MC_ORI-PATH_MODE	

Ytterligare informationer

Som regel skapas 5-axel-program av CAD/CAM-systemen och matas inte in till styrningen. Därför vänder sig följande förklaringar huvudsakligen till programmerare för postprocessorer.

Följande kommandon för orienteringsprogrammeringen står till förfogande:

Kommando	Betydelse
ORIEULER:	Euler-vinkel med rotationsordningsföljd ZX'Z"
ORIRPY:	RPY-vinkel med rotationsordningsföljd XY'Z"
ORIRPY2:	RPY-vinkel med rotationsordningsföljd ZY'X"
ORIVIRT1:	Virtuella orienteringsaxlar med fritt definierbar rotationsordningsföljd via: MD21120 \$MC_ORIAX_TURN_TAB_1
ORIVIRT2:	Virtuella orienteringsaxlar med fritt definierbar rotationsordningsföljd via: MD21130 \$MC_ORIAX_TURN_TAB_2
ORIAPOS:	Virtuella orienteringsaxlar med positioner för roterande axlar

Märk

Via maskindata kan olika varianter definieras av maskintillverkaren. Följ anvisningarna från maskintillverkaren.

Programmering i Eulervinklar ORIEULER, rotationsordningsföljd Z X' Z"

De vid orienteringsprogrammeringen ORIEULER med A2, B2, C2 programmerade värdena interpreteras som Eulervinkel (i grader).

7.2 Tre-, fyr- och fem-axel-transformation (TRAORI)

Den nya orienteringsvektorn resulterar ur följande tre rotationer av den ursprungliga orienteringsvektorn

1. med den roterande axeln A_2 runt koordinataxeln Z
2. med den roterande axeln B_2 runt den nya koordinataxeln X'
3. med den roterande axeln C_2 runt den nya koordinataxeln Z''

I detta fall är värdet av C_2 (vridning runt ny Z -axel) utan betydelse och måste inte programmeras.

Programmering i RPY-vinklar $ORIRPY$, rotationsordningsföljd $X' Y' Z''$

De vid orienteringsprogrammeringen $ORIRPY$ med A_2 , B_2 , C_2 programmerade värdena interpreteras som RPY-vinklar (i grader) med rotationsordningsföljd $X' Y' Z''$.

Märk

I motsats till programmering med $ORIEULER$ har vid $ORIRPY$ alla tre värdena inflytande på orienteringsvektorn.

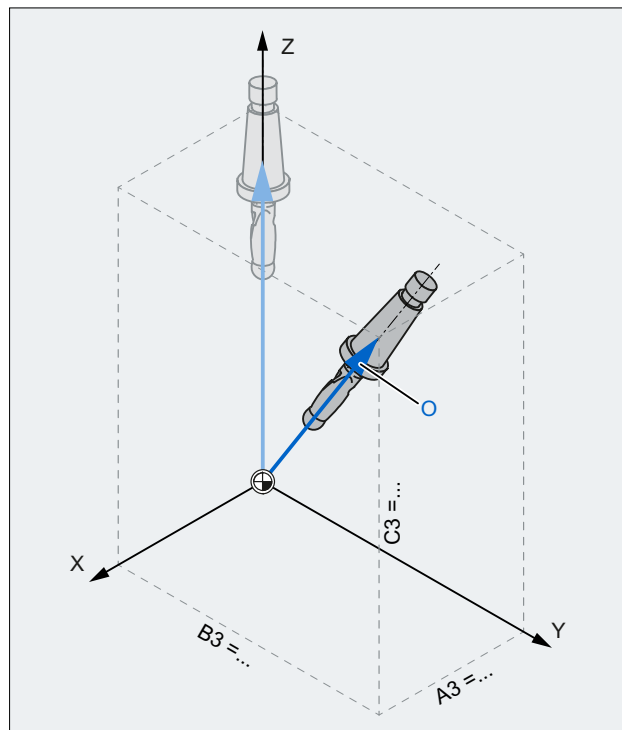
Den nya orienteringsvektorn resulterar ur följande tre rotationer av den ursprungliga orienteringsvektorn

1. med den roterande axeln A_2 runt koordinataxeln X
2. med den roterande axeln B_2 runt den nya koordinataxeln Y'
3. med den roterande axeln C_2 runt den nya koordinataxeln Z''

Programmering av riktningsvektorn

Komponenterna för riktningsvektorn programmeras med A_3 , B_3 , C_3 . Vektorn pekar i riktning mot verktygsinfästningen; längden på vektorn är därvid utan betydelse.

Ej programmerade verktygskomponenter sätts lika med noll.



X, Y, Z Koordinataxlar i WKS

A3, B3, Komponenter till riktningsvektorn
C3

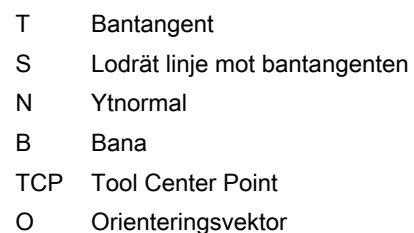
O Orienteringsvektor

Bild 7-1 Programmering av riktningsvektorn

Programmering av verktygsorienteringen med LEAD och TILT

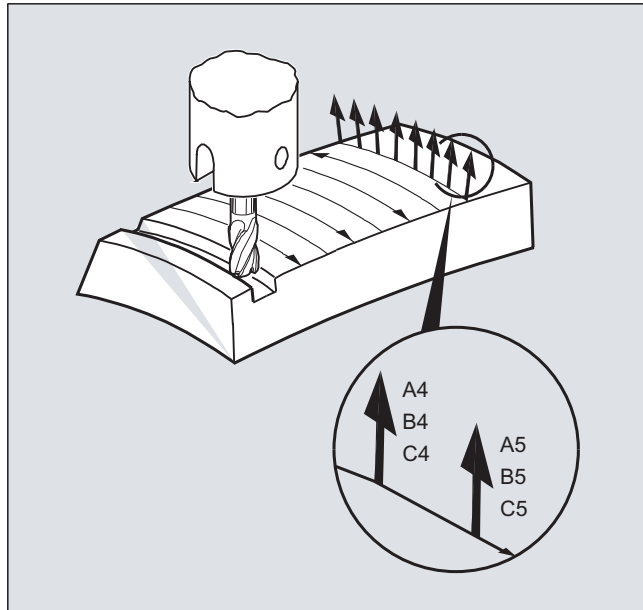
Den resulterande verktygsorienteringen fastställs genom:

- Bantangent
- Ytnormalvektor
I blockbörjan A4, B4, C4 och i blockslutet A5, B5, C5
- Försprångsvinkel LEAD
Vinkeln i det av bantangenten och ytnormalvektorn bildade planet
- Sidledsvinkel TILT i blockslutet
Vinkel i planet vinkelrätt mot bantangenten relativ till ytnormalvektorn



7.2.5 Ändplansfräsning (A4, B4, C4, A5, B5, C5)

Ändplansfräsning tjänar till bearbetning av valfritt böjda ytor.



För denna typ av 3D-fräsning behöver du beskrivningen rad för rad av 3D-banorna på arbetsstycksytan.

Beräkningarna görs under hänsynstagandet till verktygsform och verktygedimensioner vanligen i CAM. De färdigberäknade NC-blocken läses sedan in i styrningen via postprocessorerna.

Programmering av banböjningen

Beskrivning av ytorna

Beskrivningen av banböjningen görs via ytnormalvektorerna med följand komponenter:

A4, B4, C4 Startvektor i blockbörjan

A5, B5, C5 Slutvektor i blockslutet

Står endast startvektorn i ett block förblir ytnormalvektorn konstant över hela blocket. Står endast slutvektorn i ett block så interpoleras från slutvärdet i det föregående blocket via storcirkelinterpolering till det programmerade slutvärdet.

Är start- och slutvektor programmerade så interpoleras mellan båda riktningarna likaså via storcirkelinterpolering. Härigenom låter sig kontinuerliga släta banvägar skapas.

I grundläget pekar ytnormalvektorer oberoende av det aktiva planet G17 till G19 i Z-riktningen.

Längden för en vektor är utan betydelse.

Ej programmerade verkforkomponenter sätts lika med noll.

Vid aktiv ORIWKS (se "Referens för orienteringsaxlarna (ORIWKS, ORIMKS) (Sida 334)") hänförs sig ytnormalvektorerna till den aktiva framen och vrids med vid framevridning.

Maskintillverkare

Ytnormalvektorn måste stå inom ett via maskindatum inställbart gränsvärde vinkelrätt till bantangenten annars matas larm ut.

7.2.6 Referens för orienteringsaxlarna (ORIWKS, ORIMKS)

Vid orienteringsprogrammering i arbetsstyckskoordinatsystemet via

- Euler- resp. RPY-vinkel eller
- Orienteringsvektor

kan förloppet för vridningsrörelsen ställas in via ORIMKS/ORIWKS.

Märk**Maskintillverkare**

Interpoleringstypen för orienteringen fastläggs med maskindatumet:

MD21104 \$MC_ORI_IPO_WITH_G_CODE

= FALSE: Referens är G-kommandona ORIWKS och ORIMKS

= TRUE: Referens är G-kommandona till den 51:a gruppen (ORIAxes, ORIVect, ORIPlane, ...)

Syntax

ORIMKS=...

ORIWKS=...

Betydelse

ORIMKS:	Vridning i maskinkoordinatsystemet
ORIWKS:	Vridning i arbetsstyckskoordinatsystemet

Märk

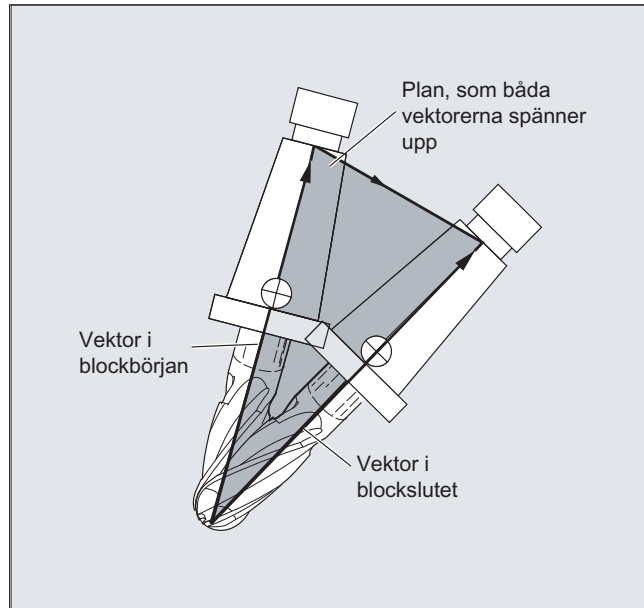
ORIWKS är grundläge. Är det för ett fem-axel-program inte klart från första början på vilken maskin det ska köras så ska principiellt ORIWKS väljas. Vilka rörelser maskinen verkligen utför beror på maskinkinematiken.

Med ORIMKS kan verkliga maskinrörelser programmeras t.ex. för att förhindra kollisioner med anordningar och liknande.

Ytterligare information

Vid **ORIMKS** är den utförda verktygsrörelsen **beroende** av maskinkinematiken. Vid orienteringsändring med i rummen fast verktygsspets interpoleras linjärt mellan positionerna för de roterande axlarna.

Vid **ORIWKs** är den utförda verktygsrörelsen **oberoende** av maskinkinematiken. Vid orienteringsändring med i rummen fast verktygsspets flyttar sig verktyget i det av start- och slutvektor bildade planet.



Singulära lägen

Märk

ORIWKs

Orienteringsrörelser i området för det singulära läget för fem-axel-maskinen kräver stora rörelser för maskinaxlarna. (Till exempel är för ett vridande svänghuvud med C som vridaxel och A som svängaxel alla lägen med A=0 singulära.)

Maskintillverkare

För att inte överbelasta maskinaxlarna sänker hastighetsstyrningen banhastigheten kraftigt i närheten av de singulära ställena.

Med maskindata

```
$MC_TRAFO5_NON_POLE_LIMIT
```

```
$MC_TRAFO5_POLE_LIMIT
```

kan transformationen parametreras så att orienteringsrörelserna i närheten av polen läggs genom polen och en snabb bearbetning är möjlig.

Singulära ställen behandlas endast med MD `$MC_TRAFO5_POLE_LIMIT` .

Litteratur:

/FB3/ Funktionshandbok Specialfunktioner; 3- till 5-axel-transformation (F2),
Kapitel "Singulära ställen och deras behandling".

7.2.7 Programmering av orienteringsaxlarna (ORIAXES, ORIVECT, ORIEULER, ORIRPY, ORIRPY2, ORIVIRT1, ORIVIRT2)

Funktionen "Orienteringsaxlar" beskriver verktygets orientering i rummet och uppnås genom programmering av offsets för de roterande axlarna. En ytterligare tredje frihetsgrad kan uppnås genom den extra vridningen av verktyget runt sig självt. Denna verktygsorientering görs valfritt i rummet via en tredje roterande axel och kräver sex-axel-transformationen. Verktygets egenvridning runt sig självt fastläggs beroende av interpoleringstypen för vridvektorena med vridvinkeln THETA (se "Vridningar av verktygsorienteringen (ORIROTA, ORIROTR, ORIROTT, ORIROTC, THETA) (Sida 345)".

Orienteringsaxlar programmeras via axelbeteckningarna A2, B2, C2.

Syntax

```
N... ORIAXES/ORIVECT           ; Linjär eller storcirkelinterpolering
N... G1 X Y Z A B C

N... ORIPLANE                   ; Orienteringsinterpolering i planet

N... ORIEULER/ORIRPY/ORIRPY2   ; Orienteringsvinkel Euler-/RPY-vinkel
N... G1 X Y Z A2= B2= C2=       ; Vinkelprogrammering virtuella axlar

N... ORIVIRT1/ORIVIRT2         ; Virtuella orienteringsaxlar def. 1/2
N... G1 X Y Z A3= B3= C3=       ; Riktningsvektorprogrammering
```

Märk

För orienteringsändringar längs en i rummet förefintlig konisk mantelyta kan ytterligare offsets för roterande axlar för orienteringsaxlarna programmeras (se "Orienteringsprogrammering längs en konisk mantelyta (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO) (Sida 338)".

Betydelse

ORIAXES:	Linjär interpolering av maskin- eller orienteringsaxlar
ORIVECT:	Storcirkelinterpolering (identisk med ORIPLANE)
ORIMKS: ORIWKS:	Vridning i maskinkoordinatsystemet Vridning i arbetsstyckskoordinatsystemet Beskrivning se "Referens för orienteringsaxlarna (ORIWKS, ORIMKS) (Sida 334)".
A= B= C=:	Programmering av maskinaxelposition

ORIEULER:	Orienteringsprogrammering via Euler-vinkel
ORIRPY:	Orienteringsprogrammering via RPY-vinkel Ordningsföljden för vridningen är XYZ, varvid gäller: • A2 är rotationsvinkeln runt X • B2 är rotationsvinkeln runt Y • C2 är rotationsvinkeln runt Z
ORIRPY2:	Orienteringsprogrammering via RPY-vinkel Ordningsföljden för vridningen är ZYX, varvid gäller: • A2 är rotationsvinkeln runt Z • B2 är rotationsvinkeln runt Y • C2 är rotationsvinkeln runt X
A2= B2= C2=:	Vinkelprogrammering virtuella axlar
ORIVIRT1/ORIVIRT2:	Orienteringsprogrammering via virtuella orienteringsaxlar Definition 1: Fastläggande enligt MD21120 \$MC_ORIAX_TURN_TAB_1 Definition 2: Fastläggande enligt MD21130 \$MC_ORIAX_TURN_TAB_2
A3= B3= C3=:	Riktningsvektorprogrammering för riktningsaxeln

Ytterligare informationer

Maskintillverkare

Med MD21102 \$MC_ORI_DEF_WITH_G_CODE fastläggs hur de programmerade vinklarna A2, B2, C2 definieras:

Definition görs enligt MD21100 \$MC_ORIENTATION_IS_EULER (standard) eller definition görs enligt G-grupp 50 (ORIEULER, ORIRPY, ORIVIRT1, ORIVIRT2).

Med MD21104 \$MC_ORI_IPO_WITH_G_CODE fastläggs vilken interpoleringstyp som är verksam: ORIWS/ORIMKS eller ORIAXES/ORIVECT.

Driftsläge JOG

Orienteringsvinkeln interpoleras i detta driftsläge alltid linjärt. Vid kontinuerlig och inkrementell förflyttning via förflyttningstangenter kan endast en orienteringsaxel förflyttas. Via handrattarna kan orienteringsaxlarna förflyttas samtidigt.

För den manuella förflyttningen av orienteringsaxlar verkar den kanalspecifika övermanningsomkopplaren för matning resp övermanningsomkopplaren för snabbtransport vid snabbgångsöverlagring.

Med följande maskindata är en separat hastighetsföreskrift möjlig:

MD21160 \$MC_JOG_VELO_RAPID_GEO

MD21165 \$MC_JOG_VELO_GEO

MD21150 \$MC_JOG_VELO_RAPID_ORI

MD21155 \$MC_JOG_VELO_ORI

Märk**SINUMERIK 840D sl med "Transformationspaket Handling"**

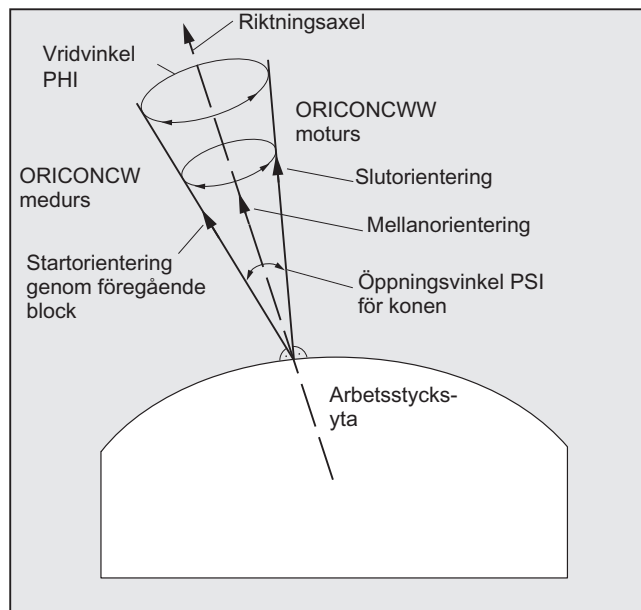
Med funktionen "Kartesisk manuell förflyttning" kan i JOG-drift translationen av geometriaxlar ställas in separat från varandra i referenssystemen MKS, WKS och TCS.

Litteratur:

Funktionshandbok Tilläggsfunktioner; Kinematisk transformation (M1)

7.2.8**Orienteringsprogrammering längs en konisk mantelyta (ORIPLANE, ORICONCW, ORICONCCW, ORICONTO, ORICONIO)**

Med den utvidgade orienteringen är det möjligt att utföra orienteringsändringar längs en konisk mantelyta som befinner sig i rymden. Interpoleringen av orienteringsvektorn på en konisk mantelyta görs med de modala kommandona ORICONxx. För interpoleringen i ett plan kan slutorienteringen programmeras med ORIPLANE. Generellt fastläggs startorienteringen av de föregående blocken.



Programmering

Slutorienteringen fastläggs antingen genom angivande av vinkelprogrammeringen i Euler- eller RPY-vinkel med A2, B2, C2 eller genom programmering av positionerna för de roterande axlarna med A, B, C. För orienteringsaxlarna längs den koniska mantelytan är ytterligare programmeringsuppgifter erforderliga:

- Vridaxel för konen som vektor med A6, B6, C6
- Öppningsvinkel PSI med beteckningen NUT
- Intermediär orientering i den koniska mantelytan med A7, B7, C7

Märk

Programmering av riktningsvektorn A6, B6, C6 för konens vridaxel

Programmeringen av en slutorientering är inte absolut nödvändigt. Finns ingen slutorientering angiven då interpoleras en hel konmantel med 360 grader.

Programmering av konens öppningsvinkel med NUT=winkel

Angivande av en slutorientering är absolut nödvändigt.

En fullständig konmantel med 360 grader kan inte interpoleras på detta sätt.

Programmering av den intermediära orienteringen A7, B7, C7 i konmanteln

Angivande av en slutorientering är absolut nödvändigt. Orienteringsändringen och vridriktningen fastläggs entydigt av de tre vektorerna start-, slut- och intermediär orientering. Alla tre vektorerna måste härvid vara olika från varandra. Är den programmerade intermediära orienteringen parallell till start- eller slutorienteringen då genomförs en linjär storcirkelinterpolering av orienteringen i det plan som bildas av start- och slutvektorn.

Utvidgad orienteringsinterpolering på en konisk mantelyta

N... ORICONCW oder ORICONCCW
 N... A6= B6= C6= A3= B3= C3=
 eller
 N... ORICONTO
 N... G1 X Y Z A6= B6= C6=
 eller
 N... ORICONIO
 N... G1 X Y Z A7= B7= C7=
 N... PO[PHI]=(a2, a3, a4, a5)
 N... PO[PSI]=(b2, b3, b4, b5)

Interpolering på en konmantel med riktningsvektor medurs/moturs för konen och slutorientering eller tangentiell övergång och angivande av slutorienteringen eller angivande av slutorienteringen och en intermediär orientering i konmanteln med polynom för vridvinkel och polynom för öppningsvinkel

Parametrar

ORIPLANE:	Interpolering i planet (storcirkelinterpolering)
ORICONCW:	Interpolering på en konisk mantelyta medurs
ORICONCCW:	Interpolering på en konisk mantelyta moturs
ORICONTO:	Interpolering på en koniska mantelyta tangentiell övergång

7.2 Tre-, fyr- och fem-axel-transformation (TRAORI)

A6= B6= C6=:	Programmering av vridaxlen till konen (normerad vektor)
NUT=winkel:	Öppningsvinkel för konen i grader
NUT=+179:	Förflyttningsvinkel mindre eller lika med 180 grader
NUT=-181:	Förflyttningsvinkel större eller lika med 180 grader
ORICONIO:	Interpolering på en konisk mantelyta
A7= B7= C7=:	Intermediär orientering (programmering som normerad vektor)
PHI:	Vridvinkel för orienteringen runt riktningsaxeln till konen
PSI:	Öppningsvinkel för konen
möjliga polynom PO[PHI]=(a2, a3, a4, a5) PO[PSI]=(b2, b3, b4, b5)	Förutom respektive vinklar är också polynom maximalt 5:e graden programmerbara

Exempel: Olika orienteringsändringar

Programkod	Kommentar
...	
N10 G1 X0 Y0 F5000	
N20 TRAORI(1)	; Orienteringstransformation till.
N30 ORIVECT	; Interpolera verktygsorientering som vektor.
...	; Verktygsorientering i planet.
N40 ORIPANE	; Välja storcirkelinterpolering.
N50 A3=0 B3=0 C3=1	
N60 A3=0 B3=1 C3=1	; Orientering i Y/Z-planet vridet med 45 grader, vid blockslutet uppnås orienteringen $(0, 1/\sqrt{2}, 1/\sqrt{2})$.
...	
N70 ORICONCW	; Orienteringsprogrammering på konmantel:
N80 A6=0 B6=0 C6=1 A3=0 B3=0 C3=1	; Orienteringsvektorn interpoleras på en konmantel med riktningen $(0, 0, 1)$ till orienteringen $(1/\sqrt{2}, 0, 1/\sqrt{2})$ medurs, vridvinkeln uppgår härvid till 270 grader.
N90 A6=0 B6=0 C6=1	; Verktygsorienteringen genomgår ett helt varv på samma konmantel.

Ytterligare information

Ska orienteringsändringar beskrivas på en valfritt i rymden liggande konisk mantelyta då måste den vektor runt vilken verktygsorienteringen ska vridas vara känd. Dessutom måste start- och slutorienteringen vara föreskrivna. Startorienteringen resulterar ur det föregående blocket och slutorienteringen måste antingen programmeras eller fastläggas genom andra villkor.

Programmering i planet ORIPANE motsvarar ORIVECT

Programmeringen av storcirkelinterpoleringen tillsammans med vinkelpolynom motsvarar linjär- och polynominterpolering av konturer. Verktygsorienteringen interpoleras i ett plan som bildas av start- och slutorienteringen. Programmeras ytterligare polynom då kan orienteringsvektorn också vickas ur planet.

Programmering av cirklar i ett plan G2/G3, CIP och CT

Den utvidgade orienteringen motsvarar interpoleringen av cirklar i ett plan. För motsvarande programmeringsmöjligheter för cirklar med medelpunktsuppgift eller radieuppgift som G2/G3, cirkel via intermediär punkt CIP och tangentiell cirkel CT se

Litteratur: Programmeringshandbok Grunder, "Programmera vägkommandon".

Orienteringsprogrammering**Interpolering av orienteringsvektorn på en konisk mantelyta ORICONxx**

För interpoleringen av orienteringar på en konisk mantelyta kan fyra olika interpoleringstyper väljas från G-gruppen 51:

1. Interpolering på en konmantel medurs `ORICONCW` med angivande av slutorienteringen och konriktningen eller öppningsvinkeln. Riktningsvektorn programmeras med beteckningarna `A6`, `B6`, `C6` och öppningsvinkeln för konen med beteckningen `NUT=` värdeområdet i intervallet 0 till 180 grader.
2. Interpolering på en konmantel moturs `ORICONCWW` med angivande av slutorienteringen och konriktningen eller öppningsvinkeln. Riktningsvektorn programmeras med beteckningarna `A6`, `B6`, `C6` och öppningsvinkeln för konen med beteckningen `NUT=` värdeområdet i intervallet 0 till 180 grader.
3. Interpolering på en konmantel `ORICONIO` med angivande av slutorienteringen och en intermediär orientering som programmeras med beteckningarna `A7`, `B7`, `C7`.
4. Interpolering på en konmantel `ORICONTO` med tangentiell övergång och angivande av slutorienteringen. Riktningsvektorn programmeras med beteckningarna `A6`, `B6`, `C6`.

7.2.9**Orienteringsuppgift med två kontaktpunkter (ORICURVE, PO[XH]=, PO[YH]=, PO[ZH]=)****Programmering av orienteringsändringen genom en andra rymdkurva ORICURVE**

En ytterligare möjlighet för programmeringen av orienteringsändringar består däri att förutom verktygsspetsen längs en rymdkurva också programmera rörelsen av en andra kontaktpunkt för verktyget med `ORICURVE`. Därmed kan orienteringsändringar för verktyget, som vid programmeringen av verktygsvektorn själv, fastläggas entydigt.

Maskintillverkare

Följ anvisningarna från maskintillverkaren för via maskindatum inställbara axelbeteckningar för programmeringen av den 2:a orienteringsbanan för verktyget.

Programmering

Vid denna interpoleringstyp kan för de båda rymdkurvorna punkter programmeras med `G1` resp. polynom med `POLY`. Cirklar och evolventer är inte tillåtna. Dessutom kan en spline-interpolering med `BSPLINE` och funktionen "Sammanfattning av korta spline-block" aktiveras.

Litteratur:

Funktionshandbok Grundfunktioner; Banstyrningsdrift, precisionsstopp, LookAhead (B1), Kapitel: Sammanfattning av korta spline-block

De andra splinetyperna `ASPLINE` och `CSPLINE` samt aktiveringen av en kompressor med `COMPON`, `COMPCURV` eller `COMPCAD` är inte tillåtna.

Rörelsen för de två kontaktpunkterna till verktyget kan föreskrivas vid programmeringen av orienteringspolynomen för koordinaterna upp till maximalt 5:e graden.

Utvidgad orienteringsinterpolering med extra rymdkurva och polynom för koordinaterna

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)`

Angivande av rörelsen för den andra kontaktpunkten till verktyget och ytterligare polynom för respektive koordinater

Parametrar

<code>ORICURVE</code>	Interpolering av orienteringen med uppgift om rörelsen för två kontaktpunkter till verktyget.
<code>XH YH ZH</code>	Beteckningar för koordinaterna till den andra kontaktpunkten för verktyget till den extra konturen som rymdkurva
möjliga polynom <code>PO[XH]=(xe, x2, x3, x4, x5)</code> <code>PO[YH]=(ye, y2, y3, y4, y5)</code> <code>PO[ZH]=(ze, z2, z3, z4, z5)</code>	Förutom respektive slutpunkter kan rymdkurvorna dessutom programmeras med polynom.
<code>xe, ye, ze</code>	Slutpunkter för rymdkurvan
<code>xi, yi, zi</code>	Koefficienter för polynom maximalt 5:e graden

Märk**Beteckning XH YH ZH för programmeringen av en 2:a orienteringsbana**

Beteckningarna måste väljas så att ingen konflikt uppstår med andra beteckningar för linjäraxlarna

X Y Z axlar

och roterande axlar som

A2 B2 C2 Eulervinkel resp. RPY-vinkel

A3 B3 C3 riktningsvektorer

A4 B4 C4 resp. A5 B5 C5 ytnormalvektorer

A6 B6 C6 vridvektorer resp. A7 B7 C7 koordinater för intermediär punkt

eller andra interpoleringsparametrar.

7.3 Orienteringspolynom (PO[Winkel], PO[Koordinate])

Oberoende av vilken polynominterpolering för G-gruppen 1 momentant är aktiv, kan två olika typer av orienteringspolynom upp till maximalt 5:e graden vid en tre- till fem-axel-transformation programmeras.

1. Polynom för **vinkel**: Försprångsvinkel LEAD, sidledsvinkel TILT hänförd till det plan som bildas av start- och slutorienteringen.
2. Polynom för **koordinater**: XH, YH, ZH den andra rymdkurvan för verktygsorienteringen till en referenspunkt på verktyget.

Vid en sex-axel-transformation kan för verktygsorienteringen dessutom vridningen av vridvektorn THT med polynom upp till maximalt 5:e graden för vridningar av verktyget självt programmeras.

Syntax

Orienteringspolynom av typ 1 för **vinkel**

N... PO[PHI]=(a2, a3, a4, a5)

Tre- till fem-axel-transformation

N... PO[PSI]=(b2, b3, b4, b5)

Orienteringspolynom av typ 2 för **koordinater**

N... PO[XH]=(xe, x2, x3, x4, x5)

Beteckningar för koordinaterna för den andra orienteringsbanan för verktygsorienteringen

N... PO[YH]=(ye, y2, y3, y4, y5)

N... PO[ZH]=(ze, z2, z3, z4, z5)

Dessutom kan i båda fallen ett polynom programmeras för **vridningen** vid sex-axel-transformationer med

N... PO[THT]=(c2, c3, c4, c5)

banrelativ interpolering av vridningen

eller

N... PO[THT]=(d2, d3, d4, d5)

absolut, relativ och tangentiell interpolering till orienteringsändringen

avr orienteringsvektorn. Detta är möjligt när transformationen stöder en vridvektor med en genom vridvinkeln THETA programmerbar och interpolerbar offset.

Betydelse

PO[PHI]	Vinkel i planet mellan start- och slutorientering
PO[PSI]	Vinkel som beskriver utvippningen av orienteringen ur planet mellan start- och slutorienteringen
PO[THT]	Vridvinkel som uppstår genom vridning av vridvektorn till en av de med THETA programmerade G-kommandona i grupp 54
PHI	Försprångsvinkel LEAD
PSI	Sidledsvinkel TILT

7.3 Orienteringspolynom (*PO[Winkel]*, *PO[Koordinate]*)

THETA	Vridning runt verktygsriktningen i Z
PO[XH]	X-koordinat för referenspunkten på verktyget
PO[YH]	Y-koordinat för referenspunkten på verktyget
PO[ZH]	Z-koordinat för referenspunkten på verktyget

Ytterligare information

Orienteringspolynom kan inte programmeras

- när splineinterpoleringarna ASPLINE, BSPLINE, CSPLINE är aktiva.
Polynom av typ1 för orienteringsvinkel är möjliga för varje interpoleringstyp utom spline dvs. vid linjärinterpolering med snabbtransport G00 resp. med matning G01 vid polynominterpolering med POLY och vid cirkel- resp. evolventinterpolering med G02, G03, CIP, CT, INVCW och INCCCW.
Polynom av typ2 för orienteringskoordinater är däremot endast möjligt när linjärinterpolering med snabbtransport G00 resp. med matning G01 eller polynominterpolering med POLY är aktiv.
- när orienteringen interpoleras med hjälp av axelinterpoleringen ORIAxes. I detta fall kan polynom med PO[A] och PO[B] för orienteringsaxlarna A och B programmeras direkt.

Orienteringspolynom av typ 1 med ORIVECT, ORIPLANE och ORICONxx

Vid storcirkelinterpolering och konmantelinterpolering med ORIVECT, ORIPLANE und ORICONxx är endast orienteringspolynom av typ 1 möjligt.

Orienteringspolynom av typ 2 med ORICURVE

Är interpoleringen med extra rymdkurva ORICURVE aktiv interpoleras de kartesiska komponenterna i orienteringsvektorn och endast orienteringspolynom av typ 2 är möjligt.

7.4 Vridningar av verktygsorienteringen (ORIROTA, ORIROT, ORIROT, ORIROT, THETA)

Ska vid maskintyper med rörligt verktyg också orienteringen av verktyget kunna förändras så programmeras varje block med en slutorientering. Beroende av maskinkinematiken kan antingen orienteringsriktningen för orienteringsaxlarna eller vridriktningen för orienteringsvektorn THETA programmeras. För dessa vridvektorer är olika interpoleringstyper programmerbara:

- ORIROTA: Vridvinkel för en absolut föreskriven vridriktning.
- ORIROT: Vridvinkel relativ till planet mellan start- och slutorientering.
- ORIROT: Vridvinkel relativ till ändring av orienteringsvektorn.
- ORIROT: Tangentiell vridvinkel till bantangenten.

Syntax

Endast när interpoleringstypen ORIROTA är aktiv kan vridvinkeln eller vridvektorn programmeras enligt de fyra möjliga typerna på följande sätt:

1. Direkt positionerna för de roterande axlarna A , B , C
2. Eulervinkel (i grader) via $A2$, $B2$, $C2$
3. RPY-vinkel (i grader) via $A2$, $B2$, $C2$
4. Riktningvektor via $A3$, $B3$, $C3$ (vridvinkel med hjälp av $THETA=<Wert>$)

Om ORIOTR oder ORIOTT är aktiva kan vridvinkeln endast programmeras direkt med $THETA$.

En vridning kan också programmeras ensam i ett block utan att en orienteringsändring äger rum. Därvid har ORIOTR och ORIOTT ingen betydelse. I detta fall interpreteras vridvinkeln alltid i relation till den absoluta riktningen (ORIROTA).

```
N... ORIROTA           Fastlägga interpolering av vridvektorn
N... ORIROT
N... ORIROT
N... ORIROT
N... A3= B3= C3= THETA=<Wert> Fastlägga vridning av orienteringsvektorn
N... PO[THT]=(d2, d3, d4, d5) Interpolera vridvinkel med polynom 5:e graden
```

Betydelse

ORIROTA:	Vridvinkel för en absolut föreskriven vridriktning
ORIROT:	Vridvinkel relativ till planet mellan start- och slutorientering
ORIROT:	Vridvinkel som tangentiell vridvektor för orienteringsändringen
ORIROT:	Vridvinkel som tangentiell vridvektor för bantangenten
THETA:	Vridning av orienteringsvektorn
THETA=<Wert>:	Vridvinkel i grader som uppnås vid blockslutet

7.4 Vridningar av verktygsorienteringen (ORIOTA, ORIOTR, ORIOTT, ORIOTC, THETA)

THETA= Θ_e :	Vridvinkel med slutvinkel Θ_e för vridvektorn
THETA=AC (...):	Koppla om blockvis till absolut måttuppgift
THETA=AC (...):	Koppla om blockvis till kedjemåttuppgift
Θ_e :	Slutvinkel för vridvektorn både absolut med G90 och även relativt med G91 (kedjemåttuppgift) är aktiv
PO[THT]=(...):	Polynom för vridvinkeln

Exempel: Vridningar av orienteringarna

Programkod	Kommentar
N10 TRAORI	; Aktivera orienteringstransformation.
N20 G1 X0 Y0 Z0 F5000	; Orientering av verktyget
N30 A3=0 B3=0 C3=1 THETA=0	; i Z-riktning med vridvinkel 0
N40 A3=1 B3=0 C3=0 THETA=90	; i X-riktning och vridning med 90 grader
N50 A3=0 B3=1 C3=0 PO[THT]=(180,90)	; Orientering:
N60 A3=0 B3=1 C3=0 THETA=IC(-90)	; i Y-riktning och vridning till 180 grader
N70 ORIOTT	; förblir konstant och vridning till 90 grader
N80 A3=1 B3=0 C3=0 THETA=30	; Vridvinkel relativ till orienteringsändring
	; Vridvektor i vinkeln 30 grader till X-Y planet

Vid interpoleringen av block N40 interpoleras vridvinkeln linjärt från startvärdet 0 grader till slutvärdet 90 grader. I blocket N50 ändras vridvinkeln från 90 grader till 180 grader enligt parabeln $\theta(u) = +90u^2$. I N60 kan också en vridning utföras utan att en orienteringsändring äger rum.

Vid N80 vrids verktygsorienteringen från Y-riktningen till X-riktningen. Därvid ligger orienteringsändringen i X-Y planet och vridvektorn bildar en vinkel på 30 grader till detta plan.

Ytterligare information

ORIOTA

Vridvinkeln THETA interpoleras relativt en absolut fastlagd riktning i rymden. Den grundläggande vridriktningen sker via maskindata

ORIOTR

Vridvinkeln THETA interpreteras relativt till det plan som bildas av start- och slutorienteringen.

ORIOTT

Vridvinkeln THETA interpreteras relativt till orienteringsändringen. För THETA=0 interpoleras vridvektorn tangentiellt till orienteringsändringen och skiljer sig sedan från ORIOTR endast när minst ett polynom för "Vippvinkeln PSI" programmerades för orienteringen. Därur resulterar en orienteringsändring som inte förlöper i planet. Genom en extra programmerad vridvinkel THETA kan då t.ex. vridvektorn interpoleras så att den alltid bildar ett bestämt värde till orienteringsändringen.

*7.4 Vridningar av verktygsorienteringen (ORIROTA, ORIROT, ORIROT, ORIROT, THETA)***ORIROT**

Vridvektorn interpoleras relativt till bantangenten med en genom vinkeln `THETA` programmerbar offset. För offsetvinkeln kan därvid också ett polynom `PO[THT] = (c2, c3, c4, c5)` maximalt 5:e graden programmeras.

7.5 Banrelativa orienteringar

7.5.1 Orienteringstyper relativt till banan

Med denna utvidgade funktion uppnås den relativa orienteringen inte endast vid blockslutet utan över hela banförloppet. Den i det föregående blocket uppnådda orienteringen överförs med hjälp av storcirkelinterpolering till den programmerade slutorienteringen. Principiellt finns det två möjligheter att programmera den önskade orienteringen relativt till banan:

1. Verktygsorienteringen och även vridningen av verktyget interpoleras med ORIPATH, ORPATHS relativt till banan.
2. Orienteringsvektorn programmeras och interpoleras såsom hittills vanligt. Med ORIOTC utförs vridningen av orienteringsvektorn relativt till bantangenten.

Syntax

Interpoleringstypen för orienteringen och vridningen av verktyget programmeras med:

N... ORIPATH	Banrelativ orientering
N... ORIPATHS	Banrelativ orientering med jämnning av orienteringsförloppet
N... ORIOTC	Banrelativ interpolering av vridvektorn

En genom ett hörn i banförloppet förorsakad brytning i orienteringen kan jämnas med ORIPATHS. Riktningen och väglängden för lyftningsrörelsen programmeras genom vektorn med komponenterna A8=X, B8=Y C8=Z .

Med ORIPATH/ORIPATHS kan olika referenser till bantangenten programmeras via de tre vinklarna

- LEAD= angiven försprångsvinkel relaterad till banan och ytan
- TILT= angiven sidledsvinkel relaterad till banan och ytan
- THETA= vridvinkel

för hela banförloppet. Till vridvinkeln THETA kan med PO[THT] = (. . .) extra polynom maximalt 5:e graden programmeras.

Märk

Maskintillverkare

Följ anvisningarna från maskintillverkaren. Via projekterbara maskin- och settingdata kan ytterligare inställningar göras till den banrelativa orienteringstypen. Ytterligare förklaringar se

Litteratur:

/FB3/ Funktionshandbok Specialfunktioner; 3- till 5-axel-transformation (F2),
Kapitel "Orientering"

Betydelse

Interpoleringen av vinklarna `LEAD` och `TILT` är olika inställbar via maskindatum:

- Den med `LEAD` och `TILT` programmerade referensen för verktygsorienteringen respekteras över hela blocket.
- Försprångsvinkel `LEAD`: Vridning runt riktningen vinkelrätt mot tangenten och normalvektorn `TILT`: Vridning av orienteringen runt normalvektorn.
- Försprångsvinkel `LEAD`: Vridning runt riktningen vinkelrätt mot tangenten och normalvektorn sidledsvinkel `TILT`: Vridning av orienteringen runt riktningen för bantangenten.
- Vridvinkel `THETA`: Vridning av verktyget runt sig självt med en extra tredje roterande axel som orienteringsaxel vid sex-axel-transformation.

Märk

Banrelativ orientering tillsammans med `OSC`, `OSS`, `OSSE`, `OSD`, `OST` otillåten

Den banrelativa orienteringsinterpoleringen `ORIPATH` resp. `ORIPATHS` och `ORIOTC` kan inte programmeras tillsammans med jämning av orienteringsförloppet med ett av G-kommandona ur grupp 34. Härför måste `OSOF` vara aktiv.

7.5.2 Banrelativ vridning av verktygsorienteringen (`ORIPATH`, `ORIPATHS`, vridvinkel)

Vid en sex-axel-transformation kan förutom verktygsorienteringen också verktyget vridas runt sig självt valfritt i rummet med en tredje roterande axel. Vid banrelativ vridning av verktygsorienteringen med `ORIPATH` resp. `ORIPATHS` kan den extra vridningen programmeras via vridvinkeln `THETA`. Alternativt härtill kan vinklarna `LEAD` och `TILT` programmeras genom en vektor som ligger i planet vinkelrätt mot verktygsriktningen.

Maskintillverkare

Följ anvisningarna från maskintillverkaren. Via maskindatum kan interpoleringen av vinklarna `LEAD` och `TILT` ställas in olika.

Syntax

Vridning av verktygsorienteringen och verktyget

Verktygsorienteringstypen relativ till banan aktiveras med `ORIPATH` eller `ORIPATHS`.

N... `ORIPATH`

Aktivera orienteringstyp relaterad till banan

N... `ORIPATHS`

Aktivera orienteringstyp relaterad till banan med jämning av orienteringsförloppet

Aktivering av de tre möjliga vinklarna med vridverkan:

N... `LEAD=`

Vinkel för den programmerade orienteringen relativ till ytnormalvektorn

N... TILT=

Vinkel för den programmerade orienteringen i planet vinkelrätt mot bantangenten relativ till ytnormalvektorn

N... THETA=

Vridvinkel relativ till orienteringsändringen runt verktygsriktningen för den tredje roterande axeln

Värdena för vinklarna i blockslutet programmeras med LEAD=Wert, TILT=Wert resp. THETA=Wert. Förutom de konstanta vinklarna kan för alla tre vinklarna polynom maximalt 5:e graden programmeras.

N... PO[PHI]=(a2, a3, a4, a5)

Polynom för försprångsvinkeln LEAD

N... PO[PSI]=(b2, b3, b4, b5)

Polynom för sidledsvinkeln TILT

N... PO[THT]=(d2, d3, d4, d5)

Polynom för vridvinkeln THETA

Vid programmeringen kan de högre polynomkoefficienterna som är noll utelämnas. Exempel PO[PHI]=a2 ger för försprångsvinkeln LEAD en parabel.

Betydelse

Banrelativ verktygsorientering

ORIPATH:	Verktygsorientering relaterad till banan
ORIPATHS:	Verktygsorientering relaterad till banan, brytning i orienteringsförloppet jämnas ut
LEAD:	Vinkel relativ till ytnormalvektorn i det av bantangenten och ytnormalvektorn bildade planet
TILT:	Vridning av orienteringen runt Z-riktningen resp. vridning runt bantangenten
THETA:	Vridning runt verktygsriktningen mot Z
PO[PHI]:	Orienteringspolynom för försprångsvinkeln LEAD
PO[PSI]:	Orienteringspolynom för sidledsvinkeln TILT
PO[THT]:	Orienteringspolynom för vridvinkeln THETA

Märk

Vridvinkel THETA

För vridningen av verktyget med tredje roterande axel som orienteringsaxel runt sig självt är en sex-axel-transformation erforderlig.

7.5.3 Banrelativ interpolering för verktygsvidningen (ORIOTC, THETA)

Interpolering med vridvektorer

Förutom med ORIOTC programmerad vridning av verktyget relativt till bantangenten kan vridvektorn också interpoleras med en genom vridvinkeln THETA programmerbar offset.

Därvid kan för offsetvinkeln ett polynom upp till maximalt 5:e graden programmeras med PO[THT].

Syntax

N... ORIOTC	Utföra vridning av verktyget relativt till bantangenten
N... A3= B3= C3= THETA=Wert	Fastlägga vridning av orienteringsvektorn
N... A3= B3= C3= PO[THT]=(c2, c3, c4, c5)	Offsetvinkel med polynom maximalt 5:e graden

En vridning kan också programmeras ensam i ett block utan att en orienteringsändring äger rum.

Betydelse

Banrelativ interpolering av vridningen av verktyget vid sex-axel-transformation

ORIOTC:	Ansätta tangentiell vridvektor till bantangenten
THETA=Wert:	Vridvinkel i grader som uppnås vid blockslutet
THETA= θ_e :	Vridvinkel med slutvinkel Θ_e för vridvektorn
THETA=AC (...):	Koppla om blockvis till absolut måttuppgift
THETA=IC (...):	Koppla om blockvis till kedjemåttuppgift
PO[THT]=(c2, c3, c4, c5):	Offsetvinkel med polynom 5:e graden

Märk

Interpolering av vridvektorn ORIOTC

Ska mot orienteringsriktningen för verktyget också vridningen av verktyget relativt till bantangenten ställas in, då är detta endast möjligt vid en sex-axel-transformation.

Vid aktiv ORIOTC

Vridvektorn ORIOTA kan inte programmeras. I händelse av en programmering matas LARM 14128 "Absolutprogrammering för verktygsvridning vid aktiv ORIOTC" ut.

Orienteringsriktning för verktyget vid tre- till fem-axel-transformation

Orienteringsriktningen för verktyget kan som vid tre- till fem-axel-transformation programmeras på vanligt sätt via Eulervinkel resp. RPY-vinkel eller riktningsvektorerna. Också är orienteringsändringar för verktyget i rymden genom programmering av storcirkelinterpoleringen ORIVECT, den linjära interpoleringen av orienteringsaxlarna ORIAXES, alla interpoleringar på en konisk mantelyta ORICONxx samt tilläggsinterpoleringen till rymdkurvan med två kontaktpunkter för verktyget ORICURVE möjligt.

G....:	Uppgift om rörelsetyp för de roterande axlarna
X, Y, Z:	Angivande av linjäraaxlarna

ORIXES:	Linjär interpolering av maskin- eller orienteringsaxlar
ORIVECT:	Storcirkelinterpolering (identisk med ORIPLANE)
ORIMKS:	Vridning i maskinkoordinatsystemet
ORIWKS:	Vridning i arbetsstyckskoordinatsystemet
	Beskrivning se Kap. Vridningar av verktygsorienteringen
A= B= C=:	Programmering av maskinaxelposition
ORIEULER:	Orienteringsprogrammering via Euler-vinkel
ORIRPY:	Orienteringsprogrammering via RPY-vinkel
A2= B2= C2=:	Vinkelprogrammering virtuella axlar
ORIVIRT1:	Orienteringsprogrammering via virtuella orienteringsaxlar
ORIVIRT2:	(Definition 1), fastläggande enligt MD \$MC_ORIAX_TURN_TAB_1 (Definition 2), fastläggande enligt MD \$MC_ORIAX_TURN_TAB_2
A3= B3= C3=:	Riktningsvektorprogrammering för riktningsaxeln
ORIPLANE:	Interpolering i planet (storcirkelinterpolering)
ORICONCW:	Interpolering på en konisk mantelyta medurs
ORICONCCW:	Interpolering på en konisk mantelyta moturs
ORICONTO:	Interpolering på en koniska mantelyta tangentiell övergång
A6= B6= C6=:	Programmering av vridaxlen till konen (normerad vektor)
NUT=winkel	Öppningsvinkel för konen i grader
NUT=+179	Förflyttningsvinkel mindre eller lika med 180 grader
NUT=-181	Förflyttningsvinkel större eller lika med 180 grader
ORICONIO:	Interpolering på en konisk mantelyta
A7= B7= C7=:	Intermediär orientering (programmering som normerad vektor)
ORICURVE XH YH ZH t.ex. med polynom $PO[XH]=(x_e,$ $x_2, x_3, x_4, x_5)$	Interpolering av orienteringen med uppgift om rörelsen för två kontaktpunkter till verktyget. Förutom respektive slutpunkter är extra rymdkurvepolynom programmerbara.

Märk

Interpoleras verktygsorienteringen med aktiv ORIXES via orienteringsaxlarna då uppfylls den banrelativa inställningen av vridvinkeln endast vid blockslutet.

7.5.4 Jämning av orienteringsförloppet (ORIPATHS A8=, B8=, C8=)

Vid accelerationskontinuerliga orienteringsändringar på konturen är avbrott i banrörelserna, som kan uppträda speciellt i ett hörn på konturen, oönskade. Den brytning i orienteringsförloppet som uppstår härigenom kan jämnas genom infogande av ett eget intermediärt block. Orienteringsändringen sker sedan accelerationskontinuerligt, när också ORIPATHS är aktiv under omorienteringen. I denna fas kan en lyftningsrörelse för verktyget genomföras.

Maskintillverkare

Följ anvisningarna från maskintillverkaren för eventuellt fördefinierade maskindata och settingdata med vilka denna funktion aktiveras.

Via maskindatum är inställbart hur lyftningsvektorn interpreteras:

1. I verktygskoordinatsystemet definieras Z-koordinaten genom verktygsriktningen.
2. I arbetsstyckskoordinatsystemet definieras Z-koordinaten genom det aktiva planet.

Ytterligare förklaringar till funktionen "Banrelativ orientering" se

Litteratur:

Funktionshandbok Specialfunktioner; Fleraxeltransformationer (F2)

Syntax

För kontinuerliga verktygsorienteringar relaterade till hela banan är ytterligare programmeringsuppgifter erforderliga vid ett hörn på konturen. Riktningen och väglängden för denna rörelse programmeras genom vektorn med komponenterna A8=X, B8=Y C8=Z.

```
N... ORIPATHS A8=X B8=Y C8=Z
```

Betydelse

ORIPATHS:	Verktygsorientering relaterad till banan, en brytning i orienteringsförloppet jämnas ut.
A8= B8= C8=:	Vektorkomponenter för riktning och väglängd
X, Y, Z:	Lyftningsrörelse i verktygsriktning

Märk

Programmering av riktningsvektorn A8, B8, C8

Är längden för denna vektor lika med noll sker ingen lyftningsrörelse.

ORIPATHS

Den banrelaterade verktygsorienteringen blir aktiv med ORIPATHS. Annars överförs orienteringen med hjälp av linjär storcirkelinterpolering från start- till slutorienteringen.

7.6 Komprimering av orienteringen (COMPON, COMPCURV, COMPCAD, COMPSURF)

NC-program i vilka en orienteringstransformation (TRAORI) är aktiv och verktygsorienteringen är programmerad (likgiltigt vilken typ) kan komprimeras under respekterande av föreskrivna toleranser.

Programmering

Verktygsorientering

Om en orienteringstransformation (TRAORI) är aktiv kan vid 5-axel maskiner verktygsorienteringen programmeras på följande sätt (kinematikoberoende):

- Programmering av riktningsektorn via:
A3=<...> B3=<...> C3=<...>
- Programmering av Eulervinkel resp. RPY-vinkel via:
A2=<...> B2=<...> C2=<...>

Vridning av verktyget

Vid **6-axel** maskiner kan programmeras förutom för verktygsorienteringen också för vridningen av verktyget.

Programmeringen av vridvinkeln sker med:

THETA=<...>

Se " Vridningar av verktygsorienteringen (Sida 345) ".

Märk

NC-block i vilka dessutom en vridning är programmerad kan komprimeras endast om vridvinkeln ändrar sig **linjärt**. Dvs. för vridvinkeln får inget polynom vara programmerat med PO[THT]=(...).

Allmän form för ett komprimerbart NC-block

Den allmänna formen för ett komprimerbart NC-block kan därför se ut på följande sätt:

N... X=<...> Y=<...> Z=<...> A3=<...> B3=<...> C3=<...> THETA=<...> F=<...>

resp.

N... X=<...> Y=<...> Z=<...> A2=<...> B2=<...> C2=<...> THETA=<...> F=<...>

Märk

Positionsvärdena kan anges direkt (t.ex. X90) eller indirekt via parametertillordningar (t.ex. X=R1*(R2+R3)).

Programmering av verktygsorienteringen genom positioner för roterande axlar

Verktygsorienteringen kan också vara angiven genom positionerna för roterande axlar t.ex. i formen:

7.6 Komprimering av orienteringen (COMPON, COMPCURV, COMPCAD, COMPSURF)

N... X=<...> Y=<...> Z=<...> A=<...> B=<...> C=<...> THETA=<...> F=<...>

I detta fall genomförs komprimeringen på två olika sätt beroende på om en storcirkelinterpolering genomförs eller inte. När ingen storcirkelinterpolering äger rum så framställs den komprimerade orienteringsändringen genom axiella polynom för de roterande axlarna på vanligt sätt.

Konturprecision

Beroende på den inställda kompressionsmoden (MD20482 \$MC_COMPRESSOR_MODE) blir för geometriaxlarna och orienteringsaxlarna vid komprimeringen antingen de projekterade axelspecifika toleranserna (MD33100 \$MA_COMPRESS_POS_TOL) eller de följande via settingdata inställbara kanalspecifika toleranserna verksamma:

SD42475 \$SC_COMPRESS_CONTUR_TOL (maximal konturavvikelse)

SD42476 \$SC_COMPRESS_ORI_TOL (maximal vinkelavvikelse för verktygsorienteringen)

SD42477 \$SC_COMPRESS_ORI_ROT_TOL (maximal vinkelavvikelse för vridvinkeln till verktyget) (disponibel endast vid 6-axel maskiner)

Litteratur:

Funktionshandbok Grundfunktioner; 3- till 5-axel-transformation (F2),
Kapitel: "Komprimering av orienteringen"

Aktivering / inaktivering

Kompressor-funktioner kopplas till genom de modala G-koderna COMPON, COMPCURV, COMPCAD resp. COMPSURF.

Kompressor-funktionen avslutas med COMPOF.

Se " NC-block-kompression (COMPON, COMPCURV, COMPCAD) (Sida 255) ".

Märk

Orienteringsrörelsen komprimeras endast vid aktiv storcirkelinterpolering (dvs. ändringen i verktygsorienteringen sker i det plan som bildas av start- och slutorienteringen).

En storcirkelinterpolering genomförs under följande villkor:

- MD21104 \$MC_ORI_IPO_WITH_G_CODE = 0,
ORIWKS är aktiv och
orienteringen är programmerad med hjälp av vektorer (med A3, B3, C3 resp. A2, B2, C2).
 - MD21104 \$MC_ORI_IPO_WITH_G_CODE = 1 och
ORIVECT resp. ORIPLANE är aktiv.
Verktygsorienteringen kan vara programmerad antingen som riktningvektor eller med positioner för roterande axlar. Är ett av G-kommandona ORICONxx eller ORICURVE aktivt eller är polynom programmerade för orienteringsvinklarna (PO[PHI] och PO[PSI]) genomförs ingen storcirkelinterpolering.
-

Exempel

I det följande programexemplet komprimeras en cirkel som närmas med ett polygontåg. Verktygsorienteringen flyttar sig därvid synkront därtill på en konisk mantel. Fastän de efter varandra programmerade orienteringsändringarna ej förlöper kontinuerligt, genererar kompressor-funktionen ett jämnt förlopp av orienteringen.

Programmering	Kommentar
DEF INT ANZAHL=60	
DEF REAL RADIUS=20	
DEF INT COUNTER	
DEF REAL WINKEL	
N10 G1 X0 Y0 F5000 G64	
\$SC_COMPRESS_CONTUR_TOL=0.05	; Maximal avvikelse för konturen = 0,05 mm
\$SC_COMPRESS_ORI_TOL=5	; Maximal avvikelse för orienteringen = 5 grader
TRAORI	
COMPCURV	; Det körs ingen cirkel som bildas av polygo- ner. Orienteringen flyttar sig därvid på en kon runt Z-axeln med en öppningsvinkel på 45 grader.
N100 X0 Y0 A3=0 B3=-1 C3=1	
N110 FOR COUNTER=0 TO ANZAHL	
N120 WINKEL=360*COUNTER/ANZAHL	
N130 X=RADIUS*cos(WINKEL) Y=RADIUS*sin(WINKEL)	
A3=sin(WINKEL) B3=-cos(WINKEL) C3=1	
N140 ENDFOR	

7.7 Koppla till/från jämning av orienteringsförloppet (ORISON, ORISOF)

"Jämning av orienteringsförloppet" kopplas till/från i detaljprogrammet genom kommandona i G-gruppen 61. Kommandona är modalt verksamma.

Förutsättningar

- System med 5/6-axeltransformation.
- Kompressor-funktion COMPCAD är aktiv.

Syntax

```
ORISON
...
ORISOF
```

Betydelse

ORISON:	Koppla till jämning av orienteringsförloppet
ORISOF:	Koppla från jämning av orienteringsförloppet

Exempel

Programkod	Kommentar
...	
TRAORI()	; Tillkoppling av orienteringstransformationen.
COMPCAD	; Tillkoppling av kompressor-funktion COMPCAD.
ORISON	; Tillkoppling av orienteringsjämningen.
\$SC_ORISON_TOL=1.0	; Maximal vinkelavvikelse för verktygsorienteringen = 1,0 grader.
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	; Frånkoppling av orienteringsjämningen.
...	

7.7 Koppla till/från jämning av orienteringsförloppet (ORISON, ORISOF)

Orienteringen svängs med 90 grader i XZ-planet från -45 till +45 grader. Genom jämnings av orienteringsförloppet uppnår orienteringen inte längre de maximala vinkelvärdena från -45 resp. +45 grader.

7.8 Kinematisk transformation

7.8.1 Koppla till frontsidtransformation (TRANSMIT)

Frontsidtransformationen (TRANSMIT) koppas till i detaljprogrammet eller synkronaktion via anvisningen TRANSMIT.

Syntax

```
TRANSMIT
TRANSMIT (<n>)
```

Betydelse

TRANSMIT:	Koppla till TRANSMIT med första TRANSMIT-datablocket
TRANSMIT (n):	Koppla till TRANSMIT med <n>-te TRANSMIT-datablocket

Märk

En i kanalen aktiv transformation TRANSMIT kopplas från genom:

- Koppla från transformation: TRAFOOF
- Tillkoppling av en annan transformation: t.ex. TRACYL, TRAANG, TRAORI

7.8.2 Koppla till cylindermanteltransformation (TRACYL)

Cylindermanteltransformationen (TRACYL) koppas till i detaljprogrammet eller synkronaktion via anvisningen TRACYL.

Syntax

```
TRACYL (<d>)
TRACYL (<d>, <n>)
TRACYL (<d>, <n>, <k>)
```

Betydelse

TRACYL (<d>):	; Koppla till TRACYL med det första TRACYL--datablocket och arbetsdiameter <d>
TRACYL (<d>, <n>):	Koppla till TRACYL med det <n>-te TRACYL--datablocket och arbetsdiameter <d>
<d>:	Referens- resp. arbetsdiameter Värdet måste vara större än 1.

<n>:	TRACYL-datablocknummer (option)	
	Värdeområde:	1, 2
<k>:	Parametern <k> är endast relevant för transformationstyp 514	
	k = 0:	utan spårväggskompensering
	k = 1:	med spårväggskompensering
	Anges inte parametern, verkar det parameterade grundläget: \$MC_TRACYL_DEFAULT_MODE_<n> mit <n> = TRACYL-datablocknummer	

Märk

En i kanalen aktiv transformation TRACYL kopplas från genom:

- Koppla från transformation: TRAFOOF
- Tillkoppling av en annan transformation: t.ex. TRAANG, TRANSMIT, TRAORI

Exempel

Programkod	Kommentar
...	
N40 TRACYL(40.)	; Koppla till TRACYL med det första TRACYL--datablocket och arbetsdiameter 40 mm.
...	

Ytterligare information**Programuppbyggnad**

Ett detaljprogram för att fräsa ett spår med TRACYL-transformation 513 (TRACYL med spårväggskompensering) består som regel av följande steg:

1. Välja verktyg.
2. Välja TRACYL.
3. Välja passande koordinatförflyttning (FRAME).
4. Positionering.
5. Programmera OFFN.
6. Välja WRK.
7. Framkörningsblock (inkörning av WRK och framkörning till spårväggen).
8. Kontur för spårmittlinjen.
9. Välja bort WRK.
10. Bortkörningsblock (urkörning av WRK och bortkörning från spårväggen).
11. Positionering.

12. TRAFOOF.

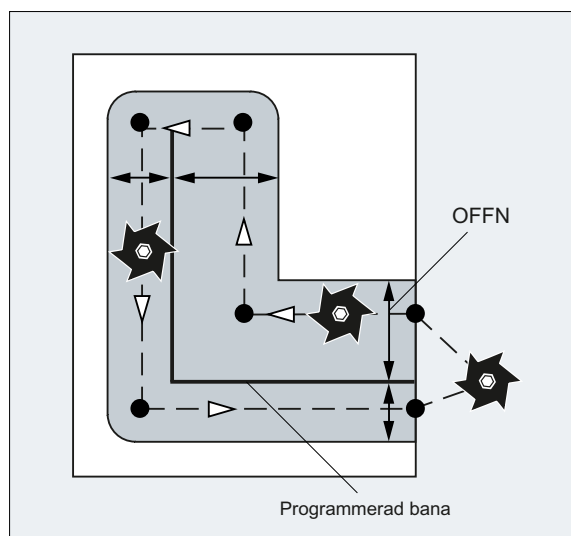
13. Åter välja ursprunglig koordinatförflyttning (FRAME).

Kontur-offset (OFFN)

För att fräsa spår med TRACYL-transformation 513 programmeras i detaljprogrammet spårets mittlinje och via adressen OFFN **den halva spårbredden**.

För att undvika en skada på spårväggen blir OFFN verksam först med vald verktygsradiekompensering.

En ändring av OFFN inom detaljprogrammet är möjlig. Därmed kan spårmittlinjen förflyttas från mitten:



Märk

OFFN bör vara minst så stor som verktygsradien för att utesluta en skada på spårväggen som ligger mittemot.

Märk

OFFN med TRACYL verkar på ett annat sätt än utan TRACYL. Eftersom OFFN även utan TRACYL räknas in vid aktiv WRK, bör OFFN efter TRAFOOF återställas in på noll.

OBSERVERA

Verkan av OFFN är beroende av transformationstypen

Vid TRACYL-transformationen 513 (TRACYL med spårväggskompensering) programmeras för OFFN den halva spårbredden.

Vid TRACYL-transformationen 512 (utan spårväggskompensering) verkar värdet för OFFN däremot som avmätt för WRK.

Verktysradiekompensering (WRK)

Vid TRACYL-transformation 513 räknas WRK inte in relativt till spårväggen utan till den programmerade mitten för spåret. För att verktyget ska gå till vänster om spårväggen ska i stället för G41 anvisningen G42 programmeras eller värdet för OFFN anges med negativt förtecken.

Verktysdiameter

Med TRACYL och ett verktyg vars diameter är mindre än spårbredden, skapas inte samma spårväggsgeometri som med ett verktyg vars diameter är lika med spårbredden. Till förbättring av noggrannheten rekommenderas att välja verktysdiameter endast något mindre än spårbredden.

Axelanvändning**Märk**

Följande axlar kan inte användas som positioneringsaxel resp. pendelaxel:

- geometriaxeln i cylindermantelytans perifeririktning (Y-axel)
- den extra linjärxeln vid spårväggskompensering (Z-axeln)

7.8.3 Koppla till snedvinkeltransformation med programmerbar vinkel (TRAANG)

Snedvinkeltransformationen koppas till med programmerbar vinkel i detaljprogrammet eller synkronaktion via anvisningen TRAANG.

Syntax

```
TRAANG
TRAANG ( )
TRAANG ( , <n> )
TRAANG ( <α> )
TRAANG ( <α> , <n> )
```

Betydelse

TRAANG: TRAANG () :	Koppla till TRAANG med det första TRAANG-datablocket och den sist giltiga vinkelnl <α>
TRAANG (, <n>) :	Koppla till TRAANG med det <n>-te TRAANG-datablocket och den sist giltiga vinkelnl <α>
TRAANG (<α>) :	Koppla till TRAANG med det första TRAANG-datablocket och vinkelnl <α>
TRAANG (<α> , <n>) :	Koppla till TRAANG med det <n>-te TRAANG-datablocket och vinkelnl <α>

< α >:	Vinkel för den lutande axeln (option)	
	Värdeområde:	$-90^\circ < \alpha < +90^\circ$
	Anges ingen vinkel, verkar den i maskindatumet parametrerade grundinställningen: MD2xxxx \$MC_TRAANG_ANGLE_<n>	
<n>:	TRAANG-datablocknummer (option)	
	Värdeområde:	1, 2

Märk

En i kanalen aktiv snedvinkeltransformation TRAANG kopplas från genom:

- Koppla från transformation: TRAFOOF
- Tillkoppling av en annan transformation: t.ex. TRACYL, TRANSMIT, TRAORI

Exempel

Programkod	Kommentar
N20 TRAANG(45)	; Koppla till TRAANG med det första TRAANG-datablocket och vinkel 45°.

7.8.4 Sned instickning på slipmaskiner (G5, G7)

G-kommandona G7 och G5 tjänar till förenkling av programmeringen för den sneda instickningen på slipmaskiner med transformationen "Lutande axel" (TRAANG), så att vid instickningen endast den lutande axeln flyttar sig.

Endast den önskade ändpositionen för insticksrörelsen i X och Z ska programmeras. Den tillhörande startpositionen beräknas och uppsöks av NC vid G7, utgående från den aktuella positionen för X-axeln, den programmerade ändpositionen och vinkeln α för den lutande axeln.

Startpositionen resulterar ur skärningspunkten mellan de båda räta linjerna:

- Rät linje parallell till Z-axeln på avståndet för den aktuella positionen för X-axeln
- Rät linje parallell till den lutande axeln genom den programmerade ändpositionen

Med den efterföljande G5 förflyttas den lutande axeln till den programmerade ändpositionen.

Syntax

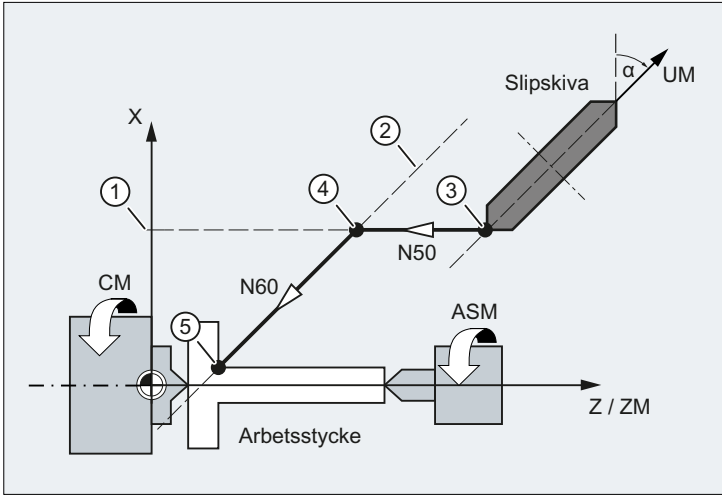
```
G7 <Endpos_X> <Endpos_Z>
G5 <Endpos_X>
```

Betydelse

G7:	Beräkna och uppsöka startposition till den sneda instickningen
G5:	Förflytta den lutande axeln till den programmerade ändpositionen

<Endpos_X>:	Ändposition för X-axeln
<Endpos_Z>:	Ändposition för Z-axeln

Exempel



- ① Parallell till Z-axeln på avståndet för den aktuella positionen för X-axeln
- ② Parallell till den lutande axeln genom den programmerade ändpositionen
- ③ Utgångsposition
- ④ Instickning: Startposition
- ⑤ Instickning: Ändposition
- X geometriaxel
- Z geometriaxel
- ZM maskinaxel
- UM maskinaxel

Programkod	Kommentar
N... G18	; Välja XZ-plan
N40 TRAANG(45.0)	; Transformation TRAANG tillkopplas, vinkel = 45°
N50 G7 X40 Z70 F4000	; Beräkna och uppsöka startposition
N60 G5 X40 F100	; Förflytta lutande axel till ändposition
N70 ...	

7.9 Koppla till sammanlänkad transformation (TRACON)

En sammanlänkad transformation kopplas till i detaljprogrammet eller synkronaktion via anvisningen TRACON.

Syntax

```
TRACON (<Trafo_Nr>,<Par_1>,<Par_2>,...)
...
TRAFOOF
```

Betydelse

TRACON:	Koppla till sammanlänkad transformation En tidigare aktiverad andra transformation kopplas implicit med TRACON().		
<Trafo_Nr>:	Nummer på den sammanlänkade transformationen		
	Typ:	INT	
	Värdeområde:	0 ... 2	
	Värde:	0, 1	första/enda sammanlänkade transformation
		2	andra sammanlänkade transformation
		Ingen uppgift	likabetydande med 0 resp. 1
	Observera: värden skilda från 0, 1, 2 skapar ett fellarm.		
<Par_1>,<Par_2>,...:	Parameter för de med varandra sammanlänkade transformationerna (t. ex. vinkel för den lutande axeln) Vid ej inställda parametrar blir förinställningarna eller de sist använda parametrarna verksamma. Det måste genom att sätta komma sörjas för att de angivna parametrarna utvärderas i den ordningsföljd som de väntas när för parametrar som står tidigare förinställningarna ska verka. Speciellt måste vid angivande av minst en parameter ett komma stå före denna också när angivande av <Trafo_Nr> inte är nödvändig alltså till exempel TRACON(, 3.7). Vid TRACON med Transmit eller TRAORI (5:e axelbearbetning) verkar den andra parametern Par_2 inte som vinkel för den sneda axeln. Vid TRACON med TRACYL (mantelytbearbetning) används par för enhetsdiametern.		
TRAFOOF:	Koppla från den sist tillkopplade (sammanlänkade) transformationen		

Exempel

Programkod	Kommentar
...	

7.9 Koppla till sammanlänkad transformation (TRACON)

Programkod	Kommentar
N230 TRACON(1,45.)	; Koppla till första sammanlänkade transformation
	; Den tidigare aktiva transformationen väljs automa-
	tiskt bort.
	; Parametern för den lutande axeln är 45°.
...	
N330 TRACON(2,40.)	; Andra sammanlänkade transformation
	; Parametern för den lutande axeln är 40°.
...	
N380 TRAFOOF	; Andra sammanlänkade transformation.
...	

7.10 Kartesisk PTP-körning

7.10.1 Koppla till/från kartesisk PTP-körning (PTP, PTPG0, PTPWOC, CP)

Den kartesiska punkt-till-punkt- resp. PTP-körningen kopplas till/från i NC-programmet med kommandona i G-gruppen 49.

Kommandona är modalt verksamma. Förinställning är förflyttningen med en kartesisk banrörelse (CP).

På annat sätt än vid CP genomförs vid aktiv PTP-körning bara en transformation i den kartesiska målpunkten och maskinaxlarna förflyttas synkront.

För att den kartesiska målpunkten ska kunna entydigt omräknas till maskinaxelvärden, är förutom positions- och vinkeluppgifter ytterligare informationer nödvändiga, som markerar axellägena. Dessa uppgifter ges via de inställbara adresserna STAT (Sida 368) och TU (Sida 372).

Förutsättning

Transformation TRAORI, TRANSMIT, RCTRA eller ROBX är aktiv.

Syntax

```
PTP / PTPG0 / PTPWOC
...
CP
```

Betydelse

PTP:	Koppla till punkt-till-punkt-rörelse PTP Den programmerade kartesiska positionen i G0- och G1-block uppsöks med en synkronaxelrörelse.
PTPG0:	Koppla till punkt-till-punkt-rörelse PTPG0 Endast i G0-block uppsöks den programmerade kartesiska positionen med en synkronaxelrörelse. I G1-block kopplas om till banrörelsen CP.
PTPWOC:	Koppla till punkt-till-punkt-rörelse PTPWOC (endast möjligt vid aktiv orienteringstransformation) Liksom PTP, dock utan utjämningsrörelse, som förorsakas genom rörelserna hos roterande axlar och orienteringsaxlar.
CP:	Koppla från punkt-till-punkt-rörelse och koppla till banrörelse CP Med CP utförs en kartesisk banrörelse.

Märk**PTPWOC**

I kombination med en RCTRA- eller ROBX-transformation är användningen av PTPWOC meningsfull!

Exempel

Se:

- Exempel 1: PTP-körning av en 6-axelrobot med ROBX-transformation (Sida 375)
- Exempel 2: PTP-körning vid generisk 5-axel transformation (Sida 376)
- Exempel 3: PTPG0 och TRANSMIT (Sida 376)

7.10.2 Ge länkarnas position (STAT)

Positionsuppgiften med kartesiska koordinater och uppgiften över verktygsorienteringen räcker inte till, för att entydigt fastlägga maskinpositionen, eftersom vid lika verktygsorientering flera länkpositioner är möjliga. Beroende på om vilken kinematik det handlar existerar upp till 8 olika länkpositioner. Dessa olika länkpositioner är transformationsspecifika.

För att undvika tvetydigheter, anges länkarnas position under adressen STAT.

Märk

Styrningen tar hänsyn till programmerade STAT-värden endast vid PTP-rörelser. Vid CP-rörelser blir de ignorerade, eftersom vid förflyttning med aktiv transformation ett positionsbyte normalt inte är möjligt. Vid förflyttning med aktiv CP övertas positionen för målpunkten av startpunkten.

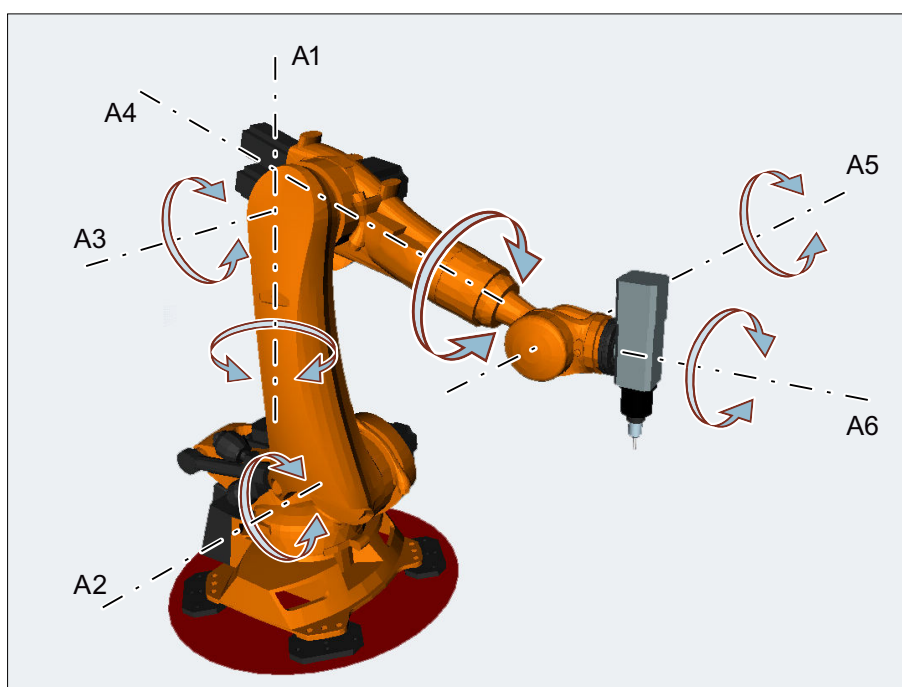
Syntax

STAT=<Wert>

Betydelse

STAT:	Inställbara adresser för angivande av länkpositionerna
<Wert>:	Binär- eller decimalvärde Innehåller för var och en av de möjliga positionerna en bit. Betydelsen för bits bestäms av respektive transformation.

Användningen av STAT ska åskådliggöras med exemplet för en 6-axlig knickarmrobot med frässpindel. Den kinematiska transformationen ska ske via robottransformationen ROBX (Förutsättning: Compile-cykel "RMCC/ROBX Transformation Robotik kompletterad" är laddad och aktiv).

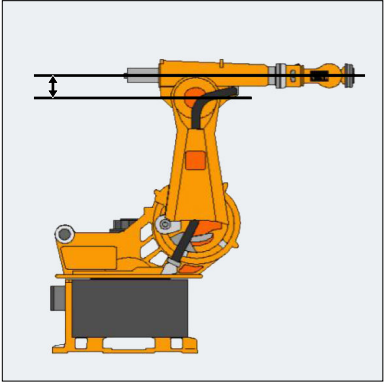


Axlarna A1, A2 och A3 är huvudaxlarna i knickarmroboten. Med huvudaxlarna positioneras axlarna A4, A5 och A6, som betecknas som huvud- eller handaxlar, i arbetsrummet. Genom de extra rörelsemöjligheterna hos handaxlarna kan frässpindeln orienteras så i rummet, som det är nödvändigt för bearbetningsuppgiften. Vid lika verktygsorientering är därvid olika ledinställningar möjliga.

Valet av de ledställningar som ska användas för bearbetningen görs genom programmering av bit 0 ... 2 på den inställbara adressen STAT:

Bit 0	Position för skärningspunkten mellan de manuella axlarna (A4, A5, A6)	
	= 0	Grundområde (Shoulder Right) Roboten befinner sig i grundområdet, när X-värdet för skärningspunkten mellan de manuella axlarna, relaterad till A1-koordinatsystemet, är positivt.
	= 1	Underuppområde (Shoulder Left) Roboten befinner sig i underuppområdet, när X-värdet för skärningspunkten mellan de manuella axlarna, relaterad till A1-koordinatsystemet, är negativt.

Exempel: Skärningspunkten för handaxlarna befinner sig i grundområdet

Bit 1	Position för axel 3	
	Vinkeln, vid vilken värdet för bit 1 ändras, är beroende av robottyp. För robot, vars axlar 3 och 4 korsar varandra, gäller:	
	= 0	A3 < 0° (Elbow Down)
	= 1	A3 ≥ 0° (Elbow Up)
Observera: För robotar med en offset mellan axel 3 och 4 är den vinkel vid vilken värdet för bit 1 ändras, beroende av storleken på denna offset.		
		Offset mellan A3 och A4
Bit 2	Position för axel 5	
	= 0	A5 ≤ 0° (No Handflip)
	= 1	A5 > 0° (Handflip)

Programexempel:

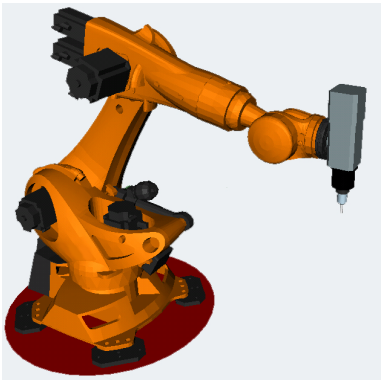
Programkod	Kommentar
...	
N14 T="T8MILLD20" D1	; \$TC_DP3[1,1]=132.95
N16 ORIMKS	
N17 G1 PTP X1665.67 Y0 Z1377.405 A=0 B=0 C=0 STAT=... F2000	; STAT-värdet bestämmer ledställningarna (se nedan).
...	

STAT=1 ('B001')

→ Shoulder Left

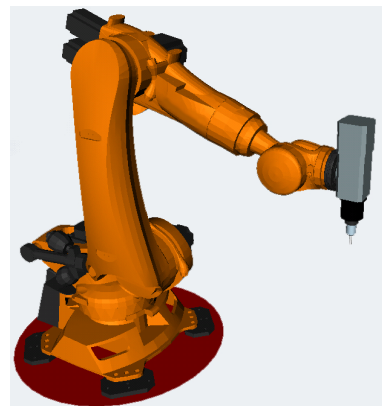
→ Elbow Down

→ No Handflip



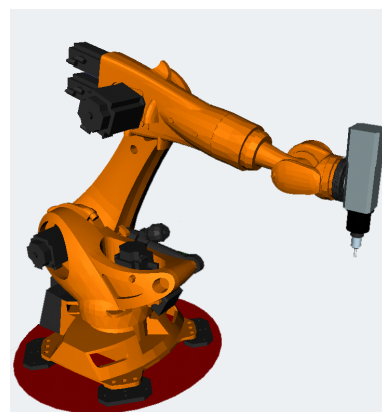
STAT=2 ('B010')

- Shoulder Right
- Elbow Up
- No Handflip



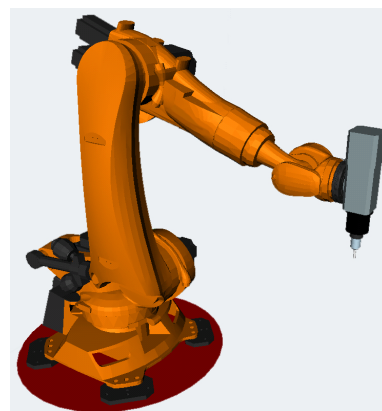
STAT=5 ('B101')

- Shoulder Left
- Elbow Down
- Handflip



STAT=6 ('B110')

- Shoulder Right
- Elbow Up
- Handflip



TRANSMIT

Vid TRANSMIT används adressen STAT för att upplösa tvetydigheten beträffande polerna.

När den roterande axeln måste rotera 180° resp. konturen vid CP skulle föra genom polen gäller:

Bit 0	Endast relevant vid \$MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 1 resp. 2:	
	= 0	Roterande axel förflyttas +180° resp. roterar medurs.
	= 1	Roterande axel förflyttas -180° resp. roterar moturs.
Bit 1	Endast relevant vid \$MC_TRANSMIT_POLE_SIDE_FIX_1/2 = 0:	
	= 0	Det körs genom polen. Den roterande axeln roterar inte.
	= 1	Det roteras runt polen. Därvid är bit 0 från STAT relevant.

7.10.3 Ange förtecken för axelvinkel (TU)

För att vid roterande axlar också kunna uppsöka axelvinklar, som är större än +180° eller mindre än -180°, utan speciell förflyttningsstrategi (t. ex. mellanpunkter), måste förtecknen för axelvinklarna anges under den inställbara adressen TU.

Märk

Styrningen tar hänsyn till programmerade TU-värden endast vid PTP-rörelser. Vid CP-rörelser ignoreras de.

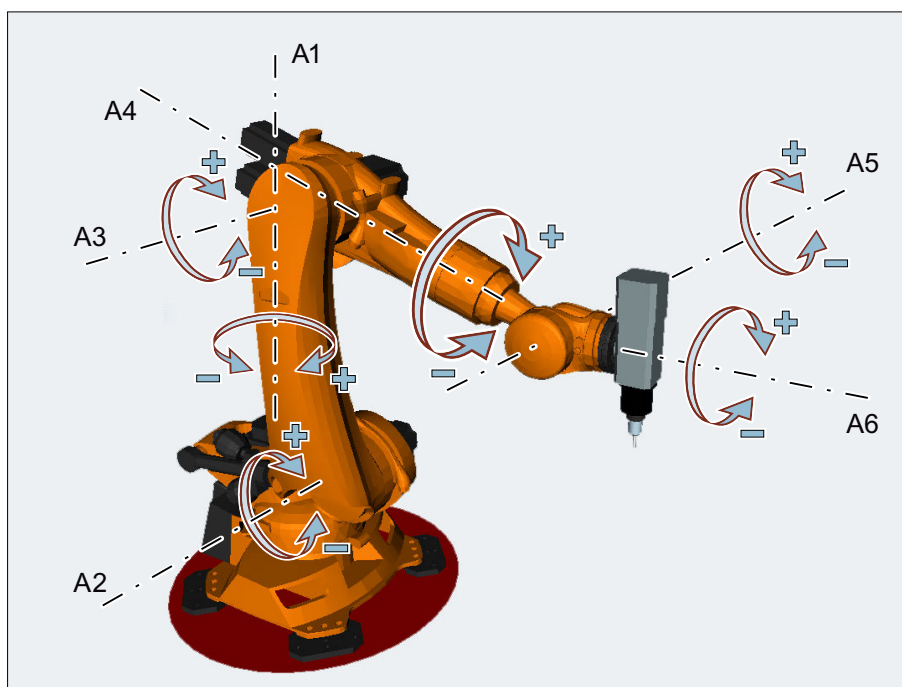
Syntax

TU=<Wert>

Betydelse

TU:	Inställbara adresser för angivande av axelvinklarnas förtecken		
<Wert>:	Binär- eller decimalvärde		
	För varje axel som ingår i transformationen finns det en bit som visar förtecknet för axelvinkeln (θ) och därmed förflyttningsriktningen.		
	Bit	= 0	Axelvinkelförtecken: + Axelvinkelområde: $0^\circ \leq \theta < 360^\circ$
		= 1	Axelvinkelförtecken: - Axelvinkelområde: $-360^\circ < \theta < 0^\circ$

Exempel: 6-axlig knickarmrobot



Bit	Betydelse	Värde	Axelvinkelförtecken	Axelvinkel
Bit 0 ¹⁾	Förtecken för axelvinkeln för A1	= 0	+	$\geq 0^\circ$
		= 1	-	$< 0^\circ$
Bit 1 ¹⁾	Förtecken för axelvinkeln för A2	= 0	+	$\geq 0^\circ$
		= 1	-	$< 0^\circ$
Bit 2 ¹⁾	Förtecken för axelvinkeln för A3	= 0	+	$\geq 0^\circ$
		= 1	-	$< 0^\circ$
Bit 3 ¹⁾	Förtecken för axelvinkeln för A4	= 0	+	$\geq 0^\circ$
		= 1	-	$< 0^\circ$
Bit 4 ¹⁾	Förtecken för axelvinkeln för A5	= 0	+	$\geq 0^\circ$
		= 1	-	$< 0^\circ$
Bit 5 ¹⁾	Förtecken för axelvinkeln för A6	= 0	+	$\geq 0^\circ$
		= 1	-	$< 0^\circ$

¹⁾ De verkliga TU-bitnumren resulterar ur kanalaxelnumren för robotaxlarna! I exemplet är robotaxlarna (A1 till A6) de första sex axlarna i kanalen, följaktligen används TU-bits 0 ... 5. Vid en annan kanalaxeltillordning för robotaxlarna skulle TU-bitnumren för robotaxlarna ändras motsvarande (t. ex.: robotaxlarna är den 3:e till 8:e kanalaxeln, dvs. TU-bits 2 ... 7 används för robotaxlarna).

TU=19 (motsvarar TU='B010011) skulle alltså betyda:

Bit	Värde		Axelvinkel
0	= 1	$\Rightarrow$	$\theta_{A1} < 0^\circ$
1	= 1	$\Rightarrow$	$\theta_{A2} < 0^\circ$
2	= 0	$\Rightarrow$	$\theta_{A3} \geq 0^\circ$

3	= 0	⇒	$\theta_{A4} \geq 0^\circ$
4	= 1	⇒	$\theta_{A5} < 0^\circ$
5	= 0	⇒	$\theta_{A6} \geq 0^\circ$

Märk

Vid axlar med ett förflyttningsområde $> \pm 360^\circ$ flyttas alltid längs den kortaste vägen, eftersom axelpositionen genom TU-informationen inte kan bestämmas entydigt.

Programmeras vid en position ingen TU, förflyttas beroende på MD30455

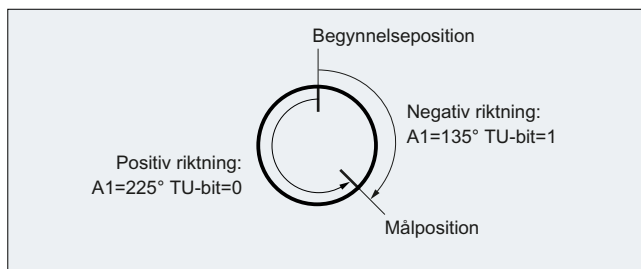
\$MA_MISC_FUNCTION_MASK den kortare eller längre vägen (se Kapitel "Hänsynstagande till SW-Limits vid PTP-körning" i Funktionshandbok Tilläggsfunktioner).

TRANSMIT

Vid PTP-körning vid aktiv TRANSMIT är adressen TU utan betydelse!

Exempel

Den i följande bild angivna roterande axelposition kan uppsökas i negativ eller i positiv riktning. Under adressen A1 programmeras vinkelpositionen. Först genom TU-uppgiften blir förflyttningsriktningen entydig.



7.10.4 Exempel 1: PTP-körning av en 6-axelrobot med ROBX-transformation

I det följande användningsexemplet visas den kartesiska PTP-körningen och de NC-kommandon som står i samband därmed som exempel.

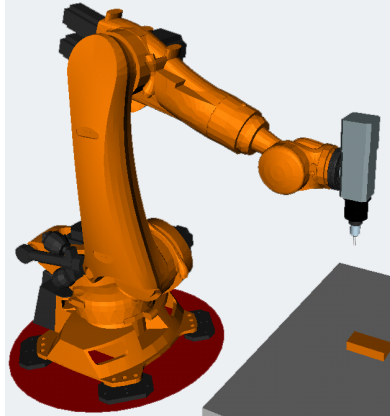


Bild 7-3 6-axlig knickarmrobot med frässpindel

```
N1 G90
N2 T="T8MILLD20" D1 M6
N3 TRAORI
; $P_UIFR[1]=CTTRANS(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
```

7.10.5 Exempel 2: PTP-körning vid generisk 5-axel transformation

Antagande: En rätvinklig CA-kinematik ligger till grund.

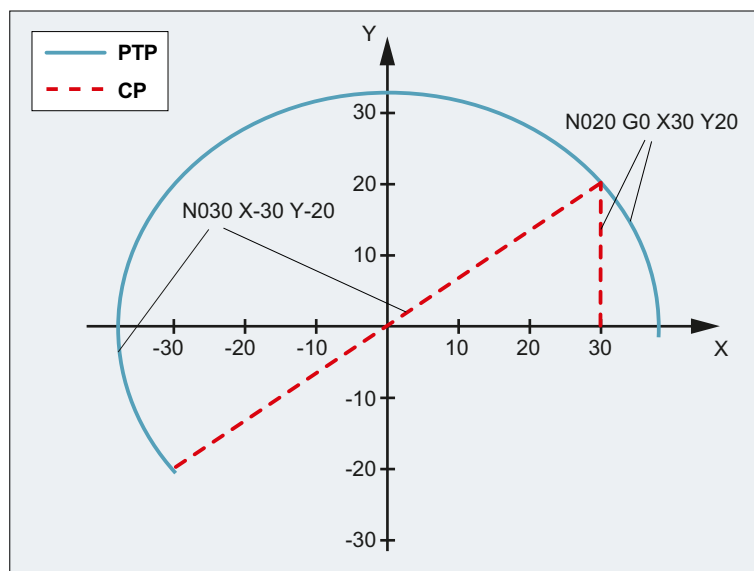
Programkod	Kommentar
TRAORI	; Transformation CA-kinematik till
PTP	; Koppla till PTP-körning
N10 A3=0 B3=0 C3=1	; Roterande axelpositioner C=0 A=0
N20 A3=1 B3=0 C3=1	; Roterande axelpositioner C=90 A=45
N30 A3=1 B3=0 C3=0	; Roterande axelpositioner C=90 A=90
N40 A3=1 B3=0 C3=1 STAT=1	; Roterande axelpositioner C=270 A=-45

Välja entydigt uppsökningsläge för roterande axelposition:

I block N40 kör därvid de roterande axlarna genom programmeringen av **STAT = 1** den längre vägen från sin startpunkt (C=90, A=90) till slutpunkten (C=270, A=-45). Med **STAT=0** skulle de roterande axlarna däremot köra den kortare vägen till ändpunkten (C=90, A=45).

7.10.6 Exempel 3: PTPG0 och TRANSMIT

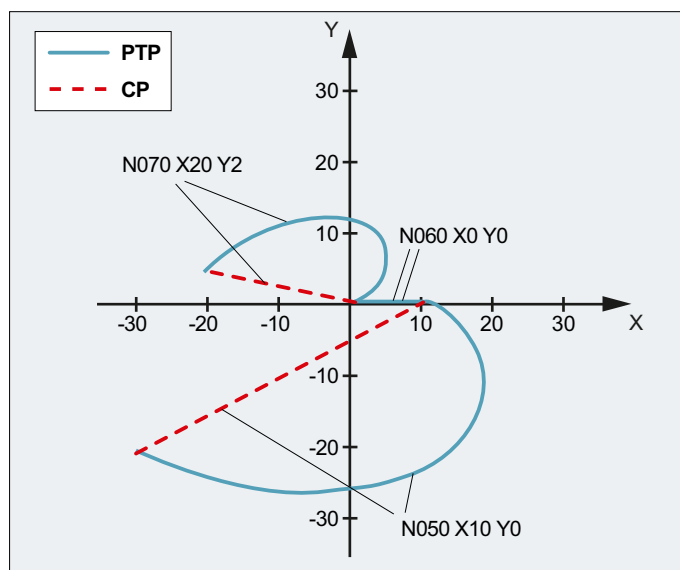
Körning runt polen med PTPG0 och TRANSMIT



Programkod	Kommentar
N001 G0 X30 Z0 F10000 T1 D1 G90	; Utgångsläge absolutmått
N002 SPOS=0	
N003 TRANSMIT	; Transformation TRANSMIT
N010 PTPG0	; Till varje G0-block automatiskt PTP och därefter åter CP
N020 G0 X30 Y20	

Programkod	Kommentar
N030 X-30 Y-20	
N120 G1 X30 Y20	
N110 X30 Y0	
M30	

Utkörning ur polen med PTPG0 och TRANSMIT



Programmering	Kommentar
N001 G0 X90 Z0 F10000 T1 D1 G90	; Utgångsläge
N002 SPOS=0	
N003 TRANSMIT	; Transformation TRANSMIT
N010 PTPG0	; Till varje G0-block automatiskt PTP och därefter åter 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	

7.11 Randvillkor vid valet av en transformation

Funktion

Valet av transformationer är möjligt via detaljprogram resp. MDA. Därvid skall följande observeras:

- Ett intermediärt rörelseblock infogas inte (faser/radier).
- En spline-blocksekvens måste vara avslutad; om inte visas ett meddelande.
- Verktygsfinkompensering måste vara bortvalt (FTOCOF); om inte visas ett meddelande.
- Verktygsradiekompensering måste vara bortvalt (G40); om inte visas ett meddelande.
- En aktiverad verktygslängdkompensering övertas av styrningen till transformationen.
- Den före transformationen verksamma aktuella framen väljs bort av styrningen.
- En aktiv arbetsfältsbegränsning väljs bort av styrningen för de axlar som berörs av transformationen (motsvarar WALIMOF).
- Skyddsområdesövervakning väljs bort.
- Banstyrningsdrift och översläpning stoppas.
- Alla i maskindatum angivna axlar måste vara blockrelaterat synkroniserade.
- Utbytta axlar byts tillbaka; om inte visas ett meddelande.
- Vid beroende axlar matas ett meddelande ut.

Verktygsväxling

En verktygsväxling är tillåten endast vid bortvald verktygsradiekompensering.

En växling av verktygslängdkompenseringen och ett till-/bortval av verktygsradiekompenseringen får inte vara programmerade i samma block.

Framebyte

Alla anvisningar som bara hänför sig till baskoordinatsystemet är tillåtna (FRAME, verktygsradiekompensering). Ett framebyte vid G91 (kedjemått) behandlas dock inte speciellt – annorlunda än vid inaktiv transformation. Det inkrement som ska köras utvärderas i arbetsstyckskoordinatsystemet för den nya framen – oberoende av vilken frame som verkade i det föregående blocket.

Uteslutanden

Av transformationen berörda axlar kan inte användas:

- som Preset-axel (Iarm),
- för fixpunktuppsökningen (Iarm),
- för referenskörningen (Iarm).

7.12 Välja bort transformation (TRAFOOF)

Med den fördefinierade proceduren TRAFOOF kopplas alla aktiva transformationer och frames bort.

Märk

För bortvalet av transformationen gäller samma randvillkor (Sida 378) som för tillvalet.

Frames som behövs därefter måste kopplas aktiva genom förnyad programmering.

Syntax

```
...  
TRAFOOF
```

Betydelse

TRAFOOF:	Koppla från alla aktiva transformationer/frames
----------	---

Kinematiska följder

8.1 Radering av komponenter (DELOBJ)

Funktionen `DELOBJ()` "raderar" komponenter genom att återställa den tillordnade systemvariabeln till sitt defaultvärde:

- Element till kinematiska följder
- Skyddsområden, skyddsområdeselement och kollisionspar
- Transformationsdata

Syntax

```
[<RetVal>=] DELOBJ (<CompType> [ , , , <NoAlarm> ] )  
[<RetVal>=] DELOBJ (<CompType> , <Index1> [ , , <NoAlarm> ] )  
[<RetVal>=] DELOBJ (<CompType> [ , <Index1> ] [ , <Index2> ] [ , <NoAlarm> ] )
```

Betydelse

DELOBJ:	Radering av element till kinematiska följder, skyddsområden, skyddsområdeselement, kollisionsspar och transformationsdata	
<CompType>:	Typ av komponent som ska raderas	
	Datatyp:	STRING
	Värde:	"KIN_CHAIN_ELEM"
	Betydelse:	Systemvariabler för alla kinematiska element: \$NK_...
	Värde:	"KIN_CHAIN_SWITCH"
	Betydelse:	Systemvariabel \$NK_SWITCH[<i>]
	Värde:	"KIN_CHAIN_ALL"
	Betydelse:	Alla kinematiska element och brytare. Samma betydelse som det successiva anropet från DELOBJ med "KIN_CHAIN_ELEM" och "KIN_CHAIN_SWITCH"
	Värde:	"PROT_AREA"
	Betydelse:	Systemvariablerna till skyddsområdena:
	• \$NP_PROT_NAME • \$NP_CHAIN_NAME • \$NP_CHAIN_ELEM • \$NP_1ST_PROT	
	Värde:	"PROT_AREA_ELEM"
	Betydelse:	Systemvariabler för skyddsområdeselement till maskinskyddsområden och/eller automatiska verktygsskyddsområden:
	• \$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	
	Värde:	"PROT_AREA_COLL_PAIRS"
	Betydelse:	Systemvariabler för kollisionssparen:
	• \$NP_COLL_PAIR • \$NP_SAFETY_DIST	
	Värde:	"PROT_AREA_ALL"
	Betydelse:	Alla skyddsområden, skyddsområdeselement och kollisionsspar (systemvariabler \$NP_...) Samma betydelse som det successiva anropet från DELOBJ med "PROT_AREA", "PROT_AREA_ELEM" och "PROT_AREA_COLL_PAIRS"
	Värde:	"TRAFO_DATA"
	Betydelse:	Systemvariabler för alla transformationer \$NT_...

<Index1>:	Index för den första komponent som ska raderas (option)	
	Datotyp:	INT
	Defaultvärde:	-1
	Värdeområde:	$-1 \leq x \leq$ (maximalt antal projekterade komponenter -1)
	Värde	Betydelse
	0, 1, 2,	Index för komponenter som ska raderas
	-1	Alla komponenter av den angivna typen raderas. <Index2> utvärderas inte.
<Index2>:	Index för den sista komponenten som ska raderas (option) Är <Index2> inte programmerat raderas endast systemvariablerna för den i <Index1> refererade komponenten.	
	Datotyp:	INT
	Defaultvärde:	Endast systemvariablerna för den i <Index1> refererade komponenten raderas.
	Värdeområde:	<Index1> < x ≤ (max. antal projekterade komponenter -1)
<NoAlarm>:	Larmundertryckning (option)	
	Datotyp:	BOOL
	Defaultvärde:	FALSE
	Värde	Betydelse
	FALSE	I händelse av fel (<RetVal> < 0) stoppas programgenomarbetningen och ett larm indikeras .
<RetVal>:	TRUE	I händelse av fel stoppas programgenomarbetningen inte och det indikeras inget larm. Användningsfall: Användarspecifik reaktion motsvarande returvärde
	Funktionens returvärde	
	Datotyp:	INT
	Värdeområde:	0, -1, -2, ... -7
	Värde	Betydelse
	0	Inget fel uppträtt.
	-1	Anrop av funktion utan parameter. Åtminstone parametern <CompType> måste angivas.
	-2	<CompType> betecknar okänd komponent
	-3	<Index1> är mindre än -1
	-4	<Index1> är större än det projekterade antalet komponenter
	-5	<Index1> har vid radering av en komponent grupp ett värde skilt från -1
	-6	<Index2> är mindre än <Index1>
	-7	<Index2> är större än det projekterade antalet komponenter

8.2 Indexbestämning med namn (NAMETOINT)

I systemvariabelfälten av typ STRING finns användarspecifika namn införda. Med hjälp av beteckningen för systemvariabeln och namnet, fastställer funktionen `NAMETOINT()` det till namnet tillhörande indexvärdet, under vilket den är lagrad i systemvariabelfältet.

Syntax

```
<RetVal> = NAMETOINT(<SysVar>,<Name>[,<NoAlarm>])
```

Betydelse

NAMETOINT:	Fastställande av systemvariabelindex	
<SysVar>:	Namn på systemvariabelfältet av typ STRING	
	Datatyp:	STRING
	Värdeområde:	Namn på alla systemvariabelfält till NC av typ STRING
<Name>:	Teckenföljd resp. namn för vilka systemvariabelindex ska fastställas.	
	Datatyp:	STRING
<NoAlarm>:	Larmundertryckning (option)	
	Datatyp:	BOOL
	Defaultvärde:	FALSE
	Värde	Betydelse
	TRUE	I händelse av fel stoppas programgenomarbetningen inte och det indikeras inget larm. Användningsfall: Användarspecifik reaktion motsvarande returvärde
	FALSE	I händelse av fel (<RetVal> < 0) stoppas programgenomarbetningen och ett larm indikeras.
<RetVal>:	Systemvariabelindex eller felmeddelande	
	Datatyp:	INT
	Värdeområde:	$-1 \leq x \leq (\text{max. antal projekterade komponenter} - 1)$
	Värde	Betydelse
	≥ 0	Det sökta namnet hittades under angivet systemvariabelindex.
	-1	Det sökta namnet hittades inte resp. ett fel har uppträtt.

Exempel

Programkod	Kommentar
<pre>DEF INT INDEX \$NP_PROT_NAME[27] = "Abdeckung" ... INDEX = NAMETOINT("\$NP_PROT_NAME", "Abdeckung")</pre>	<pre>; INDEX == 27</pre>

Undvikande av kollision med kinematiska följder

Märk

Skyddsområden

De i de följande kapitlen nämnda skyddsområdena hänför sig till funktionen "Geometrisk maskinmodellering"

Litteratur:

Funktionshandbok Specialfunktioner, Kapitel "Geometrisk maskinmodellering"

9.1 Kontroll av kollisionsspar (COLLPAIR)

Funktionen `COLLPAIR(. . .)` fastställer om två skyddsområden bildar ett kollisionsspar.

Syntax

```
[<RetVal> =] COLLPAIR(<Name_1>,<Name_2>[,<NoAlarm>])
```

Betydelse

COLLPAIR:	Kontroll av tillhörighet till ett kollisionsspar		
<RetVal>:	Funktionens returvärde		
	Datotyp:	INT	
	Värde:	≥ 0	De båda skyddsområdena bildar ett kollisionsspar. Returvärde == kollisionssparindex m (se \$NP_COLL_PAIR)
		-1	Det anges antingen mindre än två strings eller minst en av de båda är noll-stringen.
		-2	Det i den första parametern angivna skyddsområdet hittades inte.
		-3	Det i den andra parametern angivna skyddsområdet hittades inte.
		-4	Inget av de båda angivna skyddsområdena hittades.
		-5	Båda angivna skyddsområdena hittades men inte gemensamt i ett kollisionsspar.
<Name_1>:	Namnet på det första skyddsområdet		
	Datotyp:	STRING	
	Värdeområde:	Parametrerat skyddsområdesnamn	
<Name_2>:	Namn för det andra skyddsområdet		
	Datotyp:	STRING	
	Värdeområde:	Parametrerat skyddsområdesnamn	
<NoAlarm>:	Larmundertryckning (option)		
	Datotyp:	BOOL	
	Värde:	FALSE (default)	I händelse av fel (<RetVal> < 0) stoppas programgenomarbetningen och ett larm indikeras.
		TRUE	I händelse av fel stoppas programgenomarbetningen inte och det indikeras inget larm. Användningsfall: Användarspecifik reaktion motsvarande returvärde

9.2 Begära nyberäkning av maskinmodellen för undvikande av kollision (PROTA)

Skrivs systemvariabler från den kinematiska kedjan \$NK_..., den geometriska maskinmodelleringen eller undvikandet av kollision \$NP_... i detaljprogrammet, måste sedan proceduren `PROTA` anropas, så att ändringen i den NC-interna maskinmodellen för undvikande av kollision blir verksam.

Syntax

`PROTA[ (<Par> ) ]`

Betydelse

PROTA:	Begära nyberäkning av maskinmodellen för undvikande av kollision		
	• Utlöser fördekoderingsstopp. • Måste stå ensam i blocket.		
<Par>:	Parameter (option)		
	Datotyp:	STRING	
	Värde:	---	Utan parameter. En nyberäkning av maskinmodellen genomförs. Stati för skyddsområdena bibehålls.
		"R"	En nyberäkning av maskinmodellen genomförs. Skyddsområdena flyttas till sin initialiseringsstatus motsvarande \$NP_INIT_STAT.

Randvillkor

Simulering

Proceduren `PROTA` får i detaljprogrammen inte användas i samband med simuleringen (simNC).

Exempel: Undvikande av anrop av `PROTA`, under det simuleringen är aktiv.

Programkod	Kommentar
...	
IF \$P_SIM == FALSE	; IF Simulering inte aktiv
PROTA	; THEN Nyberäkna kollisionmodell
ENDIF	; ENDIF
...	

Se även

Sätta skyddsområdestillstånd (PROTS) (Sida 388)

9.3 Sätta skyddsområdestillstånd (PROTS)

Proceduren `PROTS ( . . . )` sätter status för skyddsområden på den angivna värdet.

Syntax

```
PROTS (<State>[, <Name_1>, ..., <Name_n>])
```

Betydelse

PROTS:	Sätta status för skyddsområden		
	• Måste stå ensam i blocket.		
<State>:	Status på vilken de angivna skyddsområdena ska sättas		
	Datatyp:	CHAR	
	Värde:	"A"eller "a"	Status: aktiv
		"I"eller "i"	Status: inaktiv
		"P"eller "p"	Status: Föraktiverad resp. PLC-styrd ¹⁾
"R"eller "r"		Status: NC-internt värde för initialiseringsstatus ²⁾	
<Name_1> ... <Name_n>:	Namn för ett eller flera skyddsområden som ska sättas på angiven status (option)		
	Är inget namn angivet, sätts den angivna statusen för alla definierade skyddsområden.		
	Datatyp:	STRING	
	Värdeområde:	Parametrerat skyddsområdesnamn	
Observera: Det maximala antalet skyddsområden som kan angivas som parametrar, är endast beroende av det maximalt möjliga antalet tecken per programrad.			

¹⁾ Aktiveringen / Inaktiveringen sker via: DB10.DBX234.0 - DBX241.7

²⁾ Status sätts på det NC-interna värdet för initialiseringsstatus, dvs. på det värde, som systemvariabeln \$NP_INIT_STAT hade vid tidpunkten för det sista anropet av PROTA(...) (Sida 387).

9.4 Avståndsbestämning för två skyddsområden (PROTD)

Funktionen `PROTD ( . . . )` beräknar avståndet för två skyddsområden.

Funktionsegenskaper:

- Avståndsberäkningen sker oberoende av status för skyddsområdena (aktiverade, inaktiverade, föraktiverade).
- Till avståndsberäkning för två skyddsområden används endast sådana skyddsområdeselement, som är markerade med `$NP_USAGE = "C"` eller `"A"`. Skyddsområdeselement till skyddsområdet, som är markerade med `$NP_USAGE = "V"` kommer inte i fråga.
- Skyddsområden, hos vilka alla skyddsområdeselement är markerade med `$NP_USAGE = "V"` kan inte användas till avståndsberäkning.
- Avståndsberäkningen sker med de i slutet av det föregående blocket giltiga positionerna.
- Överlagringar som räknas in i huvudkörningen, t. ex. DRF-förflyttning eller externa nollpunktsförflyttningar, ingår med de vid **interpretationstidpunkten** för funktionen giltiga värdena i avståndsberäkningen.

Märk

Synkronisering

Vid användning av funktionen `PROTD ( . . . )` är det uteslutande användarens ansvar att synkronisera huvud- och förkörning eventuellt med fördekoderingsstopp `STOPRE`.

Kollision

Föreligger mellan de angivna skyddsområdena en kollision levererar funktionen ett avstånd på 0,0. En kollision föreligger, när de båda skyddsområdena berör eller tränger in i varandra.

Det ska inte tas hänsyn till säkerhetsavståndet för kollisionskontrollen (MD10622 `$MN_COLLISION_SAFETY_DIST`) vid avståndsberäkningen.

Syntax

```
[<RetVal> =] PROTD ([<Name_1>], [<Name_2>], VAR <Vector>[, <System>])
```

Betydelse

PROTD:	Beräknar avståndet för de båda angivna skyddsområdena	
	• Måste stå ensam i blocket.	
<RetVal>:	Funktionens returvärde: Absolutvärde för avståndet för de båda skyddsområdena eller 0,0 vid kollision (se ovan: avsnitt Kollision)	
	Datotyp:	REAL
	Värdeområde:	$0,0 \leq x \leq +\text{max. REAL-värde}$

9.4 Avståndsbestämning för två skyddsområden (PROTD)

<Name_1>, <Name_2>:	Namnen för de båda skyddsområdena vars avstånd till varandra ska beräknas (option)		
	Datatype:	STRING	
	Värdeområde:	Parametrerat skyddsområdesnamn	
	Defaultvärde:	"" (tom string) Är inga skyddsområden angivna beräknar funktionen det aktuellt minsta avståndet ur alla i kollisionsmodellen ingående aktiverade och föraktiverad skyddsområden.	
<Vector>:	Returvärde: 3-dimensionell avståndsvektor från skyddsområde <Name_2> till skyddsområde <Name_1> med: • <Vector>[0]: X-koordinat i världskoordinatsystemet • <Vector>[1]: Y-koordinat i världskoordinatsystemet • <Vector>[2]: Z-koordinat i världskoordinatsystemet Vid kollision: <Vector> == Nollvektor		
	Datatype:	VAR REAL [3]	
	Värdeområde:	<Vector> [n]: $0,0 \leq x \leq \pm\text{max}$. REAL-värde	
<System>:	Måttsystem (inch / metriskt) för avstånd och avståndsvektor (option)		
	Datatype:	BOOL	
	Värde:	FALSE (default)	Måttsystem motsvarande det aktuellt aktiva G-kommandot ur G-gruppen 13 (G70, G71, G700, G710).
		TRUE	Måttsystem motsvarande det inställda grundsystemet: MD52806 \$MN_ISO_SCALING_SYSTEM

Transformationer med kinematiska följder

10.1 Koppla till transformation (TRAFOON)

En med kinematiska följder definierad transformation aktiveras med den fördefinierade proceduren TRAFOON. Anropet måste stå ensamt i ett block.

Märk

Alternativt kan en med kinematiska följder definierad transformation också aktiveras via konventionella språkkommandon som t. ex. TRAORI eller TRANSMIT. För detta måste ett motsvarande värde skilt från noll vara infört i systemvariabeln \$NT_TRAFO_INDEX.

Ytterligare informationer till \$NT_TRAFO_INDEX se "Handbok med listor Systemvariabler".

Syntax

TRAFOON(<Trafoname>,<Diameter>,<k>)

Betydelse

TRAFOON:	Procedur för aktivering av en med kinematiska följder definierad transformation	
<Trafoname>:	Namn på transformationsdatablocket	
	Datatyp:	STRING
	Värdeområde:	Alle via \$NT_NAME definierade namn för transformationsdatablock
	Observera: Namnet för transformationsdatablocket måste vara entydigt. Det får förekomma endast en gång i \$NT_NAME.	
<Diameter>:	Referens- resp. arbetsdiameter(endast TRACYL)	
	Datatyp:	REAL
	Värdet måste vara > 1.	
<k>:	Definierar användningen av spårväggskompenseringen (endast TRACYL).	
	Datatyp:	BOOL
	Värde:	FALSE utan spårväggskompensering
		TRUE med spårväggskompensering
	Motsvarar TRACYL-transformationstypen 514 (spårväggskompensering programmerbar). Är <k> inte angiven, verkar den parameterade inställningen av bit 10 i \$NT_CNTRL[<n>].	

Exempel

Programkod	Kommentar
TRAFOON["Trans_1"]	Aktiverar transformationen med namnet Trans_1.

10.2 Modifiera orienteringstransformation enligt uppmätning av maskinen (CORRTRAFO)

Vid maskiner med orienteringstransformationer, som definierades med hjälp av kinematiska följder, kan användaren nyttja den fördefinierade funktionen CORRTRAFO, för att efter en uppmätning av maskinen modifiera offsetvektorerna eller riktningsvektorerna i den kinematiska modellen för maskinen.

Syntax

```
<Corr_Status> = CORRTRAFO(<Corr_Vect>, <Corr_Index>, <Corr_Mode> [,  
<No_Alarm>])
```

Betydelse

CORRTRAFO:	Funktionsanrop
------------	----------------

10.2 Modifiera orienteringstransformation enligt uppmätning av maskinen (CORRTRAFO)

<Corr_Status>:	Funktionens returvärde	
	Datotyp:	INT
	Värden:	0 Funktionen utfördes utan fel.
		1 Ingen transformation är aktiv.
		2 Den aktuellt aktiva transformationen är ingen orienterings-transformation.
		3 Den aktiva orienteringstransformationen definierades inte med kinematiska följder.
		10 Anropsparametern <Corr_Index> är negativ.
		11 Anropsparametern <Corr_Mode> är negativ.
		12 Ogiltig hänvisning till ett avsnitt i en delföljd (1:a ställe i <Corr_Index>). Värdet får inte vara större än antalet orienteringsaxlar i delföljden.
		13 Ogiltig hänvisning till orienteringsaxlarna i en delföljd (1:a ställe i <Corr_Index>). Värdet måste vara mindre än antalet orienteringsaxlar i delföljden.
		14 Ogiltig hänvisning till en delföljd (10:e ställe i <Corr_Index>). Endast värdena 0 och 1 är tillåtna (hänvisning till Part- resp. Tool-följd). Detta felnummer uppträder också när delföljden, hänvisad till <Corr_Index>, inte existerar.
		15 I det avsnitt, till vilket hänvisas med parametern <Corr_Index>, är inget kompenseringselement definierat (\$NT_CORR_ELEM_P resp. \$NT_CORR_ELEM_T).
		20 Ogiltig kompenseringsmode (1:a ställe i <Corr_Mode>). Endast värdena 0 och 1 är tillåtna.
		21 Ogiltig kompenseringsmode (10:e och/eller 100:e ställe i <Corr_Mode>). Vid skrivande av en axelriktning får endast det 1:a stället vara skilt från noll.
		30 Det 100:e stället i <Corr_Mode> är ogiltigt. Endast värdena 0 och 1 är tillåtna.
		31 Det 1000:e stället i <Corr_Mode> är ogiltigt. Endast värdena 0 och 1 är tillåtna.
		40 Den rikttningsvektor, som ska övertas som axelriktning, är nollvektorn. Detta fel kan endast uppträda, när det 1000:e stället i <Corr_Mode> är lika med 0. Är det 1000:e stället i denna parameter lika med 1 (övervakning av den maximala kompenseringen inaktiverad), kan också nollvektorn skrivas.
		41 Vid kompenseringen av en offsetvektor är avvikelser från det aktuella värdet i minst en koordinat större än det genom settingdatum SD41610 \$SN_CORR_TRAFO_LIN_MAX föreskrivna maximalvärdet. Parametern <Corr_Vect> skrivs över av en felvektor. Det gäller också när bearbetningen avbryts med larm (se parameter <No_Alarm>). Felvektorn innehåller i de komponenter, vars kompenseringssvärde har överskridit den tillåtna gränsen (Limit), den förteckensriktiga differensen mellan det fastställda kompenseringssvärdet och Limit. Innehållet i de komponenter, som inte har överskridit sin Limit, är noll.
		42 Vid kompenseringen av en rikttningsvektor är vinkelavvikelsen från den aktuella riktningen större än det genom settingdatum

10.2 Modifiera orienteringstransformation enligt uppmätning av maskinen (CORRTRAFO)

			SD41611 \$SN_CORR_TRAFO_DIR_MAX föreskrivna maximalvärdet.
		43	Försöket, att beskriva en systemvariabel, avvisades på grund av att skrivrätter saknas.
<Corr_Vect>:	Kompenseringsvektor Innehållet i kompenseringsvektorn definieras av de följande parametrarna <Corr_Index> och <Corr_Mode>. Är <Corr_Status> = 41, skrivs innehållet i vektorn över (s. o.).		
	Datatyp:	REAL	
<Corr_Index>:	Avsnitt, vars kompenseringsselement ska modifieras / index för den orienteringsaxel, vars riktningsvektor ska modifieras		
	Datatyp:	INT	
	Parametern <Corr_Index> är decimalkodad (1:a till 10:e ställe):		
	1:a ställe:	Innehåller index för avsnittet resp. orienteringsaxeln i delföljden.	
	10:e ställe:	Hänvisar till delföljden.	
0x		Arbetsstycks-följd	
1x		Verktogs-följd	

10.2 Modifiera orienteringstransformation enligt uppmätning av maskinen (CORRTRAFO)

<Corr_Mode>:	Kompenseringsmode	
	Datotyp:	INT
	Parametern <Corr_Mode> är decimalkodad (1:a till 1000:e ställe):	
	1:a ställe:	Bestämmer, vilket element som ska kompenseras.
		xxx0 Kompensering av en linjär offsetvektor
		xxx1 Kompensering av riktningsvektorn för en orienteringsaxel
	10:e ställe:	Bestämmer, hur det kompenseringsselement, till vilket innehållet i <Corr_Index> hänvisar, ska modifieras.
		xx0x Kompenseringsvektorn skrivs omedelbart i kompenseringsselementet. Denna variant kan användas till att omedelbart beskriva kompenseringsselementet, utan att index <n> för det ifrågasvarande systemdatumet (\$NK_OFF_DIR[<n>, ...]) måste vara bekant.
		xx1x Som 0, dock med den skillnaden, att det överlämnade kompenseringsvärdet interpreteras i världskoordinater. En skillnad mellan varianterna 0 och 1 kan alltid uppstå, när den kinematiska följden innehåller ytterligare rotationer i grundläge (positioner för alla orienteringsaxlar lika med 0).
		xx2x Som 1, dock med den skillnaden att kompenseringsvärdet hänförs till hela avsnittet, dvs. i kompenseringsselementet blir ett sådant värde infört, att hela avsnittet uppnår den genom kompenseringsvärdet definierade längden.
		Observera: Värden 1 och 2 är inte tillåtna vid beskrivningen av riktningsvektorn för en orienteringsaxel.
	100:e ställe:	Bestämmer hur innehållet i parametern <Corr_Vect> ska interpreteras.
		x0xx Den överlämnade kompenseringsvektorn <Corr_Vect> innehåller den totala nya längden för kompenseringsselementet resp. för avsnittet, till vilken <Corr_Index> i förbindelse med 10:e stället i <Corr_Mode> hänvisar (absolut kompensering).
		x1xx Den överlämnade kompenseringsvektorn <Corr_Vect> innehåller endast differensen till den aktuella längden för kompenseringsselementet resp. för avsnittet, till vilken <Corr_Index> i förbindelse med 10:e stället i <Corr_Mode> hänvisar (inkrementell kompensering).
		Observera: Vid kompenseringen av riktningsvektorn för en orienteringsaxel måste innehållet i det 100:e stället vara 0.
	1000:e ställe:	Bestämmer om kompenseringen ska begränsas av följande maximalvärde:
		- SD41610 \$SN_CORR_TRAFO_LIN_MAX resp. - SD41611 \$SN_CORR_TRAFO_DIR_MAX
		0xxx Övervakning av den maximala kompenseringen är aktiv.
		1xxx Övervakning av den maximala kompenseringen är inte aktiv.

10.2 Modifiera orienteringstransformation enligt uppmätning av maskinen (CORRTRAFO)

<No_Alarm>:	Beteende i händelse av fel (returvärde > 0) (option)		
	Datatyp:	BOOL	
	Värde:	FALSE (default)	I händelse av fel stoppas programgenomarbetningen och larm 14103 indikeras.
		TRUE	I händelse av fel stoppas programgenomarbetningen inte och det indikeras inget larm. Användningsfall: Användarspecifik reaktion motsvarande returvärde

Märk

Om ett fel uppträder vid anrop av funktionen, matas antingen ett larm ut eller ett felnummer skickas tillbaka (se parameter <No_Alarm>), så att användaren själv kan reagera lämpligt på feltillståndet. Felorsaken betecknas närmare genom en larmparameter. Ett i stället för ett larm tillbaka skickat felnummer är identiskt med larmparametern.

Ytterligare informationer till CORRTRAFO

Den kinematiska strukturen för en maskin med orienteringstransformation beskrivs av en eller två kinematiska följder (delföljder), som utgår från nollpunkten till världskoordinatsystemet. En av de båda följderna, **verktygs-följden**, slutar i referenspunkten för verktyget, den andra, **arbetsstycks-följden** i nollpunkten till baskoordinatsystemet.

Funktionen CORRTRAFO mäter hävarmslängden och axelriktningen vid maskiner med orienteringstransformation och skriver dessa värden i speciella kompenseringselement. En kinematisk följd beskrivs bland annat med element av typ OFFSET, som definieras via \$NK_TYPE.

CORRTRAFO arbetar med avsnitt

De båda delföljderna kan dela upp sig var och en i maximalt fyra avsnitt:

- Avsnitt 1 börjar i startpunkt för följden och slutar i den första orienteringsaxeln.
- Avsnitt 2 är avsnittet mellan orienteringsaxeln 1 och orienteringsaxeln 2.
- Avsnitt 3 är avsnittet mellan orienteringsaxeln 2 och orienteringsaxeln 3.
- Avsnitt 4 är avsnittet mellan orienteringsaxeln 3 och slutet på verktygs- resp. arbetsstycks-följden.

Avsnitten kan vart och ett innehålla konstanta följelement av typen OFFSET eller ROT_CONST.

Den följande grafiken visar en orienteringstransformation med 2 orienteringsaxlar.

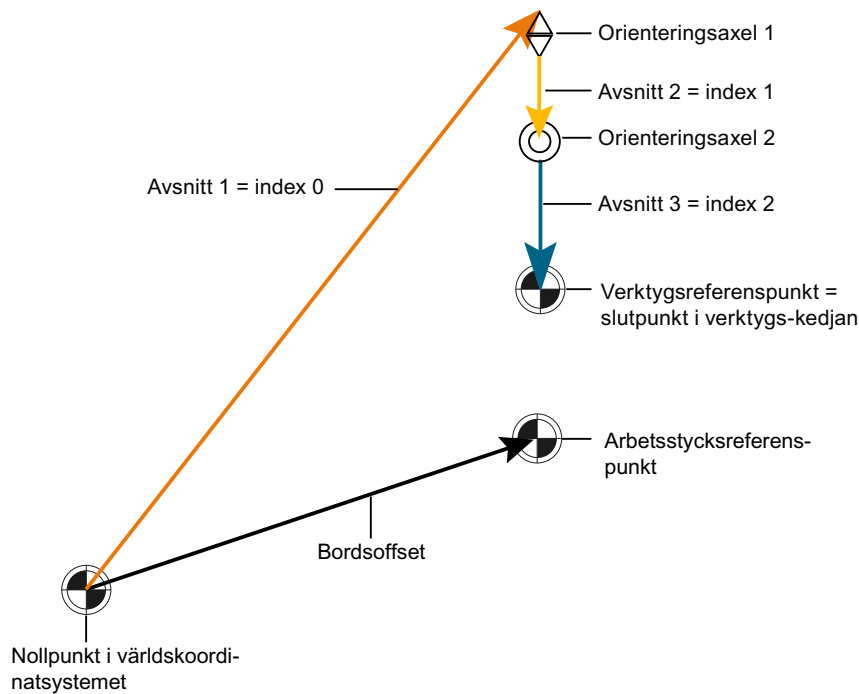


Bild 10-1 CORRTRAFO exempel

Avsnitten är entydigt definierade: Genomlöper man den kinematiska delföljden från dess startpunkt ända till slutpunkten, så har det första avsnittet index 0, det nästa index 1 osv. Index för det sista avsnittet är då alltid lika med antalet orienteringsaxlar.

Kompenseringsselement

Med systemvariabeln \$NT_CORR_ELEM_T[<n>, 0 ... 3] resp. \$NT_CORR_ELEM_P[<n>, 0 ... 3] kan i alla dessa avsnitt hänvisas till ett konstant kinematiskt följelement (följelement av typ \$NK_TYPE[<n>] = "OFFSET"). I dessa så betecknade elementen skrivs med hjälp av funktionen CORRTRAFO de kompenseringsvärden, som fastställdes vid uppmätningen av maskinen.

Exempel med transformations-Index = 1:

- \$NT_CORR_ELEM_T[1,0] = "C_AXIS_OFFSET"; offset för C-axeln (orienteringsaxel 1) i avsnitt 1 är definierad som kompenseringsselement.
- \$NT_CORR_ELEM_T[1,1] = "B_AXIS_OFFSET"; offset för B-axeln (orienteringsaxel 2) i avsnitt 2 är definierad som kompenseringsselement.
- \$NT_CORR_ELEM_T[1,2] = "BASE_TOOL_OFFSET"; offset från B-axeln till verktygsreferenspunkten i avsnitt 3 är definierad som kompenseringsselement.

Ordningsföljden för hänvisningarna i \$NT_CORR_ELEM_T/P[<n>, 0 ... 3] måste stämma överens med de ovan beskrivna avsnitten, dvs. i \$NT_CORR_ELEM_T/P[<n>, 0] kan endast ett följelement stå, som ligger före den första orienteringsaxeln osv..

Funktionen CORRTRAFO skriver i de så definierade kompenseringsselementen de värden, som bestämdes vid uppmätningen av maskinen. Modifieringen av kompenseringsvärdena definieras i CORRTRAFO via parametern <Corr_Mode>.

10.2 Modifiera orienteringstransformation enligt uppmätning av maskinen (CORRTRAFO)**Stänga följd**

Har bit 7 eller bit 8 ställts in i systemvariabeln \$NT_CNTRL[<n>], infogas i slutet av arbetsstycks-följden (bit 7) resp. före startpunkten för verktygs-följden (bit 8) internt automatiskt ytterligare konstanta följelement, som upprättar en förbindelse från slutpunkten på följden till maskinnollpunkten ("Stänga följd").

Dessa automatiskt infogade element kan inte beskrivas utifrån, utan endast läsas (se för detta systemvariabeln \$AC_TRAFO_CORR_ELEM_P/T).

Punkt för att stänga verktygs-följden

Är systemvariabeln \$NT_CLOSE_CHAIN_T inte tom, stängs verktygs-följden inte i slutpunkten på följden, utan i slutpunkten för det betecknade följelementet. Ytterligare följelement, som ligger bakom denna punkt, leder då vid aktiveringen av transformationen till en motsvarande nollpunktsförflyttning.

Index för en orienteringsaxel

Förutom den konstanta offseten mellan orienteringsaxlarna kan också riktningsvektorerna till orienteringsaxlarna beskrivas med funktionen CORRTRAFO. Med indexet till en orienteringsaxel menas här det index, som resulterar när man genomlöper den kinematiska delföljden från början till slut, varvid räkningen börjar vid noll. Indexet för en orienteringsaxel är därför alltid lika med indexet för det avsnitt som ligger framför.

Indexet för en orienteringsaxel kan också fastställas med systemvariabeln \$AC_TRAFO_ORIAX_LOC.

Maximalt tillåten förändring av ett följelement

Den maximalt tillåtna förändringen för ett följelement kan begränsas genom de båda settingdatum SD41610 \$SN_CORR_TRAFO_LIN_MAX för offsetvektorerna och SD41611 \$SN_CORR_TRAFO_DIR_MAX för riktningsvektorerna till orienteringsaxlarna. SD41610 \$SN_CORR_TRAFO_LIN_MAX anger, med vilket belopp varje enskild vektorkomponent får förändras från sitt referensvärde. SD41611 \$SN_CORR_TRAFO_DIR_MAX anger, med vilken vinkel riktningen för axelvektorn maximalt får förändra sig från sitt referensvärde. Referensvärdet är därvid alltid det motsvarande värde, som är verksamt i den aktiva transformationen vid anrop av CORRTRAFO. Det vill säga att eventuellt förändrat innehåll i kinematikdata efter aktiveringen av transformationen inte har något inflytande på verkningssättet för funktionen CORRTRAFO i datahanteringen.

Verktygskompenseringar

11.1 Kompenseringsminne

Uppbyggnad av kompenseringssminnet

Varje datafält kan anropas med ett T- och D-nummer (utom "Grund D-nr") och innehåller förutom de geometriska uppgifterna för verktyget också ytterligare poster t.ex. verktygstypen.

Grund D-nummer-struktur

Den "grunda D-nummer-strukturen" används, när verktygsförvaltningen äger rum utanför NC. I detta fall anläggs D-numren med de tillhörande verktygskompenseringsblocken utan tillordning till verktygen.

I detaljprogrammet kan fortfarande T programmeras. Detta T har dock ingen relation till det programmerade D-numret.

Användar-skärdata

Via maskindatum kan användar-skärdata konfigureras. Följ anvisningarna från maskintillverkaren.

Verktygsparameter

Märk

Enskilda värden i kompenseringssminnet

De enskilda värdena i kompenseringssminnet P1 till P25 kan läsas och skrivas av programmet via systemvariabel. Alla övriga parametrar är reserverade.

Verktygsparametrarna \$TC_DP6 till \$TC_DP8, \$TC_DP10 och \$TC_DP11 samt \$TC_DP15 till \$TC_DP17, \$TC_DP19 och \$TC_DP20 har en annan betydelse beroende på verktygstypen.

Verktygsparameter Nummer (DP)	Betydelse för systemvariablerna	Anmärkning
\$TC_DP1	Verktygstyp	Översikt se lista
\$TC_DP2	Skärläge	endast för svarvverktyg
Geometri	Längdkompensering	
\$TC_DP3	Längd 1	Beräkning enligt
\$TC_DP4	Längd 2	Typ och plan
\$TC_DP5	Längd 3	
Geometri	Radie	
\$TC_DP6 ¹⁾	Radie 1 / längd 1	Fräs-/svarv-/slipverkt.
\$TC_DP6 ²⁾	Diameter d	Spårsåg

Verktysparameter Nummer (DP)	Betydelse för systemvariablerna	Anmärkning
\$TC_DP7 ¹⁾	Längd 2 / hörnradie konisk fräs	Fräsverktyg
\$TC_DP7 ²⁾	Spårbredd b hörnradie	Spårsåg
\$TC_DP8 ¹⁾	Rundningsradie 1 för fräsverktyg	Fräsverktyg
\$TC_DP8 ²⁾	Överhäng k	Spårsåg
\$TC_DP9 ^{1) 3)}	Rundningsradie 2	reserverat
\$TC_DP10 ¹⁾	Vinkel 1 frontsidan på verktyget	koniska fräsverktyg
\$TC_DP11 ¹⁾	Vinkel 2 verktygs-längdaxel	koniska fräsverktyg
Förslitning	Längd- och radiekompensering	
\$TC_DP12	Längd 1	
\$TC_DP13	Längd 2	
\$TC_DP14	Längd 3	
\$TC_DP15 ¹⁾	Radie 1 / längd 1	Fräs-/svarv-/slipverkt.
\$TC_DP15 ²⁾	Diameter d	Spårsåg
\$TC_DP16 ¹⁾	Längd 2 / hörnradie konisk fräs	Fräsverktyg
\$TC_DP16 ³⁾	Spårbredd b hörnradie	Spårsåg
\$TC_DP17 ¹⁾	Rundningsradie 1 för fräsverktyg	Fräsa / 3D ändplansfräsa
\$TC_DP17 ²⁾	Överhäng k	Spårsåg
\$TC_DP18 ^{1) 3)}	Rundningsradie 2	reserverat
\$TC_DP19 ¹⁾	Vinkel 1 frontsidan på verktyget	koniska fräsverktyg
\$TC_DP20 ¹⁾	Vinkel 2 verktygs-längdaxel	koniska fräsverktyg
Basmått/adapter	Längdkompenseringar	
\$TC_DP21	Längd 1	
\$TC_DP22	Längd 2	
\$TC_DP23	Längd 3	
Teknologi		
\$TC_DP24	Frivinkel	endast för svarvverktyg
\$TC_DP25		reserverat

¹⁾ Gäller även vid fräsverktyg för 3D-ändplansfräsning

²⁾ Vid spårsåg verktygstyp

³⁾ reserverat (används inte av SINUMERIK 840D sl)

Anmärkningar

För de geometriska storlekarna (t.ex. längd 1 eller radie) finns flera inmatningskomponenter. Dessa räknas samman additivt till en resulterande storlek (t.ex. totallängd 1, totalradie) som sedan kommer till verkan.

Ej nödvändiga kompenseringar ska beläggas med värdet noll.

Verktygsparmetrar \$TC-DP1 till \$TC-DP23 med konturverktyg

Märk

Verktygsparmetrar som inte är uppförda i tabellen som t. ex. \$TC_DP7 utvärderas inte, dvs. deras innehåll är utan betydelse.

Verktygsparmeter Nummer (DP)	Betydelse	Skär Dn		Anmärkning
\$TC_DP1	Verktygstyp			400 till 599
\$TC_DP2	Skärläge			
Geometri	Längdkompensering			
\$TC_DP3	Längd 1			
\$TC_DP4	Längd 2			
\$TC_DP5	Längd 3			
Geometri	Radie			
\$TC_DP6	Radie			
Geometri	Gränsvinkel			
\$TC_DP10	minimal gränsvinkel			
\$TC_DP11	maximal gränsvinkel			
Förslitning	Längd- och radiekompensering			
\$TC_DP12	Slitage längd 1			
\$TC_DP13	Slitage längd 2			
\$TC_DP14	Slitage längd 3			
\$TC_DP15	Slitage radie			
Förslitning	Gränsvinkel			
\$TC_DP19	Slitage min. gränsvinkel			
\$TC_DP20	Slitage max. gränsvinkel			
Basmått/adapter	Längdkompenseringar			
\$TC_DP21	Längd 1			
\$TC_DP22	Längd 2			
\$TC_DP23	Längd 3			

Grundvärde och förslitningsvärde

De resulterande storlekarna bildas alltid ur summan av grundvärdet och förslitningsvärdet (t.ex. \$TC_DP6 + \$TC_DP15 för radien). Till verktygslängden för det första skäret adderas dessutom också basmåttet (\$TC_DP21 – \$TC_DP23). Dessutom verkar på denna verktygslängd alla andra storlekar som också kan inverka på ett vanligt verktygs effektiva verktygslängd (adapter, orienterbar verktygsbärare, settingdata).

Gränsvinkel 1 och 2

Gränsvinklarna 1 resp. 2 hänför sig alltid till vektorn från skärmittpunkten till skärreferenspunkten och räknas medurs.

11.2 Additiva kompenseringar

11.2.1 Välja additiva kompenseringar (DL)

Additiva kompenseringar kan betraktas som i bearbetningen programmerbara processkompenseringar. De hänför sig till geometriska data för ett skär och är därmed beståndsdel i skärdata för verktyget.

Data för en additiv kompensering anropas via ett DL-nummer (DL: Location dependent; kompenseringar gällande respektive användningsplats) och matas in via användargränssnittet.

Användning

Genom additiva kompenseringar kan måttfel som beror på användningsplatsen utjämnas.

Syntax

DL=<Nummer>

Betydelse

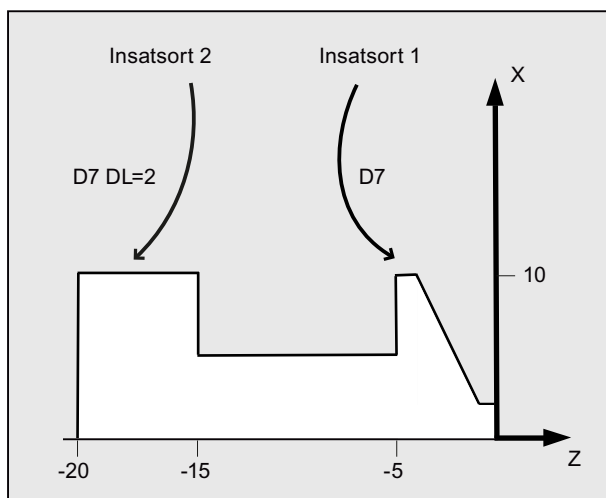
DL:	Kommando för aktivering av en additiv kompensering
<Nummer>:	Via parametern <Nummer> anges det additiva datablocket för verktygskompenseringen som ska aktiveras

Märk

Fastläggandet av antalet och aktiveringen av de additiva kompenseringarna görs via maskindata (→ Följ maskintillverkarens anvisningar!).

Exempel

Samma skär används för 2 lagersäten:



Programkod	Kommentar
N110 T7 D7	; Revolvern positioneras på plats 7. D7 och DL=1 aktiveras och körs ut i nästa block.
N120 G0 X10 Z1	
N130 G1 Z-6	
N140 G0 DL=2 Z-14	; Additivt till D7 aktiveras DL=2 och körs ut i nästa block.
N150 G1 Z-21	
N160 G0 X200 Z200	; Uppsöka verktygsväxlingspunkten.
...	

11.2.2 Fastlägga förslitnings- och inställningsvärden (\$TC_SCPxy[t,d], \$TC_ECPxy[t,d])

Förslitnings- och inställningsvärden kan läsas och skrivas via systemvariabler. Därvid orienterar sig logiken till logiken för de motsvarande systemvariablerna för verktyg och skär.

Systemvariabler

\$TC_SCPxy[<t>,<d>]:	Förslitningsvärden som via xy är tillordnade till respektive geometriparameter, varvid x motsvarar numret på förslitningsvärdet och y upprättar relationen till geometriparametern.
\$TC_ECPxy[<t>,<d>]:	Inställningsvärden som via xy är tillordnade till respektive geometriparameter, varvid x motsvarar numret på inställningsvärdet och y upprättar relationen till geometriparametern.
<t>: T-nummer på verktyget	
<d>: D-nummer på skäret till verktyget	

Märk

De fastlagda förslitnings- och inställningsvärdena adderas till geometriparametrarna och de övriga kompenseringsparametrarna (D-nummer).

Exempel

Förslitningsvärdet för längd 1 fastläggs för skäret <d> till verktyget <t> till värdet 1.0.

Parameter: \$TC_DP3 (längd 1, för svarvverktyg)

Förslitningsvärden: \$TC_SCP13 till \$TC_SCP63

Inställningsvärden: \$TC_ECP13 till \$TC_ECP63

\$TC_SCP43 [<t>,<d>] = 1.0

11.2.3 Radera additiva kompenseringar (DELDL)

Med kommandot **DELDL** raderas additiva kompenseringar för skäret till ett verktyg (frigivande av minnet). Därvid raderas både de fastlagda förslitningsvärdena och även inställningsvärdena.

Syntax

```
DELDL [<t>,<d>]
DELDL [<t>]
DELDL
<Status>=DELDL [<t>,<d>]
```

Betydelse

DELDL:	Kommando för att radera additiva kompenseringar	
<t>:	T-nummer på verktyget	
<d>:	D-nummer på skäret till verktyget	
DELDL [<t>,<d>]:	Alla additiva kompenseringar för skäret <d> till verktyget <t> raderas.	
DELDL [<t>]:	Alla additiva kompenseringar för alla skär till verktyget <t> raderas.	
DELDL:	Alla additiva kompenseringar för alla skär till alla verktyg i TO-enheten raderas (för den kanal i vilken kommandot programmeras).	
<Status>:	Rader-status	
	Värde:	Betydelse:
	0	Raderingen genomfördes med framgång.
	-	Raderingen genomfördes inte (när parametreringen betecknar exakt ett skär) eller raderingen gjordes inte fullständigt (när parametreringen betecknar flera skär).

Märk

Förslitnings- och inställningsvärden för aktiva verktyg kan inte raderas (förhåller sig analogt till raderbeteendet för $\mathbb{D}$ resp. verktygsdata).

11.3 Verkygskompensering-specialbehandling

Med settingdata SD42900 till SD42960 låter sig värderingen av förtecknen till verktygslängder och förslitning styras.

Detta gäller likaså för beteendet för förslitningskomponenterna vid spegling av geometriaxlar eller vid byte av bearbetningsplan och även för temperaturkompensering i verktygsriktningen.

Förslitningsvärden

När det i det följande refereras till förslitningsvärden menas därmed alltid summan av de egentliga förslitningsvärdena (\$TC_DP12 till \$TC_DP20) och summakompenseringarna med förslitningsvärdena (\$SCPX3 bis \$SCPX11) och inställningsvärdena (\$ECPX3 till \$ECPX11).

Närmare om summakompenseringarna se:

Litteratur:

Funktionshandbok Verkygsförvaltning

Settingdata

SD42900 \$SC_MIRROR_TOOL_LENGTH	Spegling av verktygslängdkomponenter och komponenter i basmåttet.
SD42910 \$SC_MIRROR_TOOL_WEAR	Spegling av förslitningsvärden för verktygslängdkomponenterna.
SD42920 \$SC_WEAR_SIGN_CUTPOS	Förteckenvärdering av förslitningskomponenterna beroende av skärläget.
SD42930 \$SC_WEAR_SIGN	Inverterar förtecknet för förslitningsmått.
SD42935 \$SC_WEAR_TRANSFORM	Transformation av förslitningsvärdena.
SD42940 \$SC_TOOL_LENGTH_CONST	Tillordning av verktygslängdkomponenterna till geometriaxlarna.
SD42950 \$SC_TOOL_LENGTH_TYPE	Tillordning av verktygslängdkomponenter oberoende av verktygstyp.
SD42960 \$SC_TOOL_TEMP_COMP	Temperaturkompenseringsvärde i verktygsriktningen. Är verksam även vid förefintlig verktygsorientering.

Litteratur

Funktionshandbok Grundfunktioner; Verkygskompensering (W1)

Ytterligare informationer

Hur förändrade settingdata blir verksamma

Nyvärderingen av verktygskomponenterna vid en ändring av de beskrivna settingdata blir verksam först när ett verktygsskär väljs nästa gång. Är ett verktyg redan aktivt och värderingen av data för detta verktyg ska förändras verksamt måste detta verktyg väjas på nytt.

Motsvarande gäller för det fall att den resulterande verktygslängden ändrar sig eftersom speglingstillståndet för en axel ändrades. Verktøget måste väljas på nytt efter spegelkommandot för att de ändrade verktygslängdkomponenterna ska bli verksamma.

Orienterbara verktygsbärare och nya settingdata

Settingdata SD42900 till SD42940 verkar inte på komponenterna till en eventuellt aktiv orienterbar verktygsbärare. Ett verktyg går dock alltid in med sin toltala resulterande längd (verktygslängd + förslitning + basmått) i beräkningen med en orienterbar verktygsbärare. Vid beräkningen av den resulterande totala längden tas det hänsyn till alla ändringar som förorsakas av settingsdata; dvs. vektorer för den orienterbara verktygsbäraren är oberoende av bearbetningsplanet.

Märk

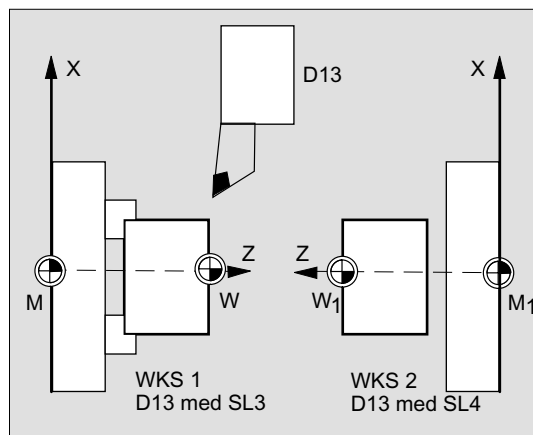
Ofta är det lämpligt vid användningen av en orienterbar verktygsbärare att definiera alla verktyg för ett ej speglat grundsystem, även de som endast används vid spegelbearbetning. Vid bearbetningen med speglade axlar vrids verktygsbäraren så att verktygets verkliga läge beskrivs riktigt. Alla verktygslängdkomponenterna verkar då automatiskt i den riktiga riktningen så att en styrning av värderingen av enskilda komponenter via settingdata beroende på speglingstillstånd för de enskilda axlarna blir överflödig.

Ytterligare användningsmöjligheter

Användningen av funktionaliteten orienterbar verktygsbärare kan också vara lämplig när på maskinen fysikaliskt ingen möjlighet planerats att vrida verktyg, men verktyg med olika orienteringar är fast installerade. Verktygsmätningen kan då göras enhetligt i en grundorientering och de för bearbetningen relevanta måtten resulterar genom vridningar av en virtuell verktygsbärare.

11.3.1 Spegla verktygslängder

Med inställda settingdata SD42900 \$SC_MIRROR_TOOL_LENGTH och SD42910 \$SC_MIRROR_TOOL_WEAR skilt från noll kan du spegla verktygslängdkomponenter och komponenter för basmättet med förslitningsvärden i deras tillhörande axlar.



SD42900 \$SC_MIRROR_TOOL_LENGTH

Settingdatum **skilt från** noll:

Verkygslängdkomponenterna (\$TC_DP3, \$TC_DP4 och \$TC_DP5) och komponenterna för basmått (\$TC_DP21, \$TC_DP22 och \$TC_DP23), vars tillhörande axlar är speglade, blir också speglade - genom förteckeninventering.

Förslitningsvärdena speglas **inte** med. Skulle dessa också speglas måste settingdatum SD42910 \$SC_MIRROR_TOOL_WEAR ställas in.

SD42910 \$SC_MIRROR_TOOL_WEAR

Settingdatum **skilt från** noll:

Förslitningsvärdena för verkygslängdkomponenterna, vars tillhörande axlar är speglade, blir också speglade - genom förteckeninventering.

11.3.2 Förteckenvärdering förslitning

Med inställda settingdata SD42920 \$SC_WEAR_SIGN_CUTPOS och SD42930 \$SC_WEAR_SIGN skilt från noll kan du invertera förteckenvärderingen för förslitningskomponenterna.

SD42920 \$SC_WEAR_SIGN_CUTPOS

Settingdatum **skilt från** noll:

Vid verktyg med relevant skärläge (svarv- och slipverktyg, verktygstyper 400) beror förteckenvärderingen för förslitningskomponenterna i bearbetningsplanet på skärläget. Vid verktygstyper utan relevant skärläge är detta settingdatum utan betydelse.

I följande tabell är de mått markerade med ett X vars förtecken inverteras via SD42920 (skilt från 0):

Skärläge	Längd 1	Längd 2
1		
2		X
3	X	X
4	X	
5		
6		
7		X
8	X	
9		

Märk

Förteckenvärderingen genom SD42920 och SD42910 är oberoende av varandra. När t.ex. förtecknet för en måttuppgift ändras genom båda settingdata förblir det resulterande förtecknet oförändrat.

SD42930 \$SC_WEAR_SIGN

Settingdatum **skilt från** noll:

Förtecknet för alla förslitningsmått inverteras. Det verkar både på verktygslängden och även på de övriga storlekarna som verktygsradie, rundningsradie osv.

Matas ett positivt förslitningsmått in blir verktyget därmed "kortare" och "tunnare", se Kapitel "Vertygskompensering, Specialbehandling", "Hur förändrade settingsdata blir verksamma".

11.3.3 Koordinatsystem i den aktiva bearbetningen (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)

Beroende på kinematiken i maskinen eller om det finns en orienterbar verktygsbärare överförs resp. transformeras de i ett av dessa koordinatsystem uppmätta förslitningsvärdena till ett lämpligt koordinatsystem.

Koordinatsystem för den aktiva bearbetningen

Ur de följande koordinatsystemen kan offsets framgå för den verktygslängd, som verktygslängdkomponenten förslitning, via det motsvarande G-kommandot i gruppen 56, räknar in i ett aktivt verktyg.

- Maskinkoordinatsystem (MKS)
- Baskoordinatsystem (BKS)
- Arbetsstyckskoordinatsystem (WKS)
- Vertygskoordinatsystem (TCS)
- Vertygskoordinatsystem för den kinematiska transformationen (KCS)

Syntax

TOWSTD
TOWMCS
TOWWCS
TOWBCS
TOWTCS
TOWKCS

Betydelse

TOWSTD:	Grundlägesvärde för kompenseringar i verktygslängden förslitningsvärde
TOWMCS:	Kompenseringar i verktygslängden i MKS
TOWWCS:	Kompenseringar i verktygslängden i WKS
TOWBCS:	Kompenseringar i verktygslängden i BKS
TOWTCS:	Kompenseringar i verktygslängden i verktygsbärarens referenspunkt (orienterbar verktygsbärare)
TOWKCS:	Kompenseringar i verktygslängden för verktygshuvudet (kinematisk transformation)

Ytterligare informationer

Särskiljningstecken

I den följande tabellen är de viktigaste särskiljningstecknen framställda:

G-kommando	Förslitningsvärde	Aktiv orienterbar verktygsbärare
TOWSTD:	Grundlägesvärde, verktygslängd	Förslitningsvärden underkastas vridningen.
TOWMCS:	Förslitningsvärde i MKS. TOWMCS är identisk med TOWSTD när ingen orienterbar verktygsbärare är aktiv.	Endast vektorn för den resulterande verktygslängden vrids utan hänsynstagande till förslitningen.
TOWWCS:	Förslitningsvärdet omräknas från WKS till MKS.	Verktogsvektorn beräknas utan hänsynstagande till förslitningen som vid TOWMCS.
TOWBCS:	Förslitningsvärdet omräknas från BKS till MKS.	Verktogsvektorn beräknas utan hänsynstagande till förslitningen som vid TOWMCS.
TOWTCS:	Förslitningsvärdet omräknas från verktygskoordinatsystemet till MKS.	Verktogsvektorn beräknas utan hänsynstagande till förslitningen som vid TOWMCS.

TOWWCS, TOWBCS, TOWTCS: Förslitningsvektorn adderas till verktygsvektorn.

Linjär transformation

Verktygslängden kan endast definieras förnuftigt i MKS när MKS framgår ur BKS genom en linjär transformation.

Ej linjär transformation

Är t.ex. med TRANSMIT en ej linjär transformation aktiv, då används vid angivande av MKS som önskat koordinatsystem automatiskt BKS.

Ingen kinematisk transformation och ingen orienterbar verktygsbärare

Är varken en kinematisk transformation eller en orienterbar verktygsbärare aktiv då faller så när som på WKS alla ytterligare fyra koordinatsystem ihop. Därmed skiljer sig endast WKS från de andra. Eftersom uteslutande verktygslängder ska värderas har translationer mellan koordinatsystemen ingen betydelse.

Litteratur:

Ytterligare informationer till verkygskompensering se:

Funktionshandbok Grundfunktioner; Verkygskompensering (W1)

Inräkning av förslitningsvärden

Settingdatum SD42935 \$SC_WEAR_TRANSFORM fastlägger vilken av de tre förslitningskomponenterna:

- Förslitning
- Summakompenseringar fin
- Summakompenseringar grov

som ska underkastas en vridning genom en adaptertransformation eller en orienterbar verktygsbärare när ett av de följande G-kommandona är aktivt:

- TOWSTD
Grundläge. För kompenseringar i verktygslängden.
- TOWMCS
Förlitningsvärden i maskinkoordinatsystemet (MKS).
- TOWWCS
Förlitningsvärden i arbetsstyckskoordinatsystemet (WKS).
- TOWBCS
Förlitningsvärden i baskoordinatsystemet (BKS).
- TOWTCS
Förlitningsvärden i verktygskoordinatsystemet i verktygshållarinfästningen (T verktygsbärarreferens)
- TOWKCS
Förlitningsvärden i koordinatsystemet för verktygshuvudet vid kinetisk transformation.

Märk

Värderingen av de enskilda förlitningskomponenterna (tillordning till geometriaxlarna, förteckenvärdering) påverkas av följande faktorer:

- Aktivt plan
 - Adaptertransformation
 - Settingdata:
 - SD42910 \$SC_MIRROR_TOOL_WEAR
 - SD42920 \$SC_WEAR_SIGN_CUTPOS
 - SD42930 \$SC_WEAR_SIGN
 - SD42940 \$SC_TOOL_LENGTH_CONST
 - SD42950 \$SC_TOOL_LENGTH_TYPE
-

11.3.4 Verkygslängd och planbyte

Med inställda settingdata SD42940 \$SC_TOOL_LENGTH_CONST skilt från noll kan du tillordna verktygslängdkomponenterna som längd, förlitning och basmått till geometriaxlarna för svarv- och slipverktyg vid ett planbyte.

SD42940 \$SC_TOOL_LENGTH_CONST

Settingdatum **skilt från** noll:

Tillordningen av verktygslängdkomponenterna (längd, förlitning och basmått) till geometriaxlarna vid byte av bearbetningsplanet (G17 - G19) förändras inte.

Följande tabell visar tillordningen av verktygslängdkomponenterna till geometriaxlarna för svarv- och slipverktyg (WZ-typ 400 till 599):

Innehåll	Längd 1	Längd 2	Längd 3
17	Y	X	Z

11.3 Verkygskompensering-specialbehandling

*)	X	Z	Y
19	Z	Y	X
-17	X	Y	Z
-18	Z	X	Y
-19	Y	Z	X

*) Varje värde skilt från 0 som inte är lika med ett av det sex uppförda värdena värderas som värdet 18.

Följande tabell visar tillordningen av verktygslängdkomponenterna till geometriaxlarna för alla andra verktyg (WZ-typ < 400 resp. > 599):

Bearbetningsplan	Längd 1	Längd 2	Längd 3
*)	Z	Y	X
18	Y	X	Z
19	X	Z	Y
-17	Z	X	Y
-18	Y	Z	X
-19	X	Y	Z

*) Varje värde skilt från 0 som inte är lika med ett av det sex uppförda värdena värderas som värdet 17.

Märk

Vid framställningen i tabellerna utgås ifrån att geometriaxlarna upp till 3 betecknas med X, Y, Z. För tillordningen av en kompensering till en axel är inte axelbeteckningen utan axelordningsföljden utslagsgivande.

11.4 Online-verkygskompensering

11.4.1 Definiera polynomfunktion (FCTDEF)

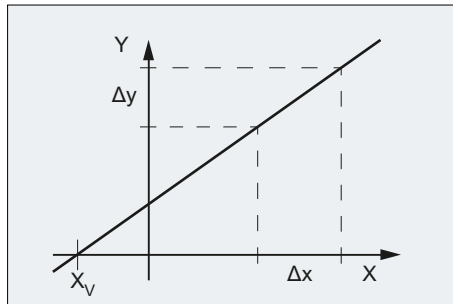
Vissa skärpningsstrategier (t. ex. skärprulle) utmärker sig därigenom att slipskivan minskar kontinuerligt (linjärt) i radien med ansättningen av skärprullen. Härtill behöver man en linjär funktion mellan ansättningen av skärprullen och skrivandet av slitagevärdet för respektive längd. Definitionen av den linjära funktionen sker via den fördefinierade proceduren FCTDEF(...) för polynomfunktioner till maximalt tredje graden.

Linjär ekvation

$$y = f(x) = a_0 + a_1 \cdot x_1$$

a_1 : Linjens lutning, med $a_1 = \Delta x / \Delta y$

a_0 : Förflyttning av linjen längs X-axeln, med $a_0 = -a_1 \cdot X_v$



Syntax

FCTDEF(<Func>,<LLimit>,<ULimit>,<a0>,<a1>,<a2>,<a3>)

Betydelse

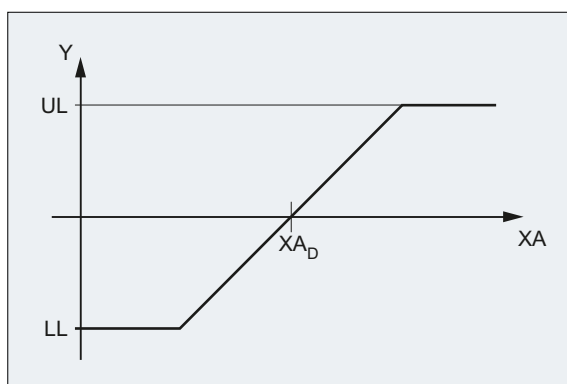
FCTDEF(...):	Definition av en polynomfunktion för PUTFTOCF(...): $y = f(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3$	
<Func>:	Funktionsnummer	
	Datotyp:	INT
	Värdeområde:	1, 2, 3
<LLimit>:	Undre gränsvärde	
	Datotyp:	REAL
<ULimit>:	Övre gränsvärde	
	Datotyp:	REAL
<a0>,<a1>,<a2>,<a3>:	Koefficienter i polynomfunktionen	
	Datotyp:	REAL

Exempel

Fastlägganden

- Funktionsnummer: 1
- Undre och övre gränsvärde: -100, 100
- Kurvans stigning: $a_1 = 1$
- Arbetspunkten ska ligga i mitten av kurvan. Kurvan måste för detta förflyttas i negativ Y-riktning med hjälp av börpositionen till axeln XA i WKS vid tidpunkten för funktionsdefinitionen i NC-programmet: $a_0 = -a_1 * XA_D = -1 * \AA_IW
- $a_2 = a_3 = 0$

Kurva



UL Övre gränsvärde

LL Undre gränsvärde

XA_D Börvärdet till axeln XA i WKS vid tidpunkten för funktionsdefinitionen i NC-programmet

Programmering

Programkod	Kommentar
FCTDEF(1,-100,100,-\$AA_IW[XA],1)	; Funktionsdefinition

11.4.2 Skriva online-verkygskompensering, kontinuerligt (PUTFTOCF)

Med den fördefinierade proceduren PUTFTOCF(...) görs en online-verkygskompensering via en tidigare med FCTDEF(...) (Sida 415) definierad polynomfunktion.

Märk

Online-verkygskompenseringen kan också göras via en synkronaktion.

Ytterligare informationer se funktionshandboken Synkronaktioner.

Syntax

```
PUTFTOCF (<Func>, <RefVal>, <ToolPar>, <Chan>, <Sp>)
```

Betydelse

PUTFTOCF (...):	Skriva online-verktygskompenseringen, kontinuerligt blockvis med hjälp av den med FCTDEF(...) definierade polynomfunktionen	
<Func>:	Funktionsnummer, fastställt vid funktionsdefinitionen med FCTDEF(...)	
	Datotyp:	INT
	Värdeområde:	1, 2, 3
<RefVal>:	Referensvärde från vilket kompenseringen ska härledas (t.ex. börvärde för en axel).	
	Datotyp:	VAR REAL
<ToolPar>:	Nummer för slitageparametern (längd 1, 2 eller 3), i vilken kompenseringsvärdet ska inberäknas	
	Datotyp:	INT
<Chan>:	Nummer på den kanal i vilken online-verktygskompenseringen ska bli verksam	
	Observera: Erforderligt endast när kompenseringen inte ska bli verksam i den aktiva kanalen.	
	Datotyp:	INT
<Sp>:	Nummer på den spindel för vilken online-verktygskompenseringen ska bli verksam	
	Observera: Erforderlig endast när i stället för det aktiva verktyget som är under användning en inte aktiv slipskiva ska kompenseras.	
	Datotyp:	INT

11.4.3 Skriva online-verktygskompensering, diskret (PUTFTOC)**Funktion**

Med den fördefinierade proceduren PUTFTOC(...) görs en online-verktygskompensering med ett fast kompenseringsvärde.

Syntax

```
PUTFTOC (<CorrVal>, <ToolPar>, <Chan>, <Sp>)
```

Betydelse

PUTFTOC (...):	Skriva online-verktygskompensering	
<CorrVal>:	Kompenseringsvärde som adderas till slitageparametern	
	Datotyp:	REAL
<ToolPar>:	Nummer för slitageparametern (längd 1, 2 eller 3), i vilken kompenseringsvärdet ska inberäknas	
	Datotyp:	INT

<Chan>:	Nummer på den kanal i vilken online-verktygskompenseringen ska bli verksam	
	Observera: Erforderligt endast när kompenseringen inte ska bli verksam i den aktiva kanalen.	
	Datatyp:	INT
<Sp>:	Nummer på den spindel för vilken online-verktygskompenseringen ska bli verksam	
	Observera: Erforderlig endast när i stället för det aktiva verktyget som är under användning en inte aktiv slipskiva ska kompenseras.	
	Datatyp:	INT

11.4.4 Koppla till/från online-verktygskompenseringen (FTOCON/FTOCOF)

Med G-kommandona FTOCON och FTOCOF kopplas online-verktygskompenseringen till resp. från.

Syntax

```
FTOCON
...
FTOCOF
```

Betydelse

FTOCON:	Koppla till online-verktygskompensering Kommandot måste programmeras i den kanal i vilken online-verktygskompenseringen ska kopplas till.
FTOCOF:	Koppla från online-verktygskompenseringen Kommandot måste programmeras i den kanal i vilken online-verktygskompenseringen ska kopplas från. Observera: Med FTOCOF körs verktygskompenseringen inte ut ytterligare. I skärspecifika kompenseringsdata bibehålls dock det med PUTFTOC/PUTFTOCF inberäknade värdet. För att slutgiltigt inaktivera online-verktygskompenseringen måste efter FTOCOF också ett till-/bortval av verktyget (T...) göras.

11.5 3D-verkygstradiekompensering

11.5.1 Välja 3D-verkygstradiekompensering för 3D-periferifräsningen (CUT3DC, CUT3DCD, ISD)

3D-verkygstradiekompenseringen (3D-WRK) för 3D-periferifräsningen utan hänsyn till begränsningsytor väljs med det modalt verksamma G-kommandot CUT3DC resp. CUT3DCD.

Den egentliga aktiveringen sker med G41 resp. G42. Verkygstradiekompenseringen kopplas från med G40.

Syntax

```
G41/G42 ORIC/ORID ISD=... CUT3DC/CUT3DCD CDOF2 X... Y... Z...
...
G40 X... Y... Z...
```

Betydelse

CUT3DC:	3D-WRK för periferifräsningen (endast vid aktiv 5-axel-transformation)	
CUT3DCD:	Till ett differensverktyg relaterad 3D-WRK för periferifräsningen (endast vid aktiv 5-axel-transformation) Radiedifferensen fastläggs med WZ-parametern \$TC_DP15.	
G41/G42 X... Y... Z...:	Koppla till WZ-radiekompensering	
	G41:	Verkygstradiekompensering till vänster om konturen
	G42:	Verkygstradiekompensering till höger om konturen
	Observera: Tillkopplingen måste göras i ett linjärt block (G0/G1).	
CDOF2:	Koppla från kollisionsövervakning för 3D-periferifräsning	
ORIC/ORID:	Via G-kommandona ORIC och ORID fastläggs beteendet vid orienteringsändringar i ytterhörnen.	
	ORIC:	Orienteringsändringar vid ytterhörn överlagrar det cirkelblock som ska infogas.
	ORID:	Orienteringsändringar vid ytterhörn utförs före infogandet av cirkelblocket.
ISD=<Value>:	Med adressen ISD kan vid periferifräsning och aktiv 3D-verkygstradiekompensering nermatningsdjupet för verktyget ändras.	
	<Value>:	Längd för nermatningsdjupet
G40 X... Y... Z...:	Koppla från verkygstradiekompensering Observera: Frånkopplingen måste göras i ett linjärt block (G0/G1) med geometriaxelrörelser.	

Märk

G-kommandon för val av 3D-WRK utvärderas i uppsökningsblocket, dvs. i normalfall i det block som innehåller G41 eller G42.

G41 resp. G42 kan också vara programmerade i block utan förflyttningsrörelse i de för kompenseringen relevanta geometriaxlarna. I detta fall är uppsökningsblocket det första förflyttningsblocket som följer på ett sådant block.

Ett byte av 3D-WRK-varianten vid aktiv verkytgsradiekompensering ignoreras utan larm.

Exempel

Programkod	Kommentar
	; Definition av verktyget D1:
\$TC_DP1[1,1]=120	; Typ (pinnfräs)
\$TC_DP3[1,1]=20	; Längdkompenseringsvektor
\$TC_DP6[1,1]=8	; Radie
N10 X0 Y0 Z0 T1 D1 F12000	; Val av verktyget.
N20 TRAORI(1)	; Tillkoppling av transformationen.
N30 G42 ORIC ISD=10 CUT3DC G64 X30	; Aktivering av 3D-periferifräsningen,
	; Orienteringsändringar vid ytterhörnen kontinuerligt,
	; Nermatningsdjup: 10 mm
N40 ORIWKS A30 B15	; Orienteringsändring i ett hörn genom angivande av axelpositioner.
N50 Y20 A3=1 C3=1	; Förflyttningsblock med orienteringsändring,
	; Angivande av orienteringen med riktningsvektor.
N60 X50 Y30	; Förflyttningsblock med konstant orientering.
N70 Y50 A3=0.5 B3=1 C3=5	; Förflyttningsblock med orienteringsändring.
N80 M63	; Block utan förflyttningsinformation.
N90 X0 ISD=20	; Förflyttningsblock med ändring av nermatningsdjup.
N100 G40 Y0	; Inaktivering av verkytgsradiekompensering.
N110 M30	

Ytterligare informationer

Ban och orientering

Den här använda varianten av periferifräsning är realiserad genom angivande av en bana (styrinje) och den tillhörande orienteringen. Vid denna typ av bearbetning är verkytgsformen utan betydelse på banan. Radien för verkytgets ingreppspunkt är ensam bestämmande.

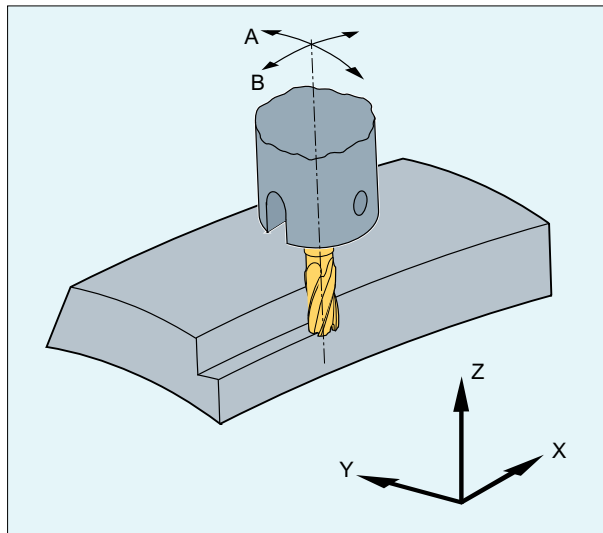


Bild 11-1 Periferifräsning

Uppsökningsbeteende

Uppsökningsbeteendet är vid 3D-varianterna för verktüysradiekompenseringen alltid NORM.

Beteende vid ytterhörn

I ytterhörnan vid periferifräsning med 3D-WRK utvärderas analogt till förhållandena vid 2½D-WRK G-kommandona i grupp 18 (hörnbeteende verktüyskompensering):

- G450: Övergångscirkel (verktüget går runt arbetsstyckshörn längs en cirkelbana)
I motsats till lösningen vid 2½D-WRK är det infogade konturelementet vid ett ytterhörn alltid en cirkel med radien 0, på vilken verktüysradiekompenseringen verkar så som på varje annan programmerad bana. Det är inte möjligt att infoga koniska sektioner i stället för cirklar. Adressen DISC har därför i detta fall ingen betydelse och utvärderas inte.
- G451: Skärningspunkt för ekvidistanterna (verktüget friskär i arbetsstyckshörnet)
Skärningspunkten bestäms genom att offsetkurvorna för de båda deltagande blocken förlängs och deras skärningspunkt i planet vinkelrätt mot verktüysorienteringen i hörnet bestäms.

Skärningspunktbeteendet (G451) används inte, när mellan de medverkande förflyttningsblocken minst ett block är infogat, som innehåller en ändring i verktüysorienteringen. I ett sådant fall infogas alltid en cirkel i hörnet.

Beteende vid orienteringsändringar i ytterhörn

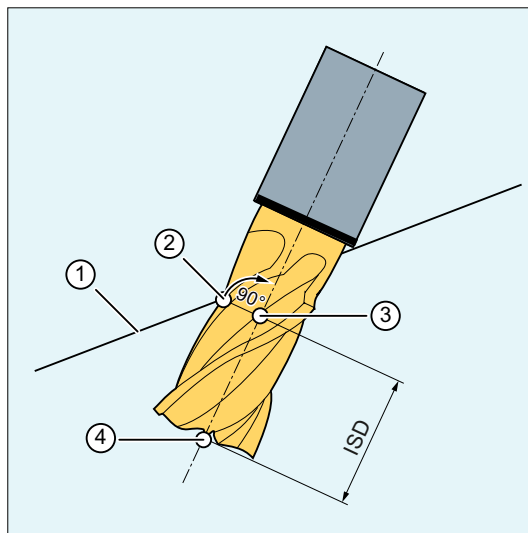
Via G-kommandona ORIC och ORID kan fastläggas om orienteringsändringarna, som programmerades mellan de båda förflyttningsblocken som bildar hörnen, utförs före början av det infogade cirkelblocket (ORID) eller samtidigt med detta (ORIC).

Nermatningsdjup

Fräsens nermatningsdjup är avståndet för fräshjälppunkten från verktüygsspetsen.

Fräshjälppunkten är den lodräta projektionen av fräsbearbetningspunkten på den programmerade banan på verktüyslängdsaxeln.

Med nermatningsdjupet blir på så sätt läget för bearbetningspunkten inställt på verktügets mantelyta.



- ① Programmerad bana
- ② Fräsbearbetningspunkt
- ③ Fräshjälpunkt
- ④ Frässpets
- ISD Nermatningsdjup (InSection Depth)

Bild 11-2 Nermatningsdjup

Verkygstradiekompenisering relaterad till ett differensverktyg

Den till ett differensverktyg relaterade 3D-WRK för periferifräsningen aktiveras av kommandot CUT3DCD. Den ska användas när den programmerade konturen är relaterad till medelpunktsbanan för ett normverktyg och bearbetningen görs med ett från detta avvikande verktyg. Vid beräkningen av 3D-verkygstradiekompeniseringen inberäknas då endast förslitningsvärdet för radien för det aktiva verktyget (\$TC_DP15) och de eventuellt programmerade verktygsoffsets OFFN och TOFFR. Grundradien (\$TC_DP6) för det aktiva verktyget inberäknas **inte**.

Fickfräsning med sneda sidoväggar för periferifräsning med CUT3DC

Vid denna 3D-verkygstradiekompenisering kompenseras en avvikelse i fräsradien genom ansättning i riktningen för ytnormalen till den yta som ska bearbetas. Därvid förblir planet, i vilket fräsens frontsida ligger, oförändrat när nermatningsdjupet ISD förblivit lika. En fräs med t.ex. mindre radie gentemot ett normverktyg skulle då inte nå ner till fickbotten, som också utgör begränsningsytan. För en automatisk ansättning av verktyget måste denna begränsningsyta vara känd för styrningen, se Kapitel "3D-periferifräsning under hänsynstagande till en begränsningsyta (CUT3DCC, CUT3DCCD) (Sida 428)".

Advanced Surface / Top Surface**Märk**

Vid användning av verktygssradiekompenseringen CUT3DCD i kombination med det licenspliktiga tillvalet "Advanced Surface" eller "Top Surface" ska inställningsrekommendationerna gällande "Advanced Surface" / "Top Surface" beaktas!

För kontroll av inställda data står speciella kontrollprogram till förfogande via SIOS-portalen:

- Kontrollprogram för Advanced Surface (<https://support.industry.siemens.com/cs/ww/en/view/78956392>)
- Kontrollprogram för Top Surface (<https://support.industry.siemens.com/cs/ww/en/view/109738423>)

11.5.2 Välja 3D-verktygssradiekompensering för 3D-ändplansfräsning (CUT3DF, CUT3DFS, CUT3DFF, CUT3DFD)

3D-verktygssradiekompenseringen (3D-WRK) för 3D-ändplansfräsningen väljs med det modalt verksamma G-kommandot CUT3DF, CUT3DFS, CUT3DFF resp. CUT3DFD.

Den egentliga aktiveringen sker med G41 resp. G42.

För att kunna beräkna verktygssradiekompenseringen, är vid 3D-ändplansfräsningen definitionen för ytnormalen till det plan som ska bearbetas nödvändig. Detta måste göras i blocket med G41 resp. G42 via adresserna A4, B4, C4 och A5, B5, C5.

Verktygssradiekompenseringen kopplas från med G40.

Syntax

```
G41/G42 ORIC/ORID CUT3DF/CUT3DFS/CUT3DFF/CUT3DFD X... Y... Z... A4=... B4=...
C4=... A5=... B5=... C5=...
...
G40 X... Y... Z...
```

Betydelse

CUT3DFS:	3D-WRK för planfräsningen med konstant orientering. Verktygssorienteringen är fastlagd genom G17 - G19 och påverkas inte av frames.
CUT3DFF:	3D-WRK för planfräsningen med konstant orientering. Verktygssorienteringen är fastlagd genom G17 - G19 och eventuellt av en frame-vriden riktning.
CUT3DF:	3D-WRK för planfräsning med orienteringsändring (endast vid aktiv 5-axel-transformation).

CUT3DFD:	Till ett differensverktyg relaterad 3D-WRK för planfräsningen med orienteringsändring (endast vid aktiv 5-axel-transformation). Radiedifferensen fastläggs med WZ-parametern \$TC_DP15. Observera: CUT3DFD är möjlig endast i kombination med "Glättning av ytnormalen vid 3D-planfräsning". Denna aktiveras via anrop av den licenspliktiga funktionen "Top Surface" med hjälp av CYCLE832(...) (Sida 771).	
G41/G42 X... Y... Z...:	Koppla till verktygskompensering Beteendet vid G41 och vid G42 är identiskt vid 3D-planfräsning. Observera: Tillkopplingen måste göras i ett linjärt block (G0/G1).	
A4/5=... B4/5=... C4/5=...:	Definition för ytnormalen till det plan som ska bearbetas	
	A4=... B4=... C4=...:	Definition i blockbörjan
	A5=... B5=... C5=...:	Definition i blockslutet
ORIC/ORID:	Via G-kommandona ORIC och ORID fastläggs beteendet vid orienteringsändringar i ytterhörnen.	
	ORIC:	Orienteringsändringar vid ytterhörn överlagrar det cirkelblock som ska infogas.
	ORID:	Orienteringsändringar utförs före cirkelblocket.
G40 X... Y... Z...:	Koppla från verktygsradiekompensering Observera: Frånkopplingen måste göras i ett linjärt block (G0/G1) med geometriaxelrörelser.	

Märk

G41 resp. G42 kan också vara programmerade i block utan förflyttningsrörelse i de för kompenseringen relevanta geometriaxlarna. I detta fall är uppsökningsblocket det första förflyttningsblocket som följer på ett sådant block.

Ett byte av 3D-WRK-varianten vid aktiv verktygsradiekompensering ignoreras utan larm.

Exempel**Exempel 1: 3D-planfräsning med CUT3DF**

Programkod	Kommentar
N10	; Definition av verktyget D1:
N20 \$TC_DP1[1,1]=121	; Verktygstyp (torusfräs)
N30 \$TC_DP3[1,1]=20	; Längdkompensering
N40 \$TC_DP6[1,1]=5	; Radie
N50 \$TC_DP7[1,1]=3	; Rundningsradie
N60	
N70	
N80 X0 Y0 Z0 A0 B0 C0 G17 T1 D1 F12000	; Val av verktyget.

Programkod	Kommentar
N90 TRAORI (1)	; Välja orienteringstransformation.
N100 B4=-1 C4=1	; Definition av planet.
N110 G41 ORID CUT3DF G64 X10 Y0 Z0	; Koppla till verktygskompensering.
N120 X30	
N130 Y20 A4=1 C4=1	; Ytterhörn, nydefinition av planet.
N140 B3=1 C3=5	; Orienteringsändring med ORID.
N150 B3=1 C3=1	; Orienteringsändring med ORID.
N160 X-10 A5=1 C5=2 ORIC	
N170 A3=-2 C3=1	; Orienteringsändring med ORIC
N180 A3=-1 C3=1	; Orienteringsändring med ORIC
N190 Y-10 A4=-1 C4=3	; Nydefinition av planet.
N200 X-20 Y-20 Z10	; Innerhörn med inledningsblock.
N210 X-30 Y10 A4=1 C4=1	; Innerhörn, nydefinition av planet.
N220 A3=1 B3=0.5 C3=1.7	; Orienteringsändring med ORIC.
N230 X-20 Y30 A4=1 B4=-2 C4=3 ORID	
N240 A3 = 0.5 B3=-0.5 C3=1	; Orienteringsändring.
N250 X0 Y30 C4=1	; Banrörelse, nytt plan, ; Orientering med relativprogrammering.
N260 BSPLINE X20 Z15	; Splinebörjan, relativprogrammering av orienteringen
N270 X30 Y25 Z18	; förblir aktiv under 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	; Helix, orientering med axelprogrammering.
N320 G1 X0 A3=-0.123 B3=0.456 C3=2.789 B4=-1 C4=5 B5=-1 C5=2	; Banrörelse, orientering, ej konstant plan.
N330 X-20 G40	; Frånkoppling av verktygsradiekompenseringen.
N340 M30	

Exempel 2: Av ett CAD-system skapat NC-program (avsnitt) med CUT3DFD

Programkod	Kommentar
N01 G710	
N03 T="12"	
N06 S5305 M03	
N07 G642	
	; Uppsökning av startpositionen i MKS under hänsynstagande till verktygslängden.
G00 G90 X-250.62787 Y-38.37944 A=DC(253.12719) B-12.49543	
G00 G90 Z251.80052	
	; Slut på positioneringen i MKS.
	;
TRAORI (1)	; Välja orienteringstransformation.

11.5 3D-verkyggradiekompensering

Programkod	Kommentar
G500	
D1	
CYCLE832(0.01, _TOP_SURFACE_SMOOTH_ON + _ORI_FI- NISH, 1)	; Anrop CYCLE832 med: ; Konturtolerans = 0,01 mm, ; Bearbetningstyp: Top Surface med glättning, ; Finbearbetning med inmatning av en orienter- ingstolerans, ; Orienteringstolerans = 1 grad
CUT3DFD	
N08 G90 G94	
N09 G00 X-269.21195 Y128.32027 Z1.18577 A3=-.216361688 B3=.934284397 C3=-.283373051	; Blocken N09 till N10 är den snabba andelen av uppsökningsrörelsen till arbetsstycket med kon- stant orientering.
N10 G00 X-251.90301 Y53.57752 Z23.85561	
N11 G01 X-247.57578 Y34.89183 Z29.52308 F50000.00000	; I blocken N11 till N21 realiseras den långsam- ma andelen av uppsökningsrörelsen. Verktyget be- finner sig redan nära arbetsstycket, dessutom är nu formbyggnads-inställningarna (t.ex. COMP- SURF) från CYCLE832 aktiva (på grund av aktiv G01). Banvägen för denna så kallade insvängnings- fas för formbyggnads-beteendet bör uppgå till ungefär 1000 gånger konturtoleransen (i detta ex- empel alltså 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	
N22 G41 X-248.06237 Y30.32400 Z9.36695 A5=-.060431854 B5=.973045457 C5=-.222554556 A3=-.214974689 B3=.929398552 C3=-.300007025 F1061.03295	; Från och med N22 är ytnormalen för första gång- en fullständigt definierat över hela blocket (dvs. ytnormalen i slutet av inledningsblocket N21 finns, därmed är ytnormalen i början av blocket N22 och ytnormalen i slutet av blocket N22). Därmed är förutsättningen för tillkoppling- en av verkygskompenseringen med G41/G42 upp- fyllt.

Programkod	Kommentar
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	
...	

Märk

Vid 3D-planfräsning med CUT3DFD är för tillkopplingen av verkygskompenseringen med G41/G42 definitionen av ytnormalen nödvändig. Programmeringen av G41/G42 utan definition av ytnormalen leder till utmaning av ett larm.

Ytterligare informationer

3D-ändplansfräsning

För denna typ av 3D-fräsning behövs beskrivningen rad för rad av 3D-banorna på arbetsstycksytan. Beräkningarna görs under hänsynstagandet till verktygsform och verktygsdimensioner vanligen i CAM. Postprocessorn skriver i detaljprogrammet – förutom NC-blocken – verktygsorienteringarna (vid aktiv 5-axeltransformation) och G-kommandot för den önskade 3D-verkygskompenseringen. Härigenom har maskinoperatören möjligheten – avvika från de för beräkningen av NC-banorna använda verktygen – att använda något mindre verktyg.

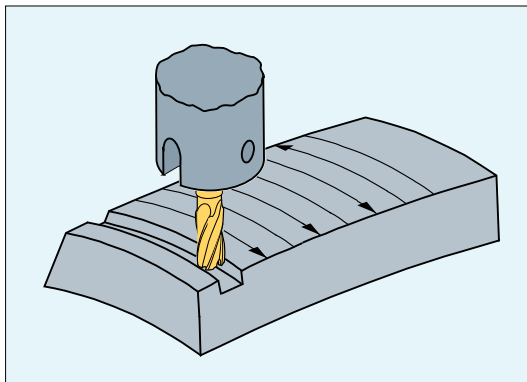


Bild 11-3 Planfräsa

Uppsökningsbeteende

Uppsökningsbeteendet är vid 3D-varianterna för verktygsradiekompenseringen alltid NORM.

Beteende vid ytterhörn

Ytterhörn behandlas vid planfräsning som cirklar med radien 0, varvid cirkelplanet spänns upp av sluttangenten till den första och starttangenten till det andra blocket. Därmed är också en orienteringsändring möjlig vid blockövergången. I ett ytterhörn infogas därmed alltid en cirkel som konturelement. Skärningspunkt beteende står vid planfräsning inte till förfogande.

Beteende vid orienteringsändringar i ytterhörn

Via G-kommandona ORIC och ORID kan fastläggas om orienteringsändringarna, som programmerades mellan de båda förflyttningsblocken som bildar hörnen, utförs före början av det infogade cirkelblocket (ORID) eller samtidigt med detta (ORIC).

Verkygstradiekompensering relaterad till ett differensverktyg

Den till ett differensverktyg relaterade 3D-verkygstradiekompenseringen väljs av kommandot CUT3DFD. Den ska användas när den programmerade konturen är relaterad till medelpunktsbanan för ett normverktyg och bearbetningen görs med ett från detta avvikande verktyg. Vid beräkningen av 3D-verkygstradiekompenseringen inberäknas då endast förslitningsvärdet för radien för det aktiva verktyget (\$TC_DP15) och de eventuellt programmerade verktygsoffsets OFFN och TOFFR. Grundradien (\$TC_DP6) för det aktiva verktyget inberäknas **inte**.

3D-plansfräsning med CUT3DFD är möjlig endast i kombination med "Glättning av ytnormalen vid 3D-planfräsning". Denna aktiveras via anrop av den licenspliktiga funktionen "Top Surface" med hjälp av CYCLE832(...) (Sida 771). Aktiveringen måste göras **före** tillkopplingen av verkygskompenseringen med G41/G42 och närmare bestämt inte direkt före verkygsgreppet, utan redan en banvägslängd tidigare, som motsvarar ungefär 1000 gånger konturtoleransen (t. ex. 1000 x 0,01 mm = 10 mm). Inaktiveringen måste förlöpa i omvänd ordningsföljd: först fränkoppling av verkygskompenseringen med G40, sedan, efter en banvägslängd, som motsvarar ungefär 1000 gånger konturtoleransen, inaktiveringen med t.ex. CUT2D (eller liknande).

För att kunna använda "Glättning av ytnormalen vid 3D-planfräsning", är dessutom frikoppling av funktionen "Interpolering av ytnormalen via polynom" nödvändig:

MD28291 \$MC_MM_SMOOTH_SURFACE_NORMALS = TRUE

Märk

För 3D-planfräsning med CUT3DFD i kombination med "Top Surface" ska inställningsrekommendationerna gällande "Top Surface" beaktas!

För kontroll av inställda data står speciella kontrollprogram till förfogande (<https://support.industry.siemens.com/cs/ww/en/view/109738423>) via SIOS-portalen.

11.5.3 3D-periferifräsning under hänsynstagande till en begränsningsyta (CUT3DCC, CUT3DCCD)

Vid 3D-periferifräsning med kontinuerlig eller konstant förändring av verkygsorienteringen programmeras ofta verktygets medelpunktsbana för ett definierat normverktyg. Eftersom i praktiken ofta inte de passande normverktygen står till förfogande kan ett verktyg som inte avviker alltför mycket från ett normverktyg ($\leq 5\%$) användas.

Med CUT3DCCD tas för ett reellt differensverktyg hänsyn till en begränsningsyta som skulle beskriva det programmerade normverktyget. NC-programmet beskriver mittpunktsbanan för normverktyget.

Med CUT3DCC tas vid användning av cylindriska verktyg hänsyn till en begränsningsyta, som det programmerade normverktyget hade uppnått. NC-programmet beskriver konturen på bearbetningsytan.

Ytnormalvektorn för begränsningsytan anges som vid 3D-planfräsning med A4, B4, C4 och A5, B5, C5.

Syntax

```
G41/G42 CUT3DCCD/CUT3DCC CDOF2 X... Y... Z... A4=... B4=... C4=... A5=... B5=...
C5=...
...
G40 X... Y... Z...
```

Betydelse

CUT3DCCD:	3D-WRK för periferifräsningen under hänsynstagande till en begränsningsyta med differensverktyg på verktygets medelpunktsbana: Ansättning mot begränsningsytan	
CUT3DCC:	3D-WRK för periferifräsningen under hänsynstagande till en begränsningsyta med 3D-radiekompensering: Kontur på bearbetningsytan	
G41/G42 X... Y... Z...:	Koppla till WZ-radiekompensering	
	G41:	Verkygstradiekompensering till vänster om konturen
	G42:	Verkygstradiekompensering till höger om konturen
	Observera: Tillkopplingen måste göras i ett linjärt block (G0/G1).	
CDOF2:	Koppla från kollisionsovervakning för 3D-periferifräsning	
A4/5=... B4/5=... C4/5=...:	Definition av ytnormalen till begränsningsytan	
	A4=... B4=... C4=...:	Definition i blockbörjan
	A5=... B5=... C5=...:	Definition i blockslutet
G40 X... Y... Z...:	Koppla från verkygstradiekompensering Observera: Frånkopplingen måste göras i ett linjärt block (G0/G1) med geometriaxelrörelser.	

Märk

G-kommandon för val av 3D-WRK utvärderas i uppsökningsblocket, dvs. i normalfall i det block som innehåller G41 eller G42.

G41 resp. G42 kan också vara programmerade i block utan förflyttningsrörelse i de för kompenseringen relevanta geometriaxlarna. I detta fall är uppsökningsblocket det första förflyttningsblocket som följer på ett sådant block.

Ett byte av 3D-WRK-varianten vid aktiv verkygstradiekompensering ignoreras utan larm.

Exempel

Programkod	Kommentar
N10 \$TC_DP1[1,1]=120	; cylinderfräs
N20 \$TC_DP6[1,1]=10	
N30 \$TC_DP15[1,1]=-3	
...	
; Bearbetning med cylinderfräs och CUT3DCCD	
N110 TRAORI	; Tillkoppling av transformationen.
N120 A4=0 B4=0 C4=1	; Definition av ytnormalen till begränsningsytan i blockbörjan.
N130 X0 Y0 Z0 A0 C0 T1 D1 F20000	
N140 X10 Y0 Z0 G41 CUT3DCCD CDOF2 G64	; Aktivering av 3D-periferifräsningen under hänsynstagande till begränsningsytan + frånkoppling av kollisionsovervakning.
N150 X20	
N160 X30 A45	; Trubbig vinkel ==> ingen ansättning
N170 X40 A-45	; Spetsig vinkel ==> ansättning
N180 X55	
N190 Y10 Z10	; Rörelse i verktygsriktning.
N200 Y20	
N210 C45	; Ren orienteringsändring.
N220 Y30 C90	
N230 A5=-1 B5=0 C5=2 Y40	; Ändring av ytan.
N240 Y50 G40	; Inaktivering av verktygsradiekompensering.
...	

Ytterligare informationer

Verktygstyp

Verktygstypen (verktygsparameter \$TC_DP1) utvärderas. Endast fräsverktyg med cylindriskt skaft (cylinder- eller pinnfräs, torusfräs och som gränsfall den cylindriska sänkfräsen) är tillåtna. Det motsvarar verktygstyperna 1 - 399 med undantag av numren 111 och 155 till 157.

Normverktyg med hörnrundning

Normverktygets hörnrundning beskrivs av verktygsparametern \$TC_DP7. Ur verktygsparametern \$TC_DP16 resulterar avvikelser i hörnrundningen vid det reella verktyget jämfört med normverktyget.

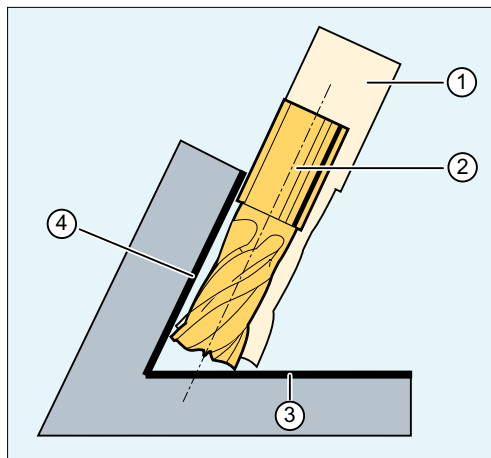
Exempel: Torusfräs med förminskad radie gentemot normverktyget

Verktygstyp	Skafttradie (R)	Hömradie (r)
Normverktyg med hörnrundning	$R = \$TC_DP6$	$r = \$TC_DP7$
Reellt verktyg med hörnrundning Verktygstyper 121 och 131 torusfräs (skaftfräs med hörnrundning)	$R' = \$TC_DP6 + \$TC_DP15 + OFFN$	$r' = \$TC_DP7 + \TC_DP16

I detta exempel är både \$TC_DP15 + OFFN och även \$TC_DP16 negativa.

3D-WRK med CUT3DCCD: Verktygets medelpunktsbana med ansättning fram till begränsningsytan

Används ett verktyg som i jämförelse med det passande normverktyget uppvisar en mindre radie då förs en i längsriktningen ansatt fräs vidare tills denna åter berör fickbotten. Därmed fräses hörnet som bildas av bearbetnings- och begränsningsytan ur, så långt som verktyget tillåter. Det rör sig därvid om ett blandat bearbetningssätt av periferi- och planfräsning. Analogt till ett verktyg med förminskad radie, ansätts vid ett verktyg med förstord radie motsvarande i den motsatta riktningen.



- ① Normverktyg
- ② Verktyg med mindre radie ansatt fram till begränsningsytan
- ③ Begränsningsyta
- ④ Bearbetningsyta

Gentemot alla andra verktygskompenseringar i G-gruppen 22 har en för CUT3DCCD angiven verktygsparameter \$TC_DP6 ingen betydelse för verktygsradien och påverkar inte den resulterande kompenseringen. Kompenseringsoffseten resulterar ur summan av förslitningsvärdet för verktygsradien (verktygsparameter \$TC_DP15) och en för beräkning av den vinkelräta offseten till begränsningsytan programmerad verktygsoffset OFFN.

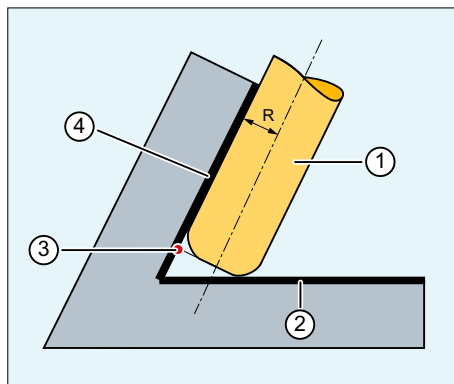
Om den yta som ska bearbetas ligger till vänster eller till höger om banan framgår inte av det skapade detaljprogrammet. Det utgår därför från en positiv radie och ett negativt förslitningsvärde för originalverktyget. Ett negativt förslitningsvärde beskriver alltid ett verktyg med förminskad diameter.

Användning av cylindriska verktyg

Vid användningen av cylindriska verktyg är en ansättning erforderlig endast när bearbetningsytan och begränsningsytan bildar en spetsig vinkel (mindre är 90 grader). Används torusfräs (skaftfräs med hörnrundning) då kräver detta en ansättning i verktygets längsriktning både vid spetsiga och även vid trubbiga vinklar.

3D-WRK med CUT3DCC: Kontur på bearbetningsytan

Är CUT3DCC aktiv med en torusfräs så hänför sig den programmerade banan till en fiktiv cylinderfräs med samma diameter. Den härur resulterande banareferenspunkten är framställd i följande bild vid användning av en torusfräs.



- ① Torusfräs
- ② Begränsningsyta
- ③ Banreferenspunkt
- ④ Bearbetningsyta
- R Skafradie (verkygstradie)

Det är tillåtet att en vinkel mellan bearbetnings- och begränsningsytan också inom ett block övergår från en spetsig till en trubbig vinkel och omvänt.

Gentemot normverkyget får det använda reella verkyget vara både större och även mindre. Därvid får den resulterande hörnradien inte bli negativ och förtecknet för den resulterande verkygstradien måste bibehållas.

Vid CUT3DCC hänför sig NC-detaljprogrammet till konturen på bearbetningsytan. Härvid anlitas som vid den vanliga verkygstradiekompenseringen den totala radien, som är sammansatt av följande komponenter:

- Verkygstradie (verkygsparemeter \$TC_DP6)
- Förslitningsvärde (verkygsparemeter \$TC_DP15)
- En för beräkning av den vinkelräta offseten till begränsningsytan programmerad verkygsoffset OFFN.

Läget för begränsningsytan bestäms av följande differens:

Dimensioner för normverkyget - verkygstradie (WZ-paremeter \$TC_DP6)

Advanced Surface / Top Surface

Märk

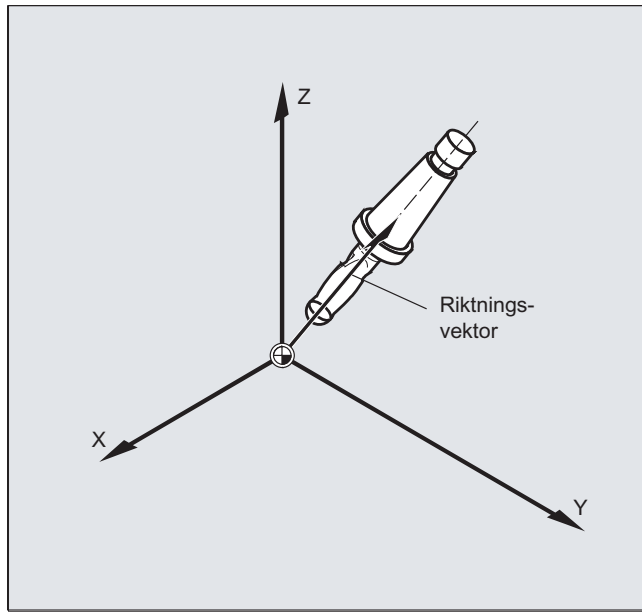
Vid användning av verkygstradiekompenseringen CUT3DCC / CUT3DCCD i kombination med den licenspliktiga funktionen "Advanced Surface" eller "Top Surface" ska inställningsrekommendationerna gällande "Advanced Surface" / "Top Surface" beaktas!

För kontroll av inställda data står speciella kontrollprogram till förfogande via SIOS-portalen:

- Kontrollprogram för Advanced Surface (<https://support.industry.siemens.com/cs/ww/en/view/78956392>)
- Kontrollprogram för Top Surface (<https://support.industry.siemens.com/cs/ww/en/view/109738423>)

11.6 Verkygsorientering (ORIC, ORID, OSOF, OSC, OSS, OSSE, ORIS, OSD, OST)

Med verkygsorientering menar man den geometriska upprikningen av verktyget i rymden. Vid en 5-axels bearbetningsmaskin kan verkygsorienteringen ställas in vid programkommandon.



Med **OSD** och **OST** aktiverade översläpningsrörelser för orienteringen bildas olika beroende på interpoleringstypen för verkygsorienteringen.

Vid aktiv vektorinterpolering interpoleras det utjämnade orienteringsförloppet också med hjälp av vektorinterpolering. Däremot jämnas vid aktiv roterande axelinterpolering orienteringen direkt med hjälp av roterande axelrörelser.

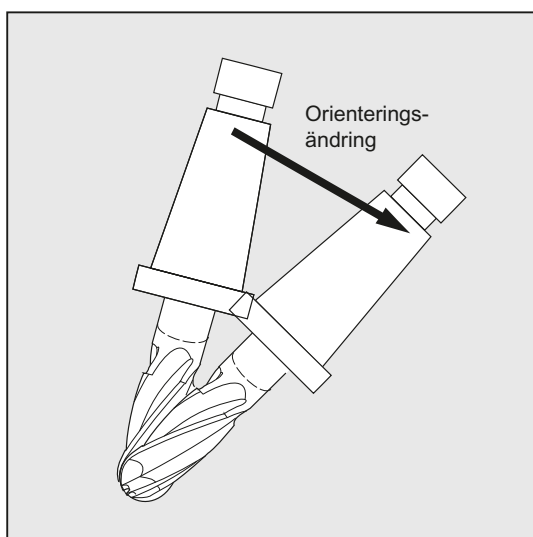
Programmering

Programmering av orienteringsändringen:

En orienteringsändring för verktyget kan programmeras genom:

- direkt programmering av de roterande axlarna A, B, C (roterande axelinterpolering)
- Euler- eller RPY-vinkel
- riktningsvektor (vektorinterpolering genom angivande av A3 eller B3 eller C3)
- LEAD/TILT (planfräsning)

Referenskoordinatsystemet är antingen maskinkoordinatsystemet (ORIMKS) eller det aktuella arbetsstyckskoordinatsystemet (ORIWKS).



Programmering av verktygsorienteringen:

ORIC:	Orientering och banrörelse parallella
ORID:	Orientering och banrörelse efter varandra
OSOF:	Ingen orienteringsjämning
OSC:	Orientering konstant
OSS:	Orienteringsjämning endast i blockbörjan
OSSE:	Orienteringsjämning i blockbörjan och -slut
ORIS:	Hastighet för orienteringsändringen vid tillkopplad orienteringsjämning i grader per mm (gäller för OSS och OSSE)
OSD:	Översläpning av orienteringen genom angivande av översläpningslängd med settingdatum: SD42674 \$SC_ORI_SMOOTH_DIST
OST:	Översläpning av orienteringen genom angivande av vinkeltoleransen i grader vid vektorinterpolering med settingdatumet: SD42676 \$SC_ORI_SMOOTH_TOL Vid roterande axelinterpolering antas den föreskrivna toleransen som maximal avvikel- se för orienteringsaxlarna.

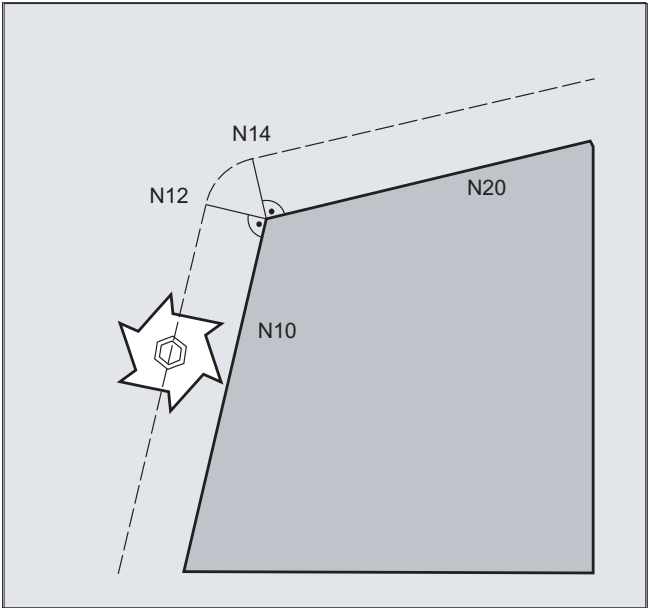
Märk

Alla kommandon för översläpning av verktygsorienteringen (OSOF, OSC, OSS, OSSE, OSD och OST) är sammanfattade i G-gruppen 34. De är modalt verksamma dvs. endast ett av dessa kommandon kan verka.

Exempel

Exempel 1: ORIC

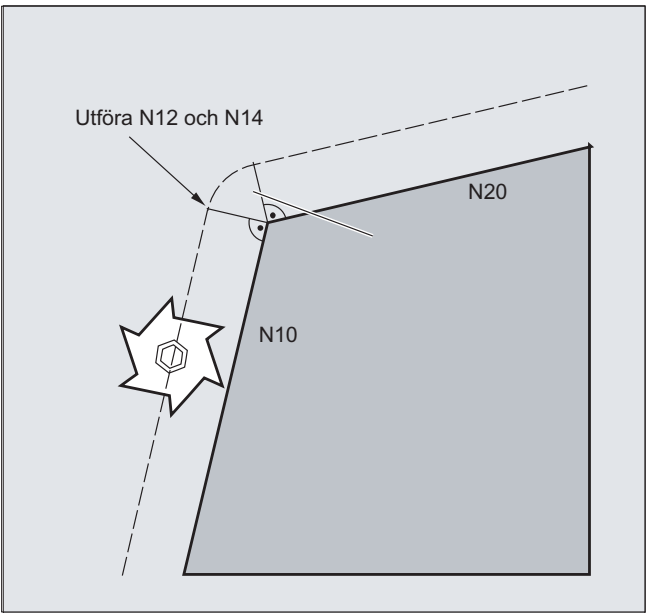
Är mellan förflyttningsblocken N10 och N20 två eller flera block med orienteringsändringar (t.ex. A2=... B2=... C2=...) programmerade och ORIC är aktiv, så delas det infogade cirkelblocket upp på dessa intermediära block i enlighet med beloppet för vinkeländringarna.



Programkod	Kommentar
ORIC	
N8 A2=... B2=... C2=...	
N10 X... Y... Z...	
N12 C2=... B2=...	; Cirkelblocket som infogas vid ytterhörnet fördelar sig på N12 och N14 i enlighet med orienteringsändringen. Cirkelrörelse och orienteringsändring utförs härvid parallellt.
N14 C2=... B2=...	
N20 X =...Y=... Z=... G1 F200	

Exempel 2: ORID

Är ORID aktiv, så utförs alla block mellan de båda förflyttningsblocken i slutet av det första förflyttningsblocket. Cirkelblocket med konstant orientering utförs omedelbart före det andra förflyttningsblocket.

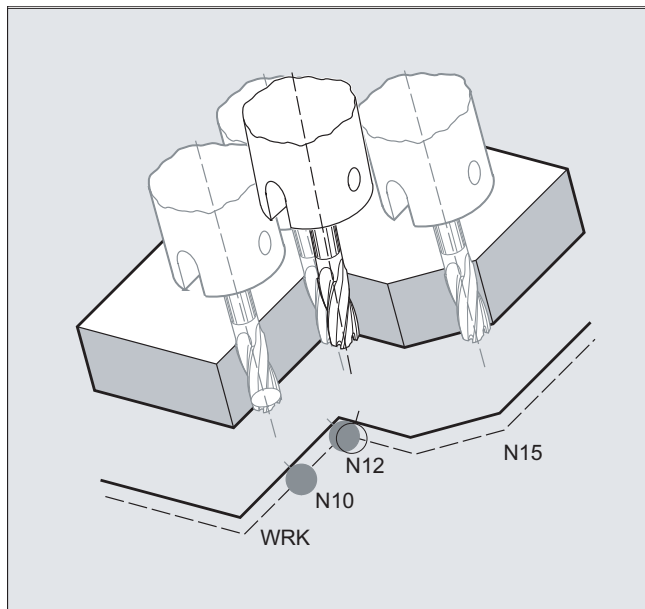


Programkod	Kommentar
ORID	
N8 A2=... B2=... C2=...	
N10 X... Y... Z...	
N12 A2=... B2=... C2=...	; Blocket N12 och N14 utförs i slutet av N10. Därefter körs cirkelblocket ut med den aktuella orienteringen.
N14 M20	; Hjälpfunktioner etc.
N20 X... Y... Z...	

Märk

För typen av orienteringsändring vid ett ytterhörn är det programkommando utslagsgivande som är aktivt i det första förflyttningsblocket för ett ytterhörn.

Utan orienteringsändring: Förändras orienteringen vid blockgränsen inte, så är verktygsarean en cirkel som berör de båda konturerna.

Exempel 3: Ändring av orienteringen vid ett innerhörn**Programkod**

```

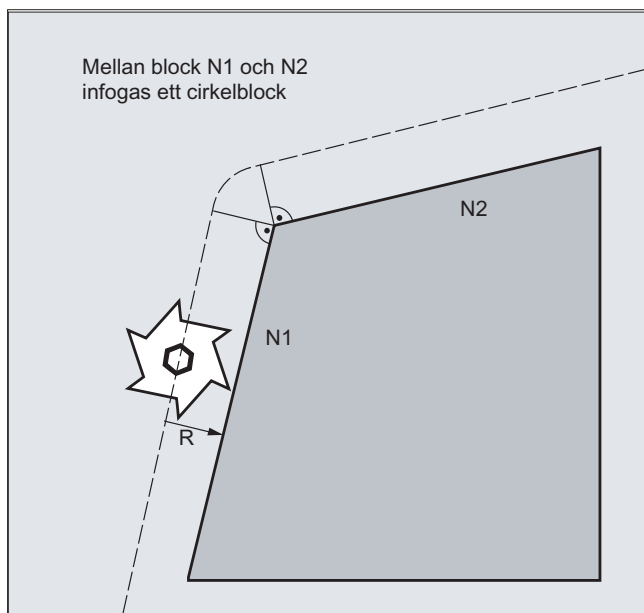
ORIC
N10 X ...Y... Z... G1 F500
N12 X ...Y... Z... A2=... B2=... C2=...
N15 X ...Y... Z... A2=... B2=... C2=...

```

Ytterligare informationer**Beteende vid ytterhörn**

Vid ett ytterhörn infogas alltid ett cirkelblock med radien på fräsen.

Med programkommandona ORIC resp. ORID kan fastläggas om orienteringsändringarna, som programmerades mellan block N1 och N2, utförs före början av det infogade cirkelblocket eller samtidigt med detta.



Är en orienteringsändring nödvändig vid ytterhörn, så kan denna valfritt göras parallellt till interpoleringen eller åtskilt med banrörelsen.

Vid **ORID** utförs först de infogade blocken utan banrörelse. Cirkelblocket infogas omedelbart före det andra av de båda förflyttningsblocken genom vilka hörnet bildas.

Är flera orienteringsblock infogade vid ett ytterhörn och har **ORIC** valts, så fördelas cirkelrörelsen i enlighet med beloppen för orienteringsändringarna för de enskilda infogade blocken på dessa.

Översläpning av orienteringen med **OSD** resp. **OST**

Vid översläpning med **G642** kan den maximala avvikelser för konturaxlarna och orienteringsaxlarna inte vara mycket olika. Den mindre toleransen av båda bestämmer formen för översläpningsrörelsen resp. vinkeltoleransen, att jämna orienteringsförloppet relativt kraftigt, utan att därvid behöva acceptera större konturavvikelser.

Genom aktivering av **OSD** resp. **OST** är det möjligt, att med en föreskriven översläpningslängd resp. vinkeltolerans "frikostigt" jämna mycket små avvikelser i orienteringsförloppet utan graverande konturavvikelser.

Märk

Till skillnad mot översläpning av konturen (och orienteringsförloppet) med **G642** bildas vid översläpningen av orienteringen med **OSD** resp. **OST** inget eget block utan översläpningsrörelsen infogas direkt i de programmerade originalblocken.

Med **OSD** resp. **OST** kan inga blockövergångar översläpas vid vilka ett byte av interpoleringstyp för verkygsorienteringen (vektor → roterande axel, roterande axel → vektor) äger rum. Dessa blockövergångar kan eventuellt översläpas med de vanliga översläpningsfunktionerna **G641**, **G642** resp. **G643**.

11.7 Fri D-nummerfördelning, skärnummer

11.7.1 Fri D-nummerfördelning, skärnummer (adress CE)

D-nummer

D-numren kan användas som kompenseringsnummer. Dessutom kan via adressen CE numret för skäret adresseras. Via systemvariabeln \$TC_DPCE kan skärnumret beskrivas.

Förinställning: Kompenseringsnr == skärnr

Via maskindata fastläggs det maximala antalet D-nummer (skärnummer) och det maximala skärantalet per verktyg (→ maskintillverkare). Die följande kommandona är motiverade endast när det maximala skärnumret (MD18105) bestämdes större än antalet skär per verktyg (MD18106). Följ anvisningarna därtill från maskintillverkaren.

Märk

Förutom den relativa D-nummerfördelningen kan D-numren också fördelas som "grunda" resp. "absoluta" D-nummer (1-32000) utan referens till ett T-nummer (inom funktionen "Grund D-nummerstruktur").

Litteratur

Funktionshandbok Grundfunktioner; Verkygskompensering (W1)

11.7.2 Fri D-nummerfördelning: Kontrollera D-nummer (CHKDNO)

Med kommandot `CKKDNO` kontrollerar du om de förefintliga D-numren har fördelats entydigt. D-numren för alla inom en TO-enhet definierade verktyg får bara uppträda en gång. Därvid tas inte någon hänsyn till utbytesverktyg.

Syntax

```
state=CHKDNO (Tno1, Tno2, Dno)
```

Betydelse

state:	=TRUE:	D-numren fördelades entydigt för det kontrollerade området.
	=FALSE:	En D-nummerkollision ägde rum eller parametreringen är ogiltig. Via Tno1, Tno2 och Dno överförs parametrarna som ledde till kollision. Dessa data kan utvärderas i detaljprogrammet.
CHKDNO (Tno1, Tno2) :		Alla D-numren för de nämnda verktygen kontrolleras.

CHKDNO (Tno1):	Alla D-nummer från Tno1 kontrolleras mot alla andra verktyg.
CHKDNO:	Alla D-nummer från alla verktyg kontrolleras mot alla andra verktyg.

11.7.3 Fri D-nummerfördelning: Byta namn på D-numren (GETDNO, SETDNO)

D-numren måste fördelas entydigt. Två olika skär till ett verktyg kan inte ha samma D-nummer.

GETDNO

Detta kommando levererar D-numret för ett bestämt skär (ce) till ett verktyg med T-numret t. Existerar inget D-nummer till de inmatade parametrarna sätts d=0. Är D-numret ogiltigt returneras ett värde större än 32000.

SETDNO

Med detta kommando tillordnar du värdet d till D-numret för ett skär ce till verktyget t. Via state returneras resultatet av denna anvisning (TRUE eller FALSE). Existerar inget datablock till de inmatade parametrarna returneras FALSE. Syntaxfel skapar ett larm. D-numren kan inte sättas explicit på 0.

Syntax

```
d = GETDNO (t,ce)
state = SETDNO (t,ce,d)
```

Betydelse

d:	D-nummer på skäret till verktyget
t:	T-nummer på verktyget
ce:	Skärnummer (CE-nummer) på verktyget
state:	Anger om kommandot kunde utföras felfritt (TRUE eller FALSE).

Exempel namnbyte på ett D-nummer

Programmering	Kommentar
\$TC_DP2[1,2] = 120	
\$TC_DP3[1,2] = 5.5	
\$TC_DPCE[1,2] = 3	; Skärnummer CE
...	
N10 def int DNrAlt, DNrNeu = 17	
N20 DNrAlt = GETDNO(1,3)	
N30 SETDNO(1,3,DNrNeu)	

Därmed tillordnas skäret CE=3 det nya D-värdet 17. Nu anropas data för detta skär via D-numret 17; både via systemvariablerna och även i programmeringen med NC-adressen.

11.7.4 Fri D-nummerfördelning: Fastställa T-nummer till föreskrivet D-nummer (GETACTTD)

Med kommandot `GETACTTD` fastställer du till ett absolut D-nummer det tillhörande T-numret. Det sker ingen kontroll med avseende på entydighet. Finns det flera lika D-nummer inom en TO-enhet returneras T-numret för det först hittade verktyget. Vid användning av "grunda" D-nummer är användningen av kommandot inte motiverad, eftersom här alltid värdet "1" returneras (inget T-nummer i dataförvaltningen).

Syntax

```
status=GETACTTD (Tnr, Dnr)
```

Betydelse

Dnr:	D-Nummer, för die die T-Nummer gesucht werden soll.	
Tnr:	Hittade T-nummer	
status:	Värde:	Betydelse:
	0	T-numret hittades. Tnr erhåller värdet för T-numret.
	-1	Till det angivna D-numret existerar inget T-nummer; Tnr=0.
	-2	D-numret är inte absolut. Tnr erhåller värdet för det först hittade verktyget som innehåller D-numret med värdet Dnr.
	-5	Funktionen kunde av ett annat skäl inte utföras.

11.7.5 Fri D-nummerfördelning: Sätta D-nummer ogiltiga (DZERO)

Kommandot `DZERO` tjänar till stöd under omriggningen. Så markerade kompenseringssatablock kontrolleras inte längre av kommandot `CHKDNO`. För att åter göras tillgängligt, måste D-numret åter sättas med `SETDNO`.

Syntax

```
DZERO
```

Betydelse

DZERO:	Kännetecknar alla D-nummer i TO-enheten som ogiltiga.
--------	---

11.8 Verkygsbärarkinematik

Förutsättningar

En verkygsbärare kan orientera ett verktyg i alla möjliga rymdriktningar endast när

- två vridaxlar v_1 och v_2 finns.
- vridaxlarna står vinkelrätt mot varandra.
- verkygslängdaxeln står vinkelrätt mot den andra vridaxeln v_2 .

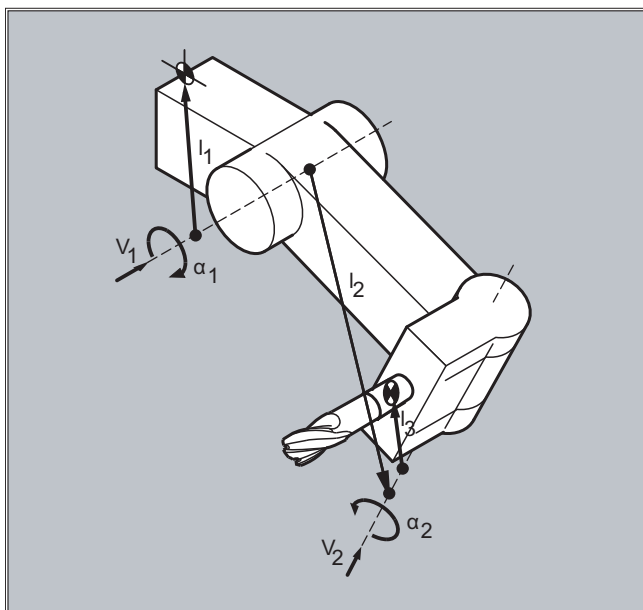
Dessutom gäller för maskiner på vilka alla möjliga orienteringar måste kunna ställas in följande krav:

- verkygsorienteringen måste stå vinkelrätt mot den första vridaxeln v_1 .

Funktion

Verkygsbärarkinematiken med maximalt två vridaxlar v_1 eller v_2 beskrivs med de 17 systemvariablerna $\$TC_CARR1[m]$ till $\$TC_CARR17[m]$. Beskrivningen av verkygsbäraren består av:

- det vektoriella avståndet från den första vridaxeln till referenspunkten för verkygsbäraren I_1 , det vektoriella avståndet från den första till den andra vridaxeln I_2 , det vektoriella avståndet från den andra vridaxeln till referenspunkten för verkyget I_3 .
- riktningsvektorerna för de båda vridaxlarna v_1 , v_2 .
- vridvinklarna α_1 , α_2 runt de båda axlarna. Vridvinkeln räknas positiv medurs när man blickar i vridaxelvektorernas riktning.



För maskiner med **upplöst kinematik** (både verktyg och arbetsstycke kan vridas) kompletterades systemvariablerna med posterna $\$TC_CARR18[m]$ till $\$TC_CARR23[m]$.

Parameter

Funktion hos systemvariablerna för orientierbara verktygsbärare			
Beteckning	x-komponent	y-komponent	z-komponent
l_1 Offsetvector	\$TC_CARR1[m]	\$TC_CARR2[m]	\$TC_CARR3[m]
l_2 Offsetvector	\$TC_CARR4[m]	\$TC_CARR5[m]	\$TC_CARR6[m]
v_1 Vridaxel	\$TC_CARR7[m]	\$TC_CARR8[m]	\$TC_CARR9[m]
v_2 Vridaxel	\$TC_CARR10[m]	\$TC_CARR11[m]	\$TC_CARR12[m]
α_1 Vridvinkel	\$TC_CARR13[m]		
α_2 Vridvinkel	\$TC_CARR14[m]		
l_3 Offsetvector	\$TC_CARR15[m]	\$TC_CARR16[m]	\$TC_CARR17[m]

Kompletteringar till systemvariablerna för orientierbara verktygsbärare			
Beteckning	x-komponent	y-komponent	z-komponent
l_4 Offsetvector	\$TC_CARR18[m]	\$TC_CARR19[m]	\$TC_CARR20[m]
Axelbeteckning Vridaxel v_1 Vridaxel v_2	Axelbeteckningar för vridaxlarna v_1 och v_2 (förbeläggning är noll) \$TC_CARR21[m] \$TC_CARR22[m]		
Kinematiktyp	\$TC_CARR23[m]		
Tool	Kinematiktyp-T ->	Kinematiktyp-P ->	Kinematiktyp-M
Part	Endast verktyget är vridbart (förbeläggning)	Endast arbetsstycket är vridbart	Arbetsstycke & verktyg är vridbara
Mixed mode			
Offset för Vridaxel v_1 Vridaxel v_2	Vinkel i grader för vridaxlarna v_1 och v_2 vid inatagandet av grundläget \$TC_CARR24[m] \$TC_CARR25[m]		
Vinkeloffset för Vridaxel v_1 Vridaxel v_2	Offset för Hirth-tandningen i grader för vridaxlarna v_1 och v_2 \$TC_CARR26[m] \$TC_CARR27[m]		
Vinkelinkrem. v_1 vridaxel v_2 vridaxel	Inkrement och Hirth-tandningen i grader för vridaxlarna v_1 och v_2 \$TC_CARR28[m] \$TC_CARR29[m]		
Min.-position Vridaxel v_1 Vridaxel v_2	Software-limit för minimiposition för vridaxlarna v_1 och v_2 \$TC_CARR30[m] \$TC_CARR31[m]		
Max.-position Vridaxel v_1 Vridaxel v_2	Software-limits för maximiposition för vridaxlarna v_1 och v_2 \$TC_CARR32[m] \$TC_CARR33[m]		
Verktygsbärare namn	I stället för ett tal kan en verktygsbärare få ett namn. \$TC_CARR34[m]		
Användare: Axelnamn 1 Axelnamn 2 Beteckning	Avsedd användning inom mätcyklerna av användaren. \$TC_CARR35[m] \$TC_CARR36[m] \$TC_CARR37[m] \$TC_CARR38[m]		
Position		\$TC_CARR39[m]	\$TC_CARR40[m]
Finför-flyttning	Parametrar som kan bli adderade till värdena i basparametrarna.		
l_1 Offsetvector	\$TC_CARR41[m]	\$TC_CARR42[m]	\$TC_CARR43[m]
l_2 Offsetvector	\$TC_CARR44[m]	\$TC_CARR45[m]	\$TC_CARR46[m]

Kompletteringar till systemvariablerna för orientierbara verktygsbärare			
I_3 Offsetvector	\$TC_CARR55[m]	\$TC_CARR56[m]	\$TC_CARR57[m]
I_4 Offsetvector	\$TC_CARR58[m]	\$TC_CARR59[m]	\$TC_CARR60[m]
v_1 vridaxel	\$TC_CARR64[m]		
v_2 vridaxel	\$TC_CARR65[m]		

Märk**Förklaringar till parametrarna**

Med "m" anges alltså numret för den verktygsbärare som ska beskrivas.

\$TC_CARR47 till \$TC_CARR54 samt \$TC_CARR61 till \$TC_CARR63 är inte definierade och leder till ett larm vid försök till åtkomst för att läsa eller skriva.

Begynnelse- resp. slutpunkterna för avståndsvektorerna på axlarna kan väljas fritt. Vridvinklarna α_1 , α_2 runt de båda axlarna definieras i verktygsbärarens grundtillstånd med 0° . Kinematiken för en verktygsbärare kan så beskrivas med valfritt många olika möjligheter.

Verktygsbärare med endast en eller ingen vridaxel kan beskrivas genom nollställning av riktningvektorerna för en eller båda vridaxlarna.

För en verktygsbärare utan vridaxel verkar avståndsvektorerna som ytterligare verktygskompenseringar, vars komponenter inte påverkas vid omkoppling av bearbetningsplanen (G17 till G19) $^\circ$.

Kompletteringar till parametrarna**Parametrar till vridaxlarna**

Systemvariablerna kompletterades med posterna \$TC_CARR24[m] till \$TC_CARR33[m] och beskrivs på följande sätt:

Offset för vridaxlarna v_1 , v_2	Förändring av positionen för vridaxeln v_1 eller v_2 vid grundläge för den orientierbara verktygsbäraren.
Vinkeloffset/ vinkelinkrement vridaxlarna v_1 , v_2	Offset eller vinkelinkrement för Hirth-tandningen för vridaxlarna v_1 och v_2 . Programmerad eller beräknad vinkel avrundas till det närmast liggande värdet, som vid heltaligt n resulterar ur $\phi = s + n \cdot d$.
Minimal- och maximalposition vridaxlarna v_1 , v_2	Minimalposition/maximalposition för vridaxeln gränsvinkel (software-limit) för vridaxel v_1 och v_2 .

Parametrar för användaren

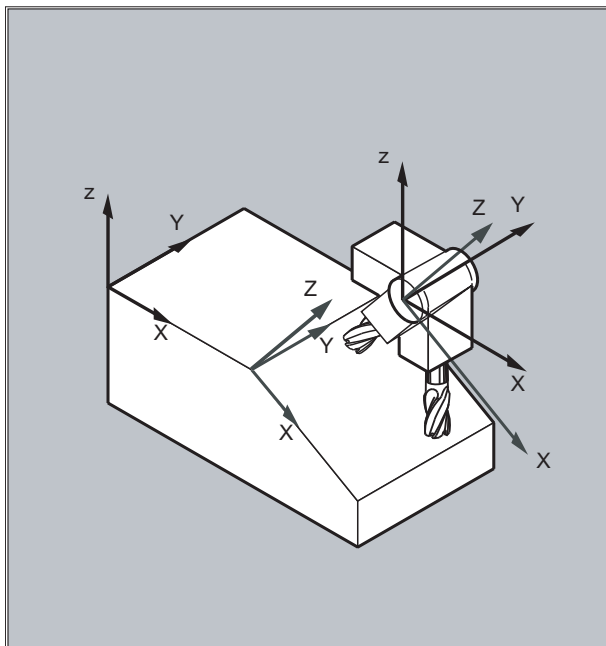
\$TC_CARR34 till \$TC_CARR40 innehåller parametrar som står användaren till fritt förfogande och fram till SW 6.4 standardmässigt inom NCK inte utvärderas vidare eller inte har någon betydelse.

Parametrar för finförflyttningen

\$TC_CARR41 till \$TC_CARR65 innehåller finförflyttningsparametrar, som kan adderas till värdena i basparametrarna. Det finförflyttningssvärde som tillordnats till en basparameter bildas när värdet 40 adderas till parameternumret.

Exempel

Den i följande exempel använda verkygsbäraren låter sig fullständigt beskrivas genom en vridning runt Y-axeln.



Programkod	Kommentar
N10 \$TC_CARR8[1]=1	; Definition av Y-komponenten till den första vridaxeln till verkygsbärare 1.
N20 \$TC_DP1[1,1]=120	; Definition för en pinnfräs.
N30 \$TC_DP3[1,1]=20	; Definition för en pinnfräs med längden 20 mm.
N40 \$TC_DP6[1,1]=5	; Definition för en pinnfräs med radien 5 mm.
N50 ROT Y37	; Framedefinition med vridning 37° runt Y-axeln.
N60 X0 Y0 Z0 F10000	; Köra till utgångsposition.
N70 G42 CUT2DF TCOFR TCARR=1 T1 D1 X10	; Ställa in radiekompensering, verkygslängdkompensering i vriden frame, verkygsbärare 1, välja verkyg 1.
N80 X40	; Utföra bearbetningen under en vridning av 37°.
N90 Y40	
N100 X0	
N110 Y0	
N120 M30	

Ytterligare informationer

Upplöst kinematik

För maskiner med upplöst kinematik (både verktyg och arbetsstycke är vridbara) kompletterades systemvariablerna med posterna `$TC_CARR18 [m]` till `$TC_CARR23 [m]` och beskrivs på följande sätt:

Det vridbara verktygsbordet bestående av:

- det vektoriella avståndet för den andra vridaxeln V_2 till referenspunkten för ett vridbart verktygsbord I_4 till den tredje vridaxeln.

De roterande axlarna bestående av:

- de båda kanalbeteckningarna för referensen till vridaxlarna V_1 och V_2 , till vars position eventuellt åtkomst förskaffas vid bestämningen av orienteringen för den orienterbara verktygsbäraren.

Kinematiktypen med ett av värdena T, P eller M:

- Kinematiktyp T: Endast verktyget är vridbart.
- Kinematiktyp P: Endast arbetsstycket är vridbart.
- Kinematiktyp M: Verktyg & arbetsstycke är vridbara.

Radering av verktygsbärardata

Med `$TC_CARR1 [0]=0` kan data för alla verktygsbärardatablock raderas.

Kinematiktypen `$TC_CARR23 [T]=T` måste beläggas med en av de tre tillåtna stora eller små bokstäverna (T,P,M) och bör av detta skäl inte raderas.

Ändring av verktygsbärardata

Alla de beskrivna värdena kan förändras genom tillordning av ett nytt värde i detaljprogrammet. Varje annat tecken än T, P eller M leder till ett larm vid försök att aktivera den orienterbara verktygsbäraren.

Läsning av verktygsbärardata

Alla de beskrivna värdena kan läsas genom tillordning av en variabel i detaljprogrammet.

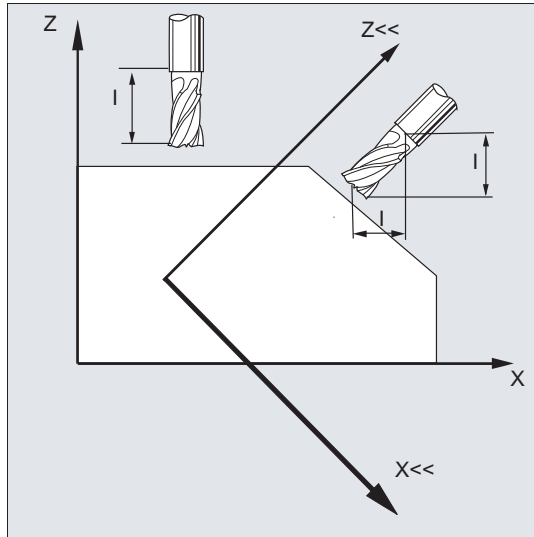
Finförflyttningar

Ett otillåtet finförflyttningsvärde identifieras först vid aktivering av en orienterbar verktygsbärare, som innehåller ett sådant värde och samtidigt är settingdatumet SD42974 `$SC_TOCARR_FINE_CORRECTION = TRUE`.

Beloppet för den tillåtna finförflyttningen begränsas till ett maximalt tillåtet värde via maskindata.

11.9 Verkygslängdkompenseringar för orienterbara verktygsbärare (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)

Med förändrad rymdorientering av verktyget ändrar sig också dess verktygslängdkomponenter.



Efter omriggning t.ex. genom manuell inställning eller byte av verktygsbärare med i rymden fast uppriktning, måste därför verktygslängdkomponenterna fastställas på nytt. Detta görs med vägkommandona TCOABS och TCOFR.

Vid en orienterbar verktygsbärare till en aktiv frame kan vid verktygsval med TCOFRZ, TCOFRY och TCOFRX den riktning, i vilken verktyget ska peka, bestämmas.

Syntax

TCARR= [<m>]
 TCOABS
 TCOFR
 TCOFRZ
 TCOFRY
 TCOFRX

Betydelse

TCARR= [<m>] :	Begära verktygsbärare med numret "m"
TCOABS:	Beräkna verktygslängdkomponenterna ur den aktuella verktygsbärarorienteringen
TCOFR:	Bestämma verktygslängdkomponenterna ur orienteringen för den aktiva ramen
TCOFRZ:	Orienterbar verktygsbärare från aktiv frame vars verktyg pekar i Z-riktningen
TCOFRY:	Orienterbar verktygsbärare från aktiv frame vars verktyg pekar i Y-riktningen
TCOFRX:	Orienterbar verktygsbärare från aktiv frame vars verktyg pekar i X-riktningen

Ytterligare informationer

Verkygslängdkompensering ur bärarorientering (TCOABS)

TCOABS beräknar verkygslängdkompenseringen ur den aktuella orienteringsvinkeln för verkygsbäraren; förvarad i systemvariablerna \$TC_CARR13 och \$TC_CARR14.

För definition av verkygsbärarkinematiken med systemvariabler se " Verkygsbärarkinematik (Sida 442) ".

För nyberäkning av verkygslängdkompenseringen vid frame-byte måste verkyget väljas en gång till.

Verkygsriktning ur aktiv frame

Den orienterbara verkygsbäraren kan ställas in så att verkyget pekar i följande riktningar:

- med TCOFR resp. TCOFRZ i Z-riktningen
- med TCOFRY i Y-riktningen
- med TCOFRX i X-riktningen

En omkoppling mellan TCOFR och TCOABS har en nyberäkning av verkygslängdkompenseringen till följd.

Begära verkygsbärare (TCARR)

Med TCARR begärs med verkygsbärarnumret m dess geometridata (kompenseringsminne).

Med m=0 väljs den aktiva verkygsbäraren bort.

Geometridata för verkygsbäraren blir aktiva först efter anrop av ett verkyg. Det valda verkyget förblir aktivt utöver bytet av en verkygsbärare.

De aktuella geometridata för verkygsbäraren kan också i detaljprogrammet definieras via de motsvarande systemvariablerna.

Nyberäkning av verkygslängdkompenseringen (TCOABS) vid frame-byte

För nyberäkning av verkygslängdkompenseringen vid frame-byte måste verkyget väljas en gång till.

Märk

Verkygsorienteringen måste manuellt anpassas till den aktiva ramen.

Vid beräkningen av verkygslängdkompenseringen beräknas i ett intermediärt steg också vridvinklarna för verkygsbäraren. Eftersom vid verkygsbärare med två vridaxlar i allmänhet två vridvinkelpar existerar, med vilka verkygsorienteringen kan anpassas till den aktiva

11.9 Verkygslängdkompenseringar för orienterbara verkygsbärare (TCARR, TCOABS, TCOFR, TCOFRX, TCOFRY, TCOFRZ)

framen, måste de i systemvariablerna lagrade vridvinkelvärdena åtminstone någorlunda motsvara de mekaniskt inställda vridvinklarna.

Märk

Verkygsorientering

Styrningen kan inte kontrollera om de via frame-orienteringen beräknade vridvinklarna kan ställas in på maskinen.

Är vridaxlarna till verkygsbäraren konstruktivt så placerade att den genom frame-orienteringen beräknade verkygsorienteringen inte kan uppnås matas ett larm ut.

Kombinationen av verkygsfinkompensering och funktionaliteterna till verkygslängdkompensering vid rörliga verkygsbärare är inte tillåten. Vid försök att samtidigt anropa de båda funktionerna följer ett felmeddelande.

Med `TOFRAME` är det möjligt att definiera en frame på grund av orienteringsriktningen för den valda verkygsbäraren. Närmare informationer se Kapitel "Frames".

Vid aktiv orienteringstransformation (3-, 4-, 5-axelstransformation) kan en verkygsbärare med en från noll-läget avvikande orientering väljas utan att därvid ett larm matas ut.

Överföringsparametrar från standard- och mätcykler

För överföringsparametrarna från standard- och mätcykler gäller definierade värdeområden.

Vid vinkelvärden är värdeområdet fastlagt på följande sätt:

- Vridning runt 1:a geometriaxeln: -180 grader till +180 grader
- Vridning runt 2:a geometriaxeln: -90 grader till +90 grader
- Vridning runt 3:e geometriaxeln: -180 grader till +180 grader

Se Kapitel Frames, "Programmerbar vridning (ROT, AROT, RPL)".

Märk

Vid överförande av vinkelvärden till en standard- eller mätcykel ska beaktas att:

Värden mindre än räkn noggrannheten för NC ska rundas till noll!

Räkn noggrannheten i NC för vinkelpositionerna är fastlagd i maskindatum:

MD10210 \$MN_INT_INCR_PER_DEG

11.10 Online-verktygslängdkompensering (TOFFON, TOFFOF)

Via systemvariabeln \$AA_TOFF[<n>] kan den effektiva verktygslängden överlagras motsvarande de tre verktygsriktningarna tredimensionellt i realtid.

Som index <n> används de tre geometriaxelbeteckningarna. Därmed är antalet aktiva kompenseringsriktningar fastlagt genom de vid samma tid aktiva geometriaxlarna.

Alla kompenseringar kan vara aktiva samtidigt.

Funktionen online-verktygslängdkompensering kan användas vid:

- Orienteringstransformation TRAORI
- Orienterbar verktygsbärare TCARR

Märk

Online-verktygslängdkompenseringen är en **option**, som måste frikopplas dessförinnan. Endast i förbindelse med en aktiv orienteringstransformation eller en aktiv orienterbar verktygsbärare är denna funktion motiverad.

Syntax

```
TRAORI
TOFFON(<Korrekturrichtung>[,<Offsetwert>])
WHEN TRUE DO $AA_TOFF[<Korrekturrichtung>]           ; I synkronaktioner.
...
TOFFOF(<Korrekturrichtung>)
```

Ytterligare förklaringar till programmering av online-verktygslängdkompenseringen i rörelsesynkronaktioner se "Synkronaktioner (Sida 597)".

Betydelse

TOFFON:	Aktivera online-verktygslängdkompensering	
	<Korrekturrichtung>:	Verktygsriktning (X, Y, Z), i vilken online-verktygslängdkompenseringen ska vara verksam.
	<Offsetwert>:	Vid aktiveringen kan för den motsvarande kompenseringsriktningen ett offsetvärde angivas som genast kan köras ut.
TOFFOF:	Återställa online-verktygslängdkompensering Kompenseringsvärdena i den angivna kompenseringsriktningen återställs och ett fördekoderingsstopp utlöses.	

Exempel

Exempel 1: Val av verktyglängdkompensering

Programkod	Kommentar
MD21190 \$MC_TOFF_MODE =1	; Absoluta värden uppsöks.
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)	; Transformation till.
N20 TOFFON(Z)	; Aktivering av online-WZL-kompensering för Z-verktygsriktning.
N30 WHEN TRUE DO \$AA_TOFF[Z]=10 G4 F5	; För Z-verktygsriktningen interpoleras en WZL-kompensering på 10.
...	
N100 XOFFSET=\$AA_TOFF_VAL[X]	; Tillordna aktuell kompensering i X-riktningen.
N120 TOFFON(X,-XOFFSET) G4 F5	; För X-verktygsriktningen körs WZL-kompenseringen åter tillbaka på 0.

Exempel 2: Borval av verktyglängdkompensering

Programkod	Kommentar
N10 TRAORI(1)	; Transformation till.
N20 TOFFON(X)	; Aktivering av online-WZL-kompensering för X-verktygsriktning.
N30 WHEN TRUE DO \$AA_TOFF[X]=10 G4 F5	; För X-verktygsriktningen interpoleras en WZL-kompensering på 10.
...	
N80 TOFFOF(X)	; Positionsoffset för X-verktygsriktningen raderas: ...\$AA_TOFF[X]=0 Ingen axel förflyttas. Till den aktuella positionen i WKS läggs positionsoffseten till enligt den aktuella orienteringen.

Ytterligare informationer

Blockförberedelse

Vid blockförberedelse i förkörningen tas det också hänsyn till den i huvudkörningen verk samma aktuella verktyglängdoffseten. För att så långt som möjligt kunna utnyttja de

maximalt tillåtna axelhastigheterna är det erforderligt att stoppa blockförberedelsen med ett fördekoderingsstopp `STOPRE` under det en verkygsoffset byggs upp.

Verkygsoffseten är vid tidpunkten för förkörningen också alltid känd när verkygslängdkompenseringarna inte mer förändras efter programstart eller när efter en förändring av verkygslängdkompenseringarn fler block genomarbetades än IPO-bufferten mellan förkörning och huvudkörning kan assimilera.

Variabel \$AA_TOFF_PREP_DIFF

Måttet för differensen mellan den aktuella i interpolatorn verksamma kompenseringen och den kompensering, som vid tidpunkten för blockförberedelsen var verksam, kan avfrågas i variabeln `$AA_TOFF_PREP_DIFF[<n>]`.

Ställa in maskindata och settingdata

För online-verkygslängdkompenseringen står följande systemdata till förfogande:

- MD20610 `$MC_ADD_MOVE_ACCEL_RESERVE` (accelerationsreserv för överlagrad rörelse)
- MD21190 `$MC_TOFF_MODE`
Innehållet i systemvariabeln `$AA_TOFF[<n>]` körs ut som absolut värde eller integreras.
- MD21194 `$MC_TOFF_VELO` (hastighet för online-verkygslängdkompenseringen)
- MD21196 `$MC_TOFF_ACCEL` (acceleration för online-verkygslängdkompenseringen)
- Settingdatum för angivande av gränsvärden:
SD42970 `$SC_TOFF_LIMIT` (övre gräns för verkygslängdkompenseringsvärdet)

Litteratur:

Funktionshandbok Specialfunktioner; F2: Fleraxeltransformationer

11.11 Modifikation av kompenseringsdata för vridbara verktyg

11.11.1 Beräkna orienteringar (ORISOLH)

Den fördefinierade funktionen ORISOLH stöder användaren när det gäller att ställa in positionerna för de roterande axlarna till en maskin, så att ett svarvverktyg förs till ett definierat, kinematikoberoende läge relativt till arbetsstycket. Förutsättning är att en 6-axeltransformation, vilken är parameterad med kinematiska följder, är aktiv

Två grundfunktioner står till förfogande:

- Upprikta verktyg
De båda vinklarna β och γ föreskrivs. Funktionen beräknar de för de tre orienteringsaxlarna nödvändiga vinklarna för detta.
- Rikta upp verktyg direkt
Vinklarna mellan den andra och tredje orienteringsaxeln föreskrivs. Funktionen beräknar därur de tillhörande vinklarna β och γ , samt vinkeln för den första orienteringsaxeln som fortfarande saknas.

Märk

Ordningsföljd för orienteringsaxlarna

Genomlöper man den kinematiska följden, som beskriver maskinens uppbyggnad, från arbetsstycket till verktyget, då gäller för ordningsföljden av de tre orienteringsaxlarna i en 6-axeltransformation följande bestämmelser:

- Den orienteringsaxel, som ligger närmast till **arbetsstycket**, är den **första** orienteringsaxeln.
- Den orienteringsaxel, som ligger närmast till **verktyget**, är den **tredje** orienteringsaxeln.

Den första orienteringsaxeln kommer som regel att vara en spindel och den motsvarande rotationen realiserar därför i dessa fall genom en vridande frame.

Syntax

```
<RetVal> = ORISOLH(<Cntrl>,<W1>,<W2>)
```

Betydelse

ORISOLH:	Funktionsanrop	
<RetVal>:	Funktionens returvärde	
	Datatyp:	INT
	Värdeområde:	0, -2, -3, ..., -17
	Värden:	0 Funktionen avslutades utan fel.
		-2 Ingen giltig transformation (6-axel-orienteringstransformation) är aktiv.
		-3 Den första parametern (<Cntrl>) är negativ.
		-4 Det 1:a stället i den första parametern (<Cntrl>) är ogiltigt. Endast värdena 0 och 1 är tillåtna.
		-5 Det 10:e stället i den första parametern (<Cntrl>) är ogiltigt. Endast värdena 0 till 3 är tillåtna.
		-6 Det 100:e stället i den första parametern (<Cntrl>) är ogiltigt. Endast värdena 0 och 1 är tillåtna.
		-7 Det 1000:e stället i den första parametern (<Cntrl>) är ogiltigt. Endast värdena 0 till 3 är tillåtna.
		-8 Vid funktionen "Rikta upp verktyg direkt" är vinkeln γ för stor.
		-9 Vid funktionen "Rikta upp verktyg direkt" kränker minst en av de föreskrivna axelpositionerna en axelgräns.
		-10 Inget verktyg är aktivt.
		-11 Den begärda orienteringen kan inte ställas in.
		-12 Anpassningen av den fria axelvinkeln vid Hirthtandning är inte möjlig för den första eller enda lösningen.
		-13 Anpassningen av den fria axelvinkeln vid Hirthtandning är inte möjlig för den andra lösningen.
		-14 Anpassningen av den fria axelvinkeln vid Hirthtandning är inte möjlig för någon av de båda lösningarna.
	-15	Den första orienteringsaxeln är parametrerad som Hirthaxel.
	-16	Både den andra och även den tredje roterande axeln är parametrerad som Hirthaxel. Maximalt en av de båda axlarna får vara Hirthaxel.
	-17	Vid funktionen "Svänga direkt" är minst en av de föreskrivna axelpositionerna inte kompatibel med den tillhörande Hirthtandningen.

11.11 Modifikation av kompenseringsdata för vridbara verktyg

<Cntrl>:	Styr funktionens beteende	
	Datatyp:	INT
	Parametern <Cntrl> är decimalkodad (1:a till 1000:e ställe):	
	1:a ställe:	Det 1:a stället styr beteendet vid fel.
	xxx0	I händelse av fel (returvärde < 0) avbryts programbearbetningen och larm 14106 matas ut. Observera: Larmet matas också ut oberoende av värdet för det 1:a stället, när parametern <Cntrl> är negativ.
	xxx1	I händelse av fel (returvärde < 0) matas inget larm ut. Användaren har möjligheten i programmet, att själv reagera lämpligt.
	10:e ställe:	Styr beteendet när en orienteringsaxel med Hirthtandning finns. Observera: Denna parameter utvärderas endast vid funktionen "Rikta upp verktyg" (dvs. när det 100:e stället har värdet "0").
	xx0x	Axelpositionen rundas till den närmast liggande positionen.
	xx1x	Axelpositioner rundas så att avvikelsen för vinkeln β blir minimal från det programmerade värdet.
	xx2x	Axelpositioner rundas så att vinkeln β blir lika med det största möjliga värdet, som är mindre än det programmerade värdet (β avrundas neråt).
	xx3x	Axelpositioner rundas så att vinkeln β blir lika med det minsta möjliga värdet, som är större än det programmerade värdet (β avrundas uppåt).
	100:e ställe:	Anger vilken funktion som ska utföras, resp. vilken betydelse de båda följande parametrarna <W1> och <W2> har.
	x0xx	Funktion "Upprikta verktyg" Parameter <W1> och <W2> har följande betydelse: <W1> = β <W2> = γ De tillhörande vinklarna till orienteringsaxlarna beräknas.
	x1xx	Funktion "Upprikta verktyg direkt" <W1> är positionsuppgift för den andra orienteringsaxeln, <W2> positionsuppgift för den tredje orienteringsaxeln i en 6-axeltransformation. Positionen för den första orienteringsaxeln och vinklarna β och γ bestäms så att de är kompatibla med de båda positionsuppgifterna. Om inget fel uppträder, matas i systemvariabeln \$P_ORI_POS[<n>, <m>] alltid ut två lösningar. Därvid hänvisar det första indexet <n> (0 eller 1) till lösningen och den andra indexet <m> (0 ... 2) till orienteringsaxeln: \$P_ORI_POS[0/1, 0]: Position för den första orienteringsaxeln \$P_ORI_POS[0/1, 1]: Vinkel β \$P_ORI_POS[0/1, 2]: Vinkel γ Det kontrolleras om positionsuppgifterna <W1> och <W2> är kompatibla med eventuellt aktiva Hirthtandningar eller aktiva

11.11 Modifikation av kompenseringsdata för vridbara verktyg

			softwarelimits. Är detta inte fallet, returneras ett motsvarande felnummer (se parameter <RetVal>). Vid ett godtyckligt val av de båda vinklarna <W1> och <W2> ligger verktygets skär i allmänhet inte i bearbetsningsplanet. Vinkel γ , runt vilken skäret från bearbetsningsplanet är vridet, får inte vara större än det gränsvärde, som bestäms av settingdatum SD42999 \$SC_ORISOLH_INCLINE_TOL.
	1000:e ställe:	Anger hur positionerna för lösningarna eventuellt modifieras, när 100:e stället har värdet "0", dvs. vid funktionen "Upprikta verktyg".	
		0xxx	De beräknade axelpositionerna ska ligga så nära som möjligt vid de aktuella maskinaxelpositionerna.
		1xxx	De beräknade axelpositionerna ska för Moduloaxlar ligga så nära mitten på Moduloområdet som möjligt, för andra axlar så nära 0 som möjligt. För icke-Moduloaxlar betyder det att axelpositionerna reduceras till området -180° ... +180°.
		2xxx	De beräknade axelpositionerna ska oberoende av axeltypen reduceras till området -180° ... +180°.
<W1>:	Första vinkeln		
	Betydelsen resulterar ur det 100:e stället för parametern <Cntrl>.		
	Datatyp:	REAL	
<W1>:	Andra vinkeln		
	Betydelsen resulterar ur det 100:e stället för parametern <Cntrl>.		
	Datatyp:	REAL	

Märk

Ej programmerade parametrar har standardvärdet "0".

Ytterligare informationer

Antalet hittade lösningar tillsammans med ytterligare statusinformationer vid utförandet av funktionen ORISOLH kan läsas via följande systemvariabel:

Systemvariabel	Betydelse
\$P_ORI_POS [<n>, <m>]	Levererar vinklarna för de orienteringsaxlar, som resulterar ur orienteringsprogrammeringen.
	<n>: Index för lösningen
	Värdeområde: 0, 1
	<m>: Index för orienteringsaxeln
	Värdeområde: 0 ... 2 Ordningsföljden för orienteringsaxlarna (1 ... 3) hänför sig till definitionen för axlarna i \$NT_ROT_AX_NAME.
	Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" innehåller variablerna \$P_ORI_POS[0/1, 1] och P_ORI_POS[0/1, 2] de till de båda lösningarna hörande värdena för de båda vinklarna β och γ. Den första i \$P_ORI_POS[<n>, <m>] införda lösningen, dvs. med index <n> = 0, är alltid den som väljs av styrningen, när man direkt uppsöker den begärda orienteringen. Det andra indexet <m> hänför sig därvid till orienteringsaxeln, dvs. till \$NT_ROT_AX_NAME. De i \$P_ORI_POS[<n>, <m>] införda axelpositionerna tar hänsyn till de offsets, som är införda i \$NK_OFF och \$NK_OFF_FINE, dvs. dessa axelvinklar kan i de följande blocken utan vidare modifikation användas till att ställa in den önskade orienteringen. Är en roterande axel en Hirthaxel, rundas Lösningenspositionerna till den närmast ligande spärrläge för Hirthtandningen. I systemvariabeln \$P_ORI_DIFF kan för Hirthtandade roterande axlar axelpositionernas differenser för de exakta lösningarna och de till de Hirth-spärrade anpassade lösningarna läsas.
\$P_ORI_DIFF [<n>, <m>]	Levererar differensen mellan de exakta och de i \$P_ORI_POS till förfogande ställda positionerna för orienteringsaxlarna, som resulterar ur orienteringsprogrammeringen.
	<n>: Index för lösningen
	Värdeområde: 0, 1
	<m>: Index för orienteringsaxeln
	Värdeområde: 0 ... 2 Ordningsföljden för orienteringsaxlarna (1 ... 3) hänför sig till definitionen för axlarna i \$NT_ROT_AX_NAME.
	Innehållet kan endast vara skilt från noll, när positionerna spärras (Hirth-tandning), dvs. när systemdatum \$NT_HIRTH_INCR för den beträffande axeln är skilt från noll och när denna axel är en manuell roterande axel.

Systemvariabel	Betydelse																		
\$P_ORI_SOL	Blir vid en orienteringstransformation med mer än en orienteringsaxeln de axelvinklar beräknade, som ska leda till en föreskriven orientering, finns det i allmänhet mer än en lösning. I systemvariabeln \$P_ORI_SOL ingår antalet giltiga lösningar tillsammans med ytterligare statusinformationer. Innehållet i \$P_ORI_SOL är kodat på följande sätt:																		
Värden < 0	Allmänna feltilstånd <table> <tr> <td>-1</td><td>För den aktiva transformationen beräknades ännu inga lösningar (anrop saknas för ORISOLH).</td></tr> <tr> <td>-2</td><td>Ingen transformation är aktiv, eller den aktiva transformationen är ingen orienteringstransformation (6-axel-transformation), som kan leverera positioner till en föreskriven orienteringsprogrammering.</td></tr> <tr> <td>-4</td><td>Den önskade orienteringen kan inte ställas in med den angivna kinematiken.</td></tr> <tr> <td>-5</td><td>Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" hittades ingen lösning.</td></tr> <tr> <td>-6</td><td>Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" är vinkeln γ för stor.</td></tr> <tr> <td>-7</td><td>Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" föreskrevs en vinkel, som på grund av Hirthandningen inte kan ställas in.</td></tr> <tr> <td>-8</td><td>Den första orienteringsaxeln (Frameaxel) får inte vara parameterad som Hirthaxel.</td></tr> <tr> <td>-9</td><td>Både den andra och även den tredje roterande axeln är parameterad som Hirthaxel. Max. en av de båda axlarna får vara Hirthaxel.</td></tr> <tr> <td>-10</td><td>Ingen anpassning av lösningen/lösningarna till Hirthaxeltandningen hittades.</td></tr> </table>	-1	För den aktiva transformationen beräknades ännu inga lösningar (anrop saknas för ORISOLH).	-2	Ingen transformation är aktiv, eller den aktiva transformationen är ingen orienteringstransformation (6-axel-transformation), som kan leverera positioner till en föreskriven orienteringsprogrammering.	-4	Den önskade orienteringen kan inte ställas in med den angivna kinematiken.	-5	Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" hittades ingen lösning.	-6	Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" är vinkeln γ för stor.	-7	Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" föreskrevs en vinkel, som på grund av Hirthandningen inte kan ställas in.	-8	Den första orienteringsaxeln (Frameaxel) får inte vara parameterad som Hirthaxel.	-9	Både den andra och även den tredje roterande axeln är parameterad som Hirthaxel. Max. en av de båda axlarna får vara Hirthaxel.	-10	Ingen anpassning av lösningen/lösningarna till Hirthaxeltandningen hittades.
-1	För den aktiva transformationen beräknades ännu inga lösningar (anrop saknas för ORISOLH).																		
-2	Ingen transformation är aktiv, eller den aktiva transformationen är ingen orienteringstransformation (6-axel-transformation), som kan leverera positioner till en föreskriven orienteringsprogrammering.																		
-4	Den önskade orienteringen kan inte ställas in med den angivna kinematiken.																		
-5	Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" hittades ingen lösning.																		
-6	Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" är vinkeln γ för stor.																		
-7	Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" föreskrevs en vinkel, som på grund av Hirthandningen inte kan ställas in.																		
-8	Den första orienteringsaxeln (Frameaxel) får inte vara parameterad som Hirthaxel.																		
-9	Både den andra och även den tredje roterande axeln är parameterad som Hirthaxel. Max. en av de båda axlarna får vara Hirthaxel.																		
-10	Ingen anpassning av lösningen/lösningarna till Hirthaxeltandningen hittades.																		
Värden > 0 1:a ställe	Antal matematiskt möjliga lösningar utan hänsynstagande till axelgränser och ev. felvillkor. <table> <tr> <td>0</td><td> Det existerar inte någon lösning, dvs. den begärda orienteringen kan inte ställas in. Detta fall kan ha tre olika orsaker: - Den begärda orienteringen kan principiellt inte uppnås på grund av maskinkinematiken (ej rätvinkligt placerade orienteringsaxlar) inte heller vid valfritt förflyttningsområde för orienteringsaxlarna. I detta fall är det 10:e och 100:e stället i \$P_ORI_SOL båda noll, de till orienteringsaxlarna tillordnade statusvariablerna \$P_ORI_STAT innehåller värdet "-4". - De beräknade lösningarna kan inte uppnås, eftersom de skulle kränka axelbegränsningarna. De positioner för orienteringsaxlarna, som skulle resultera utan axelbegränsningar, kan läsas i \$P_ORI_POS. - Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" föreskrevs axelpositioner på ett sådant sätt, att antingen orienteringsvektorn eller orienteringsnormalen till verktyget är parallellt uppriktad till </td></tr> </table>	0	Det existerar inte någon lösning, dvs. den begärda orienteringen kan inte ställas in. Detta fall kan ha tre olika orsaker: - Den begärda orienteringen kan principiellt inte uppnås på grund av maskinkinematiken (ej rätvinkligt placerade orienteringsaxlar) inte heller vid valfritt förflyttningsområde för orienteringsaxlarna. I detta fall är det 10:e och 100:e stället i \$P_ORI_SOL båda noll, de till orienteringsaxlarna tillordnade statusvariablerna \$P_ORI_STAT innehåller värdet "-4". - De beräknade lösningarna kan inte uppnås, eftersom de skulle kränka axelbegränsningarna. De positioner för orienteringsaxlarna, som skulle resultera utan axelbegränsningar, kan läsas i \$P_ORI_POS. - Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" föreskrevs axelpositioner på ett sådant sätt, att antingen orienteringsvektorn eller orienteringsnormalen till verktyget är parallellt uppriktad till																
0	Det existerar inte någon lösning, dvs. den begärda orienteringen kan inte ställas in. Detta fall kan ha tre olika orsaker: - Den begärda orienteringen kan principiellt inte uppnås på grund av maskinkinematiken (ej rätvinkligt placerade orienteringsaxlar) inte heller vid valfritt förflyttningsområde för orienteringsaxlarna. I detta fall är det 10:e och 100:e stället i \$P_ORI_SOL båda noll, de till orienteringsaxlarna tillordnade statusvariablerna \$P_ORI_STAT innehåller värdet "-4". - De beräknade lösningarna kan inte uppnås, eftersom de skulle kränka axelbegränsningarna. De positioner för orienteringsaxlarna, som skulle resultera utan axelbegränsningar, kan läsas i \$P_ORI_POS. - Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" föreskrevs axelpositioner på ett sådant sätt, att antingen orienteringsvektorn eller orienteringsnormalen till verktyget är parallellt uppriktad till																		

11.11 Modifikation av kompenseringsdata för vridbara verktyg

Systemvariabel	Betydelse						
	den första orienteringsaxeln, vars position ska beräknas. Positionen för denna axel är i dessa fall inte definierad.						
	1 Det existerar en lösning. Detta fall kan ha tre olika orsaker: • Det finns på grund av den föreskrivna orienteringen och maskinkinematiken även utan hänsynstagande till axelgränserna endast en lösning (matematiskt sett rör det sig om två sammanfallande lösningar). Detta fall uppträder vid ej rätvinklig kinematik i marginalen till orienteringsområdet. \$P_ORI_POS innehåller båda (identiska) lösningarna. • Det finns endast en lösning, eftersom en andra lösning är ogiltig på grund av kränkta axelgränser. Det giltiga lösningen är då alltid den första lösningen i \$P_ORI_POS. Den andra lösningen, som skulle resultera utan hänsynstagandet till axelgränserna, kan också läsas i \$P_ORI_POS. • Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" är det normalfallet. Till de föreskrivna axelpositionerna för två orienteringsaxlar finns det i allmänhet endast en giltig position till den orienteringsaxel som saknas och ska beräknas.						
	2 Det existerar två lösningar.						
	8 Det existerar oändligt många lösningar dvs. positionen för en orienteringsaxel (polaxeln) är godtycklig. Av de båda möjliga lägena för den andra axeln/de andra axlarna är dock en utesluten på grund av kränkta axelgränser.						
	9 Det existerar oändligt många lösningar dvs. positionen för en orienteringsaxel (polaxeln) är ej bestämd. Den obestämda axeln kan fastställas ur det 100:e stället eller ur systemvariabeln \$P_ORI_STAT.						
Värden > 0 10:e ställe	Bitkodade indikering för kränkta axelgränser. Den exakta felorsaken kan fastställas ur systemvariabeln \$P_ORI_STAT. <table> <tr> <td>Bit 0 (värde 10):</td><td>För minst en lösning är minst en axelgräns för den 1:a orienteringsaxeln kränkt.</td></tr> <tr> <td>Bit 1 (värde 20):</td><td>För minst en lösning är minst en axelgräns för den 2:a orienteringsaxeln kränkt.</td></tr> <tr> <td>Bit 2 (värde 40):</td><td>För minst en lösning är minst en axelgräns för den 3:e orienteringsaxeln kränkt.</td></tr> </table>	Bit 0 (värde 10):	För minst en lösning är minst en axelgräns för den 1:a orienteringsaxeln kränkt.	Bit 1 (värde 20):	För minst en lösning är minst en axelgräns för den 2:a orienteringsaxeln kränkt.	Bit 2 (värde 40):	För minst en lösning är minst en axelgräns för den 3:e orienteringsaxeln kränkt.
Bit 0 (värde 10):	För minst en lösning är minst en axelgräns för den 1:a orienteringsaxeln kränkt.						
Bit 1 (värde 20):	För minst en lösning är minst en axelgräns för den 2:a orienteringsaxeln kränkt.						
Bit 2 (värde 40):	För minst en lösning är minst en axelgräns för den 3:e orienteringsaxeln kränkt.						
Värden > 0 100:e ställe	Bitkodad indikering för ej definierade axelpositioner (kan endast uppträda när det finns oändligt många lösningar, dvs. när det 1:a stället är lika med "9"). <table> <tr> <td>Bit 0 (värde 100):</td><td>Positionen för den 1:a orienteringsaxeln är inte definierad.</td></tr> <tr> <td>Bit 1 (värde 200):</td><td>Positionen för den 2:a orienteringsaxeln är inte definierad.</td></tr> <tr> <td>Bit 2 (värde 400):</td><td>Positionen för den 3:e orienteringsaxeln är inte definierad.</td></tr> </table>	Bit 0 (värde 100):	Positionen för den 1:a orienteringsaxeln är inte definierad.	Bit 1 (värde 200):	Positionen för den 2:a orienteringsaxeln är inte definierad.	Bit 2 (värde 400):	Positionen för den 3:e orienteringsaxeln är inte definierad.
Bit 0 (värde 100):	Positionen för den 1:a orienteringsaxeln är inte definierad.						
Bit 1 (värde 200):	Positionen för den 2:a orienteringsaxeln är inte definierad.						
Bit 2 (värde 400):	Positionen för den 3:e orienteringsaxeln är inte definierad.						

11.11 Modifikation av kompenseringsdata för vridbara verktyg

Systemvariabel	Betydelse
	Beteckningarna 1:a, 2:a och 3:e orienteringsaxeln hänför sig till definitionen av axlarna i \$NT_ROT_AX_NAME.

11.11 Modifikation av kompenseringsdata för vridbara verktyg

Systemvariabel	Betydelse
\$P_ORI_STAT [<n>]	Levererar för varje av de maximalt 3 orienteringsaxlarna status efter anrop av ORISOLH.
	<n>: Index för orienteringsaxeln (motsvarar index för den beträffande orienteringsaxeln i \$NT_ROT_AX_NAME)
	Värdeområde: 0 ... 2
	Ordningsföljden för orienteringsaxlarna (1 ... 3) hänförs till definitionen för axlarna i \$NT_ROT_AX_NAME.
	Innehållet i \$P_ORI_STAT är kodat på följande sätt:
	Värden < 0 Allmänna feltilstånd
	-1 Status är inte definierad (anrop saknas av ORISOLH).
	-2 Ingen transformation är aktiv, eller den aktiva transformationen är ingen orienteringstransformation (6-axel-transformation), som kan leverera positioner till en föreskriven orienteringsprogrammering.
	-3 Axeln ingår inte i den aktiva transformationen.
	-4 Positionen för axeln kan inte beräknas, eftersom med den givna kinematiken kan den begärda orienteringen inte heller vid valfritt antaget förflytningsområde för axeln uppnås.
	-5 Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" föreskrevs axelpositioner på ett sådant sätt, att antingen orienteringsvektorn eller orienteringsnormalen till verktyget är parallellt uppriktad till den första orienteringsaxeln, vars position ska beräknas. Positionen för denna axel är i dessa fall inte definierad.
	-6 Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" är vinkeln γ för stor.
	-7 Vid anrop av funktionen ORISOLH i mode "Rikta upp verktyg direkt" föreskrevs en vinkel, som på grund av Hirthtandningen inte kan ställas in.
	-8 Den första orienteringsaxeln (Frameaxel) får inte vara parameterad som Hirthaxel.
	-9 Både den andra och även den tredje roterande axeln är parameterad som Hirthaxel. Max. en av de båda axlarna får vara Hirthaxel.
	-10 Ingen anpassning av lösningen/lösningarna till Hirthaxeltandningen hittades.
Värden > 0 1:a ställe	Bitkodade indikering för kränkta axelgränser i den första lösningen.
	Bit 0 (värde 1): Den första lösningen kränker den undre axelgränsen.
	Bit 1 (värde 2): Den första lösningen kränker den övre axelgränsen.
Värden > 0 10:e ställe	Bitkodade indikering för kränkta axelgränser i den andra lösningen.
	Bit 0 (värde 10): Den andra lösningen kränker den undre axelgränsen.
	Bit 1 (värde 20): Den andra lösningen kränker den övre axelgränsen.
Värden > 0	Indikering av en icke-definierad axelposition.

Systemvariabel	Betydelse		
	100:e ställe	Bit 0 (värde 100):	Positionen för orienteringsaxeln är inte definierad, dvs. den begärda orienteringen uppnås med varje godtycklig inställning av den roterande axeln (polläge). Denna information ingår också i systemvariabeln \$P_ORI_SOL.
	Av de felnummer, som indikerar en kränkning av axelgränserna, kan flera uppträda samtidigt. Eftersom det vid kränkningen av en axelgräns försöks, att genom addition resp. subtraktion av mångfalden av 360° uppnå en position inom de tillåtna axelgränserna, är det – om detta inte är möjligt – inte entydigt definierat, om den undre eller den övre axelgränsen kränktes. Finns det för den begärda orienteringen ingen lösning (\$P_ORI_SOL = 0), är status för de i transformationen ingående orienteringsaxlarna "0".		

Märk**\$NT_ROT_AX_NAME**

Med denna systemvariabel hänvisas till maximalt 3 axlar, som tjänar till inställning av orienteringen. Den innehåller namnen för de följdtermer (\$NK_NAME), som definierar de maskinaxlar (roterande axlar), som måste utföra de orienteringsrörelser, som resulterar ur en kinematisk transformation. Den ordningsföljd, i vilken de maximalt tre roterande axlarna ingår i denna systemvariabel, är utan betydelse för maskinkinematiken, eftersom denna härleds ur strukturen för de kinematiska följderna. Men då den definierar den ordningsföljd, med vilken de andra variablerna har åtkomst till de roterande axlarna, måste ordningsföljden för orienteringsaxlarna i \$NT_ROT_AX_NAME stämma överens med kinematikbeskrivningen.

Märk**Statusinformationer**

Statusinformationerna, som t.ex. visar att ingen orientering kan uppnås eller endast skulle kunna uppnås under kränkning av de relevanta axelgränserna, leder inte till ett NC-larm. Det är användaren som bär ansvaret för att reagera adekvat på de nämnda villkoren.

11.11.2 Aktivera modifikation av kompenseringsdata för vridbara verktyg (CUTMOD, CUTMODK)

Modifikationen av kompenseringsdata för vridbara verktyg aktiveras i NC-programmet via språkkommandot CUTMOD (i förbindelse med orienterbara verktygsbärare) resp. CUTMODK (för orienteringstransformationer, som definierades med hjälp av kinematiska följder).

Märk

Eftersom orienterbara verktygsbärare och orienteringstransformationer, som definierades med hjälp av kinematiska följder, inte kan vara aktiva samtidigt, leder det inte till konflikter mellan de båda varianterna.

Syntax

CUTMOD = <Value>

resp.

CUTMODK = <Command>

Betydelse

CUTMOD:	Funktionsanrop i förbindelse med orienteringsbara verktygsbärare		
<Value>:	Tillordnar värde		
	Datatyp:	INT	
	Värde:	0	Funktionen är inaktiverad. De av systemvariabeln \$P_AD... levererade värdena är lika de korresponderande verktygsparametrarna.
		> 0	Funktionen aktiveras om en orienterbar verktygsbärare med det angivna numret är aktiv dvs. aktiveringen är bunden till en bestämd orienterbar verktygsbärare. De av systemvariabel \$P_AD... levererade värdena är jämfört med de korresponderande verktygsparametrarna eventuellt modifierade beroende av den aktiva vridningen. Inaktiveringen av den betecknade orienterbara verktygsbäraren inaktiverar funktionen temporärt, aktiveringen av en annan orienterbar verktygsbärare inaktiverar den permanent. I det första fallet blir funktionen därför vid förnyat val av en lika orienterbar verktygsbärare åter aktiverad, i det andra fallet är ett förnyat val nödvändigt också när vid en senare tidpunkt den orienterbara verktygsbäraren med det angivna numret aktiveras på nytt. Funktionen påverkas inte av Reset.
		-1	Funktionen aktiveras alltid om en orienterbar verktygsbärare är aktiv. Vid byte av verktygsbäraren eller vid dess bortval och ett senare förnyat val måste CUTMOD inte ställas in på nytt.
		-2	Funktionen aktiveras alltid, om en orienterbar verktygsbärare är aktiv vars nummer är lika det för den aktuellt aktiva orienterbara verktygsbäraren. Är ingen orienterbar verktygsbärare aktiv, är det lika med CUTMOD=0. Är en orienterbar verktygsbärare aktiv är det lika med ett omedelbart angivande av det aktuella verktygsbärarnumret.
		< -2	Värden mindre än 2 ignoreras dvs. detta fall behandlas som om CUTMOD inte vore programmerad. Observera: Detta värdeområde bör inte användas eftersom det är reserverat för eventuella senare kompletteringar.
CUTMODK:	Funktionsanrop i förbindelse med orienteringstransformationer, som definierades med hjälp av kinematiska följder		

11.11 Modifikation av kompenseringsdata för vridbara verktyg

<Command>:	Tillordnat kommando		
	Datatyp:	STRING	
	Värde:	"NEW"	Det sparas för "Modifikation av kompenseringsdata" relevanta tillstånd för en aktiv med kinematiska följder definierad transformation, namnet för transformationen och den aktuella konturframen. Observera: Detta kommando är inte tillåtet, när en lämplig transformation (TRAORI_DYN, TRAORI_STAT eller TRAANG_K) är aktiv.
		"OFF"	Kopplar från den aktiva "Modifikation av kompenseringsdata". De tidigare med "NEW" lagrade data förblir bibehållna. Observera: Detta kommando är också tillåtet, när CUTMODK inte är aktiv. Det blir då utan verkan. Ett eventuellt förekommande datablock för "Modifikation av kompenseringsdata" förblir bibehållet.
		"ON"	Med detta kommando blir "Modifikation av kompenseringssdata" med ett tidigare med kommandot "NEW" sparad datablock åter aktiverat. Är vid utförandet av detta kommando en transformation med namnet för det sparade datablocket aktiv, blir "Modifikation av kompenseringsdata" genast verksamt. I annat fall fördröjs aktiveringen, tills en aktiv transformation blir aktiverad.
		"CLEAR"	Kopplar som kommandot "OFF" från "Modifikation av kompenseringsdata" och raderar dessutom det sparade datablocket. Observera: Detta kommando är också tillåtet, när CUTMODK inte är aktiv.

Märk**SD42984 \$SC_CUTDIRMOD**

Den via kommandot CUTMOD resp. CUTMODK aktiverbara funktionen ersätter den via settingdatum SD42984 \$SC_CUTDIRMOD aktiverbara funktionen. Denna funktion står dock fortfarande oförändrad till förfogande. Men då det inte är lämpligt att använda båda funktionerna parallellt kan den bara aktiveras när CUTMOD är lika med noll och CUTMODK lika med nollstringen.

Ytterligare informationer**Läsa modifierade kompenseringsdata**

De modifierade kompenseringsdata ställs i de följande systemvariablerna och BTSS-variablerna till förfogande:

Betydelse	Systemvariabel	BTSS-variabel
Skärläge	\$P_AD[2]	cuttEdgeParam2
Hållarvinkel	\$P_AD[10]	cuttEdgeParam10

Betydelse	Systemvariabel	BTSS-variabel
Skärriktning	\$P_AD[11]	cuttEdgeParam11
Frivinkel	\$P_AD[24]	cuttEdgeParam24

Data är jämförda med de korresponderande verktygsparametrarna (\$TC_DP2[...], ...) alltid modifierade, när funktionen "Modifikation av kompenseringsdata för vridbara verktyg" aktiverades med kommandot `CUTMOD` resp. `CUTMODK` och verktyget vreds av en orienterbar verktygsbärare resp. en lämplig orienteringstransformation.

Ytterligare funktionsrelevanta systemvariabler

Systemvariabel	Betydelse
\$P_CUTMOD_ANG / \$AC_CUTMOD_ANG	Levererar den vinkel, runt vilken ett verktyg vreds i det aktiva bearbetningsplanet och som ligger till grund för fastställandet av modifierade skärdata vid funktionerna <code>CUTMOD</code> resp. <code>CUTMODK</code> .
\$P_CUTMOD / \$AC_CUTMOD	Läser det aktuellt giltiga värdet, som programmerades sist med kommandot <code>CUTMOD</code> (nummer för verktygsbärare, för vilken modifikation av kompenseringsdata ska aktiveras). Var det sist programmerade värdet <code>CUTMOD=-2</code> (aktivering med den aktuellt aktiva orienterbara verktygsbäraren), returneras till systemvariabeln inte värdet "-2", utan numret för den vid tidpunkten för programmeringen aktiva orienterbara verktygsbäraren.
\$P_CUTMODK / \$AC_CUTMODK	Läser namnet för den transformation, under vilken det aktuellt giltiga datablocket för "Modifikation av kompenseringsdata" lades till.
\$P_CUT_INV / \$AC_CUT_INV	Levererar värdet <code>TRUE</code> , när verktyget är så vridet att spindelrotationsriktningen måste inverteras. Därtill måste i det block till vilket respektive läsoperation hänförs sig de följande fyra villkoren vara uppfyllda: 1. Ett svarv- eller slipverktyg aktivt (verktygsdata 400 till 599 och / eller SD42950 \$SC_TOOL_LENGTH_TYPE = 2). 2. Modifikationen av kompenseringsdata aktiverades med kommandot <code>CUTMOD</code> resp. <code>CUTMODK</code>. 3. Det är en orienterbar verktygsbärare resp. en med kinematiska följder definierad orienteringstransformation aktiv, som valdes av kommandot <code>CUTMOD</code> resp. <code>CUTMODK</code>. 4. Verktyget vrids av den orienterbara verktygsbäraren resp. av den kinematiska orienteringstransformationen så att den resulterande normalen till verktygsskåret har vridits med mer än 90° (typiskt värde 180°) i förhållande till utgångsläget. Är minst en av de nämnda fyra villkoren inte uppfyllt levererar variabeln värdet <code>FALSE</code>. För verktyg vars skärläge inte är definierat är värdet i variabeln alltid <code>FALSE</code>.

Systemvariabel	Betydelse																		
\$P_CUTMOD_ERR	Feltillstånd efter det sista anropet av CUTMOD-funktionen CUTMOD-funktionen kan också anropas implicit vid verktygsväxling. Variabeln återställs på noll vid Reset. Den återställs först vid varje verktygsväxling och skrivs eventuellt ny. Variabeln är bit-kodad. Bitsen har följande betydelser: <table> <tr> <td>Bit 0:</td><td>För det aktiva verktyget är ingen giltig skärriktning definierad.</td></tr> <tr> <td>Bit 1:</td><td>Skärvinklarna (frivinkel och hållarvinkel) för det aktiva verktyget är båda noll.</td></tr> <tr> <td>Bit 2:</td><td>Frivinkeln för det aktiva verktyget har ett otillåtet värde ($< 0^\circ$ eller $> 180^\circ$).</td></tr> <tr> <td>Bit 3:</td><td>Hållarvinkeln för det aktiva verktyget har ett otillåtet värde ($< 0^\circ$ eller $> 90^\circ$).</td></tr> <tr> <td>Bit 4:</td><td>Plattvinkeln för det aktiva verktyget har ett otillåtet värde ($< 0^\circ$ eller $> 90^\circ$).</td></tr> <tr> <td>Bit 5:</td><td>Kombinationen skärläge - hållarvinkel för det aktiva verktyget är otillåten (vid skärlägena 1 till 4 måste hållarvinkeln vara $\leq 90^\circ$, vid skärlägena 5 till 8 måste den vara $\geq 90^\circ$).</td></tr> <tr> <td>Bit 6:</td><td>Otillåten vridning av det aktiva verktyget. Verktyget vreds med $\pm 90^\circ$ (med en tolerans på ungefär 1°) ur det aktiva bearbetningsplanet. Därigenom är skärläget inte längre definierat i bearbetningsplanet.</td></tr> <tr> <td>Bit 7:</td><td>Skärplattan ligger inte i bearbetningsplanet och vinkeln mellan skärplatta och bearbetningsplan överskrider den med settingdatum SD42998 \$SC_CUTMOD_PLANE_TOL föreskrivna övre gränsen.</td></tr> <tr> <td>Bit 8:</td><td>Skärplattan ligger inte i bearbetningsplanet. Beloppet för vinkelns α är större än 1°. Vinkeln α är vridvinkeln runt den koordinataxel, som står lodrätt mot både vridaxeln för vinkeln β och även vridaxeln för vinkeln γ (vid G18 X-axeln).</td></tr> </table>	Bit 0:	För det aktiva verktyget är ingen giltig skärriktning definierad.	Bit 1:	Skärvinklarna (frivinkel och hållarvinkel) för det aktiva verktyget är båda noll.	Bit 2:	Frivinkeln för det aktiva verktyget har ett otillåtet värde ($< 0^\circ$ eller $> 180^\circ$).	Bit 3:	Hållarvinkeln för det aktiva verktyget har ett otillåtet värde ($< 0^\circ$ eller $> 90^\circ$).	Bit 4:	Plattvinkeln för det aktiva verktyget har ett otillåtet värde ($< 0^\circ$ eller $> 90^\circ$).	Bit 5:	Kombinationen skärläge - hållarvinkel för det aktiva verktyget är otillåten (vid skärlägena 1 till 4 måste hållarvinkeln vara $\leq 90^\circ$, vid skärlägena 5 till 8 måste den vara $\geq 90^\circ$).	Bit 6:	Otillåten vridning av det aktiva verktyget. Verktyget vreds med $\pm 90^\circ$ (med en tolerans på ungefär 1°) ur det aktiva bearbetningsplanet. Därigenom är skärläget inte längre definierat i bearbetningsplanet.	Bit 7:	Skärplattan ligger inte i bearbetningsplanet och vinkeln mellan skärplatta och bearbetningsplan överskrider den med settingdatum SD42998 \$SC_CUTMOD_PLANE_TOL föreskrivna övre gränsen.	Bit 8:	Skärplattan ligger inte i bearbetningsplanet. Beloppet för vinkelns α är större än 1° . Vinkeln α är vridvinkeln runt den koordinataxel, som står lodrätt mot både vridaxeln för vinkeln β och även vridaxeln för vinkeln γ (vid G18 X-axeln).
Bit 0:	För det aktiva verktyget är ingen giltig skärriktning definierad.																		
Bit 1:	Skärvinklarna (frivinkel och hållarvinkel) för det aktiva verktyget är båda noll.																		
Bit 2:	Frivinkeln för det aktiva verktyget har ett otillåtet värde ($< 0^\circ$ eller $> 180^\circ$).																		
Bit 3:	Hållarvinkeln för det aktiva verktyget har ett otillåtet värde ($< 0^\circ$ eller $> 90^\circ$).																		
Bit 4:	Plattvinkeln för det aktiva verktyget har ett otillåtet värde ($< 0^\circ$ eller $> 90^\circ$).																		
Bit 5:	Kombinationen skärläge - hållarvinkel för det aktiva verktyget är otillåten (vid skärlägena 1 till 4 måste hållarvinkeln vara $\leq 90^\circ$, vid skärlägena 5 till 8 måste den vara $\geq 90^\circ$).																		
Bit 6:	Otillåten vridning av det aktiva verktyget. Verktyget vreds med $\pm 90^\circ$ (med en tolerans på ungefär 1°) ur det aktiva bearbetningsplanet. Därigenom är skärläget inte längre definierat i bearbetningsplanet.																		
Bit 7:	Skärplattan ligger inte i bearbetningsplanet och vinkeln mellan skärplatta och bearbetningsplan överskrider den med settingdatum SD42998 \$SC_CUTMOD_PLANE_TOL föreskrivna övre gränsen.																		
Bit 8:	Skärplattan ligger inte i bearbetningsplanet. Beloppet för vinkelns α är större än 1° . Vinkeln α är vridvinkeln runt den koordinataxel, som står lodrätt mot både vridaxeln för vinkeln β och även vridaxeln för vinkeln γ (vid G18 X-axeln).																		

\$P_...: Förlöpsvariabler

\$AC_...: Huvudloppsvariabler

Alla huvudkörningsvariabler kan läsas i synkronaktioner. En läsåtkomst från förkörningen genererar ett fördekoderingsstopp.

Planväxling

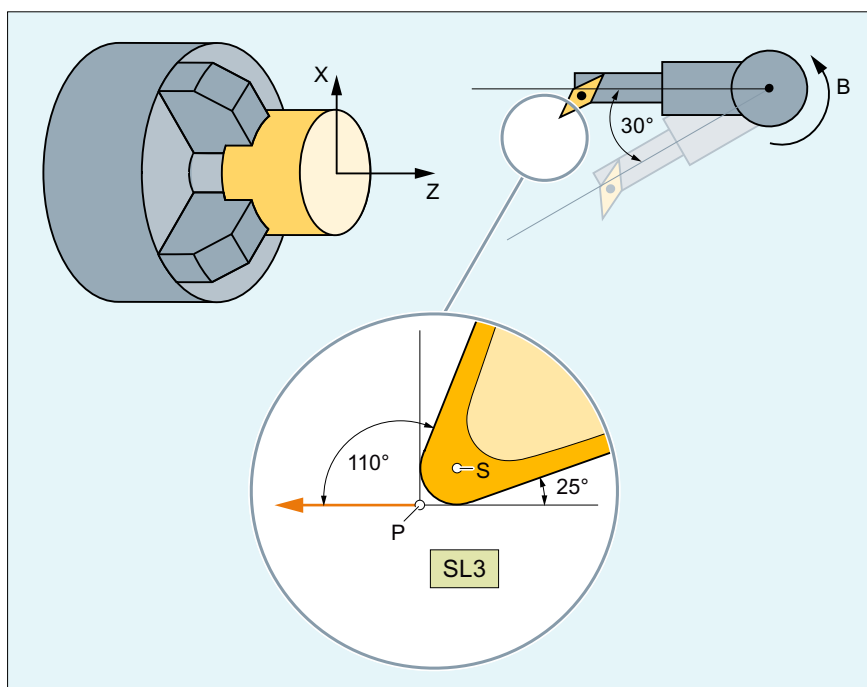
För bestämmande av modifierat skärläge, skärriktning och hållar- resp. frivinkel är betraktandet av skäret i det respektive planet (G17 - G19) utslagsgivande.

Innehåller dock settingdatum SD42940 \$SC_TOOL_LENGTH_CONST (byte av verktygslängdkomponenter vid planbyte) ett giltigt värde skilt från noll (plus eller minus 17, 18 eller 19), så bestämmer dess innehåll planet i vilket de relevanta storlekarna betraktas.

Denna prioritetsregel för settingdata över G-koden kan annulleras genom inställning av bit 18 i maskindatum \$MC_TOOL_PARAMETER_DEF_MASK. Dvs. när denna bit har ställts in, gäller alltid det med G-kommandot i grupp 6 definierade planet.

Verksamhet hos de modifierade skärdata

Det modifierade skärläget och den modifierade skärreferenspunkten blir genast verksamma vid programmeringen också för ett redan aktivt verktyg. Ett nyval av verktyget behövs inte här.



S: Skärmedelpunkt

P: Skärreferenspunkt

SL: Skärläge

Bild 11-4 Verktyg med skärläge 3 och en orienterbar verktygsbärare, som kan vrida verktyget runt B-axeln

Programkod	Kommentar
N10 \$TC_DP1[1,1]=500	
N20 \$TC_DP2[1,1]=3	; Skärläge
N30 \$TC_DP3[1,1]=12	
N40 \$TC_DP4[1,1]=1	
N50 \$TC_DP6[1,1]=6	
N60 \$TC_DP10[1,1]=110	; Hållarvinkel
N70 \$TC_DP11[1,1]=3	; Skärriktning
N80 \$TC_DP24[1,1]=25	; Frivinkel
N90 \$TC_CARR7[2]=0 \$TC_CARR8[2]=1 \$TC_CARR9[2]=0	; B-axel
N100 \$TC_CARR10[2]=0 \$TC_CARR11[2]=0 \$TC_CARR12[2]=1	; C-axel
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"	
N160 TCOABS CUTMOD=0	
N170 G18 T1 D1 TCARR=2	; X Y Z

11.11 Modifikation av kompenseringsdata för vridbara verktyg

Programkod	Kommentar		
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			

Siffervärdena i kommentarerna anger blockslutpositionerna i maskinkoordinater (MKS) i ordningsföljden X → Y → Z.

Förklaringar

I block N180 väljs först verktyget vid CUTMOD=0 och ej vriden orienterbar verktygsbärare. Eftersom alla offsetvektorer för den orienterbara verktygsbäraren är 0 uppsöks den position som motsvarar de i \$TC_DP3[1,1] och \$TC_DP4[1,1] angivna verktygslängderna.

I block N200 aktiveras den orienterbara verktygsbäraren med en vridning på 30° runt B-axeln. Eftersom skärläget inte modifieras på grund av CUTMOD=0 är som tidigare den gamla skärreferenspunkten bestämmande. Därför uppsöks i block N210 den position som bibehåller den gamla skärreferenspunkten i nollpunkten (dvs. vektorn (1, 12) vrids i Z/X-planet med 30°).

I block N260 är till skillnad från block N200 CUTMOD=2 verksam. På grund av vridningen av den orienterbara verktygsbäraren blir det modifierade skärläget 8. Därav följer också avvikande axelpositioner.

I blocken N220 resp. N270 aktiveras verktygsradiekompenseringen (WRK). Det olika skärläget i båda programstyckena har inget inflytande på slutpositionerna för de block i vilka WRK är aktiv, respektive positioner är därför identiska. Först i bortvalblocken N260 resp. N300 inverkar åter de olika skärlägena.

11.12 Arbeta med verktygsomgivningar

Översikt funktioner

- Spara verktygsomgivning (TOOLENV) (Sida 469)
- Radera verktygsomgivning (DELTOOLENV) (Sida 472)
- Läs T-, D- och DL-nummer (GETTENV) (Sida 473)
- Läs verktygslängder resp. -längdkomponenter (GETTCOR) (Sida 474)
- Ändra verktygskomponenter (SETTCOR) (Sida 480)

Översikt systemvariabler

- Läs informationer över sparade verktygsomgivningar (\$P_TOOLENVN, (\$P_TOOLENV) (Sida 474)

11.12.1 Spara verktygsomgivning (TOOLENV)

Funktionen TOOLENV tjänar till att spara alla aktuella tillstånd, som har betydelse för värderingen av de i minnet lagrade verktygsdata.

Det är i detalj följande data:

- Det aktiva G-kommandot i gruppen:
 - 6 (G17, G18, G19)
 - 56 (TOWSTD, TOWMCS, TOWWCS, TOWBCS, TOWTCS, TOWKCS)
- Den aktiva planaxeln
- Maskindata:
 - MD18112 \$MN_MM_KIND_OF_SUMCORR (egenskaper för summakompenseringar i TO-området)
 - MD20360 \$MC_TOOL_PARAMETER_DEF_MASK (definition av verktygsparametrarna)

- Settingdata:
 - SD42900 \$SC_MIRROR_TOOL_LENGTH (byte av förtecken verktygslängd vid spegling)
 - SD42910 \$SC_MIRROR_TOOL_WEAR (byte av förtecken verktygsförslitning vid spegling)
 - SD42920 \$SC_WEAR_SIGN_CUTPOS (förtecken för förslitning vid verktyg med skäranläggningar)
 - SD42930 \$SC_WEAR_SIGN (förtecken för förslitningen)
 - SD42935 \$SC_WEAR_TRANSFORM (transformationer för verktygskomponenter)
 - SD42940 \$SC_LENGTH_CONST (växling av verktygslängdkomponenterna vid planväxling)
 - SD42942 \$SC_TOOL_LENGTH_CONST_T (växling av verktygslängdkomponenterna för svarvverktyg vid planväxling)
 - SD42950 \$SC_TOOL_LENGTH_TYPE (tillordning av verktygslängdkomponenter oberoende av verktygstyp)
 - SD42954 \$SC_TOOL_ORI_CONST_M (växling av verktygsorienteringskomponenter för fräsverktyg vid planväxling)
 - SD42956 \$SC_TOOL_ORI_CONST_T (växling av verktygsorienteringskomponenter för svarvverktyg vid planväxling)
- Orienteringsandelen för den aktuella totalramen (rotationer och speglingar, inga nollpunktsförflyttningar eller skalningar)
- Orienteringsandelen och den resulterande längden för den aktiva orienterbara verktygsbäraren
- Orienteringsandelen och den resulterande längden för en aktiv transformation

Förutom nämnda data, som beskriver verktygets omgivning, lagras också T-nummer, D-nummer och DL-nummer för det aktiva verktyget, så att man senare har access till detta verktyg i samma omgivning som vid anrop av TOOLENV, utan att verktyget måste betecknas på nytt.

Syntax

```
<Status> = TOOLENV(<Name>)
```

Betydelse

TOOLENV(...):	Fördefinierad funktion för att spara en verktygsomgivning	
	Ensam i blocket:	ja

<Status>:	Funktionens returvärde. Negativa värden indikerar feltilstånd.		
	Datatyp:		INT
	Värde:	0	Funktion OK
		-1	Ingen minnesplats för verktygsomgivningar reserverad: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 Dvs. funktionaliteten "Verktygsomgivningar" finns inte.
		-2	Inga fria minnesplatser finns längre för verktygsomgivningar.
		-3	Nollstring som namn på en verktygsomgivning är inte tillåtet.
		-4	Ingen parameter (<Name>) angiven.
Parameter			
1	<Name>:	Namn, under vilket det aktuella datablocket ska lagras Finns redan ett datablock med samma namn, skrivs detta över. Status är i detta fall "0".	
Datatyp:		STRING	

Ytterligare informationer

Basmått/adaptermått - verktygslängdskompensering

Om adapterlängden eller basmåttet (skärspecifika parametrar \$TC_DP21, \$TC_DP22 och \$TC_DP23) ingår i beräkningen av verktygslängden, bestämmer vid aktiv verktygsmagasinförvaltning (endast tillgänglig med optionen "Verktygsförvaltning") värdet på maskindatumet:

MD18104 \$MN_MM_NUM_TOOL_ADAPTER (WZ-adapter i TO-området)

Eftersom en förändring av detta maskindatum endast kan bli verksam vid start av styrningen, sparas det inte i verktygsomgivningen.

Resulterande längd av orienterbara verktygsbärare och transformationer

Märk

Både vid orienterbara verktygsbärare och även vid transformationer finns det systemvariabler resp. maskindata, som verkar som extra verktygslängdskomponenter och som helt eller delvis underkastas de av dessa vridningar som kan utföras. De extra verktygslängdsandelarna, som resulterar därur, måste vid anrop av TOOLENV också lagras, eftersom de utgör en del av den omgivning, i vilken verktyget används.

Adaptertransformation

Adaptertransformationen är en egenskap hos verktygsadaptern och därmed hos hela verktyget. Den är därför inte beståndsdel i en verktygsomgivning, som kan användas på ett annat verktyg.

Genom att spara alla för bestämningen av verktygets totala längd nödvändiga data är det möjligt, att vid en senare tidpunkt beräkna den effektiva längden för ett verktyg, även om det vid denna tidpunkt inte längre är aktivt resp. när omgivningsvillkoren (t. ex. G-kommando eller settingdata) har förändrats. Likaså kan den effektiva längden för ett annat verktyg beräknas under antagande att det skulle användas under samma villkor som det verktyg, för vilket status sparades.

Maximalt antal datablock för verktygsomgivningar

Med maskindatum MD18116 \$MN_MM_NUM_TOOL_ENV bestäms, hur många datablock som maximalt kan lagras för att beskriva verktygsomgivningarna. Data ligger i TOA-området. De bibehålls också efter fränkopplingen av styrningen.

En datasäkring är inte möjlig. Det betyder att dessa data inte kan överföras mellan olika styrningar.

11.12.2 Radera verktygsomgivning (DELTOOLENV)

Med funktionen DELTOOLENV kan datablock för beskrivning av verktygsomgivningar raderas. Radera betyder att det inte längre finns någon åtkomst till det under ett visst namn sparade datablocket (ett åtkomstförsök leder till ett larm).

Märk

Datablock kan endast raderas med funktionen DELTOOLENV, genom en INITIAL.INI-download eller med en kallstart (NC-start med standard-maskindata). Det finns inga ytterligare automatiska raderingsförlöpp.

Syntax

```
<Status> = DELTOOLENV (<Name>)
<Status> = DELTOOLENV ()
```

Betydelse

DELTOOLENV (. . .) :		Fördefinierad funktion för radering av en verktygsomgivning	
		Ensam i blocket:	ja
<Status>:		Funktionens returvärde. Negativa värden indikerar feltillstånd.	
		Datatyp:	INT
		Värde:	0 Funktion OK
		-1	Ingen minnesplats för verktygsomgivningar reserverad: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 Dvs. funktionaliteten "Verktygsomgivningar" finns inte.
		-2	En verktygsomgivning med det angivna namnet existerar inte.
Parameter			
1	<Name>:	Namn på det datablock som ska raderas	
		Datatyp:	STRING
DELTOOLENV () :		DELTOOLENV () utan att ange ett namn raderar alla datablock för beskrivning av verktygsomgivningar	

11.12.3 Läs T-, D- och DL-nummer (GETTENV)

Funktionen GETTENV tjänar till att läsa de i en verktygsomgivning lagrade T-, D- och DL-numren.

Syntax

```
<Status> = GETTENV(<Name>, <TDDL>)
```

Betydelse

GETTENV(...):		Fördefinierad funktion för att läsa T-, D- och DL-nummer i ett datablock för beskrivning av en verktygsomgivning	
Ensam i blocket:		ja	
<Status>:		Funktionens returvärde. Negativa värden indikerar feltilstånd.	
		Datotyp:	INT
		Värde:	0 Funktion OK
			-1 Ingen minnesplats för verktygsomgivningar reserverad: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 Dvs. funktionaliteten "Verktygsomgivningar" finns inte.
			-2 En verktygsomgivning med det angivna namnet existerar inte.
Parameter			
1	<Name>:	Namn på det datablock, ur vilket T-, D- och DL-numren ska läsas	
		Datotyp:	STRING
2	<TDDL>:	Fältet för dessa resultatparametrar innehåller T-, D- och DL-nummer för det verktyg, vars verktygsomgivning är lagrad i det angivna datablocket:	
		• <TDDL> [0]: T-nummer • <TDDL> [1]: D-nummer • <TDDL> [2]: DL-nummer	
		Datotyp:	INT[3]
GETTENV(, <TDDL>), GETTENV("", <TDDL>):		Det är tillåtet, vid anrop at funktionen GETTENV att utelämna den första parametern eller att överlämna nollstringen som första parameter. I dessa båda specialfall returneras till <TDDL> T-, D- och DL-numren för det aktiva verktyget.	

11.12.4 Läs informationer över sparade verktygsomgivningar (\$P_TOOLENVN, \$P_TOOLENV)

Informationer över sparade verktygsomgivningar kan läsas via följande systemvariabler:

\$P_TOOLENVN:	Levererar antalet med hjälp av TOOLENV definierade (och ännu inte raderade) datablock för beskrivning av verktygsomgivningar			
	Syntax:	<n> = \$P_TOOLENVN		
	Betydelse:	<n>:	Antal definierade datablock	
			Datotyp:	INT
			Värdeområde:	0 ... MD18116 \$MN_MM_NUM_TOOL_ENV
Åtkomsten till denna systemvariabel är också tillåten när inga verktygsomgivningar är möjliga (MD18116 = 0). I detta fall är returvärdet "0".				

\$P_TOOLENV:	Levererar namnet för datablock nr <i> för beskrivning av en verktygsomgivning			
	Syntax:	<Name> = \$P_TOOLENV [<i>]		
	Betydelse:	<Name>:	Namn för datablock med nummer <i>	
			Datotyp:	STRING
		<i>:	Nummer för datablocket	
			Datotyp:	INT
			Värdeområde:	1 ... \$P_TOOLENVN
			Tillordningen av numren för datablocken är inte fix, utan kan förändra sig genom radering eller nyanläggning av datablock. Numreringen av datablocken görs internt. Hänvisar <i> till ett ej definierat datablock, returneras nollstringen. Är index <i> ogiltigt, dvs. <i> är mindre än 1 eller större än det maximala antalet datablock för verktygsomgivningar (MD18116 \$MN_MM_NUM_TOOLENV), matas följande larm ut: Larm 17020 "Otillåtet array-index1"	

11.12.5 Läs verktygslängder resp. -längdkomponenter (GETTCOR)

Funktionen GETTCOR tjänar till att läsa ut verktygslängder resp. verktygslängdskomponenter.

Därvid kan genom parametrering anges, vilka komponenter som det ska tas hänsyn till och under vilka användningsvillkor verktyget ska betraktas.

Syntax

```
<Status> = GETTCOR(<Len>[, <Comp>, <Stat>, <T>, <D>, <DL>])
```

Betydelse

GETTCOR(...):	Fördefinierad funktion för läsning av verktygslängder resp. verktygslängdskomponenter		
	Ensam i blocket:	ja	

<Status>:		Funktionens returvärde. Negativa värden indikerar feltilstånd.	
		Datatyp:	INT
		Värde:	0 Funktion OK
		-1	Ingen minnesplats för verkygssomgivningar reserverad: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 Dvs. funktionaliteten "Verkygssomgivningar" finns inte.
		-2	En verkygssomgivning med det under <Stat> angivna namnet existerar inte.
		-3	Ogiltig string i parametern <Comp>. Orsak till detta fel kan vara ogiltiga eller dubbelt programmerade tecken.
		-4	Ogiltigt T-Nummer
		-5	Ogiltigt D-Nummer
		-6	Ogiltigt DL-Nummer
		-7	Åtkomstförsök till en minnesmodul som inte finns.
		-8	Åtkomstförsök till en option som inte finns (t. ex. programmerbar verkygsorientering, verkygsförvaltning).
		-9	Stringen <Comp> innehåller en dubbelpunkt (kännetecken för specifikationerna för ett koordinatsystem), det följer dock inget giltigt tecken för beteckningen av det koordinatsystem som begärs.
Parameter			
1	<Len>:	Resultatvektor:	
		Datatyp:	REAL[11]
		Vektorkomponenterna är ordnade i följande ordningsföljd:	
		• <Len> [0]: Verkygstyp • <Len> [1]: Skärläge • <Len> [2]: Abskissa • <Len> [3]: Ordinata • <Len> [4]: Applikata • <Len> [5]: Verkygsradie	
		Som referenskoordinatsystem för längdkomponenterna gäller det i <Comp> och <Stat> definierade koordinatsystemet. Är inget koordinatsystemet definierat i <Comp> visas verkygslängderna i maskinkoordinatsystemet.	
		Tillordningen av abskissa, ordinata och applikata till geometriaxlarna är beroende av det i den använda verkygssomgivningen aktiva planet. Dvs. vid G17 är abskissan parallell till X, vid G18 till Z osv.	
		Komponenterna <Len>[6] till <Len>[10] innehåller de extra parametrarna, som kan anges för geometribeskrivningen av ett verktyg (t. ex. \$TC_DP7 till \$TC_DP11 för geometrin resp. de motsvarande komponenterna för förslitning resp. summa- och inställningskompenseringar).	
		Dessa 5 extra element och verkygsradien är definierade endast för komponenterna E, G, S och W. Deras värdering beror inte på <Stat>. De motsvarande värdena i <Len>[6] till <Len>[10] kan därför vara skilda från noll bara när minst en av de fyra nämnda komponenterna deltar. De övriga komponenterna har inget inflytande på resultatet. Måttuppgifterna hänför sig till grundsystemet för styrningen (inch resp. metriskt).	

2

<Comp>:

Verkygslängdskomponenter (option)																		
Datatyp:	STRING																	
Teckenföljden består av två delstringar, som är skilda av en dubbelpunkt från varandra.																		
Allm. form: "<SubStr_1> [: <SubStr_2>]"																		
<SubStr_1>:	Den första delstringen betecknar de verkygslängdskomponenter, som det ska tas hänsyn till vid verkygslängdsberäkningen. Ordningsföljden för tecknen i delstringen samt dess skrivsätt (stor eller liten) är valfri. Mellan tecknen kan valfritt många mellanslag eller tabulatortecken (white spaces) infogas.																	
	Observera: Tecknen i delstringen får inte dubbelt programmeras!																	
	Tecken:	<table><tr><td>-</td><td>Minustecken (endast som första tecken tillåtet!) Den totala verkygslängden minus komponenterna som specificeras i den följande stringen beräknas.</td></tr><tr><td>C</td><td>Adaptermått eller basmått (det av de båda alternativt existerande komponenterna, som är aktivt för det använda verktyget)</td></tr><tr><td>E</td><td>Inställningskompenseringar</td></tr><tr><td>G</td><td>Geometri</td></tr><tr><td>K</td><td>Kinematisk transformation (utvärderas endast vid generisk 3-, 4- och 5-axel-transformation!)</td></tr><tr><td>S</td><td>Summakompenseringar</td></tr><tr><td>T</td><td>Orienterbar verktygsbärare</td></tr><tr><td>W</td><td>Förslitning</td></tr></table>	-	Minustecken (endast som första tecken tillåtet!) Den totala verkygslängden minus komponenterna som specificeras i den följande stringen beräknas.	C	Adaptermått eller basmått (det av de båda alternativt existerande komponenterna, som är aktivt för det använda verktyget)	E	Inställningskompenseringar	G	Geometri	K	Kinematisk transformation (utvärderas endast vid generisk 3-, 4- och 5-axel-transformation!)	S	Summakompenseringar	T	Orienterbar verktygsbärare	W	Förslitning
	-	Minustecken (endast som första tecken tillåtet!) Den totala verkygslängden minus komponenterna som specificeras i den följande stringen beräknas.																
	C	Adaptermått eller basmått (det av de båda alternativt existerande komponenterna, som är aktivt för det använda verktyget)																
	E	Inställningskompenseringar																
	G	Geometri																
	K	Kinematisk transformation (utvärderas endast vid generisk 3-, 4- och 5-axel-transformation!)																
	S	Summakompenseringar																
	T	Orienterbar verktygsbärare																
W	Förslitning																	
Är den första delstringen (med undantag av white spaces) tom, så betyder det att den totala verkygslängden ska beräknas under hänsynstagande till alla komponenter. Samma gäller också när parametern <Comp> inte anges.																		
<SubStr_2>:	Den optionala andra delstringen betecknar det koordinatsystem, i vilket verkygslängden ska ges ut.																	
	Den andra delstringen består endast av ett enda relevant tecken.																	
	Tecken:	<table><tr><td>A</td><td>Inställbart koordinatsystem (ACS)</td></tr><tr><td>B</td><td>Baskoordinatsystem (BKS)</td></tr><tr><td>K</td><td>Verkygskoordinatsystem för den kinematiska transformationen (KCS)</td></tr><tr><td>M</td><td>Maskinkoordinatsystem (MKS)</td></tr><tr><td>T</td><td>Verkygskoordinatsystem (TCS)</td></tr><tr><td>W</td><td>Arbetsstyckskoordinatsystem (WKS)</td></tr></table>	A	Inställbart koordinatsystem (ACS)	B	Baskoordinatsystem (BKS)	K	Verkygskoordinatsystem för den kinematiska transformationen (KCS)	M	Maskinkoordinatsystem (MKS)	T	Verkygskoordinatsystem (TCS)	W	Arbetsstyckskoordinatsystem (WKS)				
	A	Inställbart koordinatsystem (ACS)																
	B	Baskoordinatsystem (BKS)																
	K	Verkygskoordinatsystem för den kinematiska transformationen (KCS)																
	M	Maskinkoordinatsystem (MKS)																
T	Verkygskoordinatsystem (TCS)																	
W	Arbetsstyckskoordinatsystem (WKS)																	
Anges inget koordinatsystem, görs utvärderingen i maskinkoordinatsystemet (MKS). De vridningar som det eventuellt ska tas hänsyn till bestäms genom den i <Stat> definierade verktygsomgivningen.																		

3	<_Stat>:	Namn för datablocket till beskrivning av en verktygsomgivning (option)	
		Datotyp:	STRING
		Är värdet för denna parameter nollstringen ("") eller anges den inte, används det aktuella tillståndet. Är inget verktyg specificerat, används det aktuella verktyget.	
4	<T>:	Internt T-nummer för verktyget (option)	
		Datotyp:	INT
		Är denna parameter inte angiven eller är dess värde "0", används det i <Stat> lagrade verktyget. Är värdet för denna parameter "-1", används T-numret för det aktiva verktyget. Det är också tillåtet, att explicit ange numret för det aktiva verktyget. Observera: Är <Stat> inte angivet, används det aktuella tillståndet som verktygsomgivning. Eftersom det med <T> = 0 hänvisas till det i verktygsomgivningen sparade T-numret, används i denna det aktiva verktyget, dvs. uppgifterna <T> = 0 och <T> = -1 betyder i detta specialfall detsamma.	
5	<D>:	Skärnummer för verktyget (option)	
		Datotyp:	INT
		Är denna parameter inte angiven eller är dess värde "0", rättar sig den använda D-numret efter källan till T-numret. Används T-numret från verktygsomgivningen, läses också D-numret från verktygsomgivningen, i annat fall det från det aktuellt aktiva verktyget.	
6	<DL>:	Nummer för den platsberoende kompenseringen (option)	
		Datotyp:	INT
		Är denna parameter inte angiven, rättar sig det använda DL-numret efter källan för T-numret. Används T-numret från verktygsomgivningen, läses också D-numret från verktygsomgivningen, i annat fall det från det aktuellt aktiva verktyget.	

Exempel

GETTCOR (_LEN)	Beräknar verktygslängden för det aktuellt aktiva verktyget i maskinkoordinatsystemet under hänsynstagande till alla komponenter.
GETTCOR (_LEN , "CGW:W")	Det beräknas verktygslängden bestående av adapter- resp. basmått, geometri och förslitning för det aktiva verktyget. Ytterligare andelar som t.ex. orienterbar verktygsbärare eller kinematiska transformationer tas det ingen hänsyn till. Utmatningen görs i arbetsstyckskoordinatsystemet.
GETTCOR (_LEN , "-K:B")	Det beräknas den totala verktygslängden för det aktiva verktyget utan hänsynstagande till längdkomponenterna för en eventuellt aktiv kinematisk transformation. Utmatning i baskoordinatsystemet.
GETTCOR (_LEN , ":M" , "Testenv1" , , 3)	Det beräknas den totala verktygslängden för det i verktygsomgivningen med namnet "Testenv1" lagrade verktyget i maskinkoordinatsystemet. Beräkningen sker dock oberoende av det lagrade skärnumret för skärnumret D3.

Ytterligare informationer

Adaptertransformation / orienterbar verktygsbärare / kinematisk transformation

Vridningar och komponentbyten, som också utförs av adaptertransformationen, den orienterbara verktygsbäraren och en kinematisk transformation, är beståndsdel i verktygsomgivningen. De utförs därför alltid, också när det inte ska tas hänsyn till de motsvarande längdkomponenterna. Önskas inte detta, måste de verktygsomgivningar definieras, vid vilka de motsvarande transformationerna inte är aktiva. I många fall (nämligen alltid när inga transformationer resp. orienterbara verktygsbärare används på en maskin), uppfyller de sparade datablocken för verktygsomgivningarna dessa villkor automatiskt, så att de inte måste beaktas speciellt av användaren.

Svarv- och slipverktyg: Beräkning av verktygslängden i beroende av MD20360 \$MC_TOOL_PARAMETER_DEF_MASK

Hur förslitningen resp. verktygslängden för svarv- och slipverktyg ska värderas i en eventuellt förefintlig diameteraxel, fastläggs i maskindatumet:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK (definition av verktygsparametrarna)

Bit	Värde
0	För svarv- och slipverktyg inberäknas förslitningsparametern för planaxeln som diame- tervärde:
	= 0 (default) nej
	= 1 ja
1	För svarv- och slipverktyg inberäknas verktygslängskomponenten för planaxeln som diame- tervärde:
	= 0 (default) nej
	= 1 ja

Har de ifrågavarande bits ställts in, värderas den tillhörande posten med faktorn 0,5. Denna värdering avsätter sig också i den av GETTCOR levererade verktygslängden.

Exempel:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK = 3

MD20100 \$MC_DIAMETER_AX_DEF (geometriaxel med planaxelfunktion) = "X"

X är diameteraxel (standard-svarvmaskinkonfiguration)

Programkod	Kommentar
N30 \$TC_DP1[1,1]=500	
N40 \$TC_DP2[1,1]=2	
N50 \$TC_DP3[1,1]=3.0	; Geometri L1
N60 \$TC_DP4[1,1]=4.0	
N70 \$TC_DP5[1,1]=5.0	
N80 \$TC_DP12[1,1]=12.0	; Förslitning 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)

Programkod	Kommentar
N150 R5=_LEN[4]	; 19.0 (= 5.0 + 14.0)
N160 M30	

Längdkomponenter från kinematisk transformation och orienterbar verktygsbärare

Tas det vid verktygslängdsberäkningen hänsyn till en **orienterbar verktygsbärare**, ingår följande vektorer i verktygslängdsberäkningen:

Typ	Vektorer
M	I1 och I2
T	I1, I2 och I3
P	Verktygslängd blir inte påverkad av den orienterbara verktygsbäraren.

Vid den generiska **5-axeltransformationen** ingår för Trafo-typerna 24 och 56 följande maskindata i verktygslängdsberäkningen:

Trafotyp	Maskindata
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

Trafotyp 56 (rörligt verktyg och rörligt arbetsstycke) motsvarar typ M för verktygsbärare.

Vid denna 5-axel-transformation motsvarar i de hittillsvarande programvarutillstånden vektorn MD24560/24660 \$MC_TRAFO5_JOINT_OFFSET_1/2 (vektor för den kinematiska förflyttningen för den 1:a/2:a 5-axeltransformationen i kanalen) summan av de båda vektorerna I₁ och I₃ vid den orienterbara verktygsbäraren av typ M.

För transformationen är i båda fallen endast summan relevant. Typen av sammansättning av de båda enskilda komponenterna är utan betydelse. Vid beräkningen av verktygslängden är det dock av betydelse, vilken andel som ska tillordnas verktyget och vilken verktygsbordet. Därför infördes maskindatum MD24558/24658 \$MC_TRAFO5_JOINT_OFFSET_PART_1/2 (vektor kinematisk förflyttning i bordet). Det motsvarar vektor I₃. Maskindatum MD24560/24660 \$MC_TRAFO5_JOINT_OFFSET_1/2 motsvarar nu inte längre summan av I₁ och I₃, utan bara vektor I₁. Är maskindatum MD24558/24658 \$MC_TRAFO5_JOINT_OFFSET_PART_1/2 lika med noll, så är beteendet identiskt med det tidigare beteendet.

kompatibilitet

Funktionen GETTCOR tjänar tillsammans med funktionerna TOOLENV och SETTCOR bland annat därtill att ersätta delar av funktionaliteten, som hittills var realiserad externt i mätcyklerna.

I mätcyklerna utvärderades endast en del av parametrarna, som slutningen bestämmer den effektiva verktygslängden. De nämnda funktionerna kan så parametreras, att mätcyklernas beteende vad beträffar verktygslängdsberäkningen kan reproduceras.

11.12.6 Ända verktygskomponenter (SETTCOR)

Funktionen SETTCOR tjänar därtill att förändra verktygskomponenter under hänsynstagande till alla randvillkor, som kan ingå i utvärderingen av de enskilda komponenterna.

Märk

Över terminologi: När det i det följande i samband med verktygslängd talas om verktygskomponenter, menas de vektoriellt betraktade komponenterna, av vilka den totala verktygslängden är sammansatt (t. ex. geometri eller förslitning). En sådan komponent är därför sammansatt av tre enskilda värden (L1, L2, L3), som i det följande betecknas som koordinatvärden.

Verktygskomponenten "Geometri" består alltså till exempel av de tre koordinatvärdena \$TC_DP3 till \$TC_DP5.

Syntax

```
<Status> = SETTCOR(<CorVal>, <Comp>, [<CorComp>, <CorMode>, <GeoAx>,  
<Stat>, <T>, <D>, <DL>])
```

Betydelse

SETTCOR(...):	Fördefinierad funktion för att ändra verktygskomponenter	
	Ensam i blocket:	ja

<Status>:		Funktionens returvärde. Negativa värden indikerar feltilstånd.	
		Datatyp:	INT
		Värde:	0 Funktion OK
		-1	Ingen minnesplats för verktygsgivningar reserverad: MD18116 \$MN_MM_NUM_TOOL_ENV = 0 Dvs. funktionaliteten "Verktygsgivningar" finns inte.
		-2	En verktygsgivning med det under <Stat> angivna namnet existerar inte.
		-3	Ogiltig string i parametern <Comp>. Orsak till detta fel kan vara ogiltiga eller dubbelt programmerade tecken.
		-4	Ogiltigt T-nummer.
		-5	Ogiltigt D-nummer.
		-6	Ogiltigt DL-nummer.
		-7	Åtkomstförsök till en minnesmodul som inte finns.
		-8	Åtkomstförsök till en option som inte finns (t. ex. programmerbar verktygsgivning, verktygsgivning).
		-9	Otillåtet siffervärde för parametern <CorComp>.
		-10	Otillåtet siffervärde för parametern <CorMode>.
		-11	Innehållet i parametrarna <Comp> och <CorComp> är motsä- gande.
		-12	Innehållet i parametrarna <Comp> och <CorMode> är motsä- gande.
		-13	Innehållet i parametern <GeoAx> betecknar ingen geometria- xel.
		-14	Skrivförsök på en inställningskompensering som inte finns.
Parameter			
1	<CorVal>:	Kompenseringsvektor I det genom <Stat> definierade arbetsstyckskoordinatsystemet (WKS) gäller följande tillordning: • <CorVal> [0]: Abskissa • <CorVal> [1]: Ordinata • <CorVal> [2]: Applikata Ska bara en verktygskomponent kompenseras (dvs. ingen vektorieell kompensering, se parameter <CorMode>), står kompenseringsvärdet alltid i <CorVal>[0], likgiltigt på vilken axel den verkar. Innehållet i de båda övriga komponenterna utvärderas då inte. Hänför sig <CorVal> resp. en komponent från <CorVal> till planaxeln, så värderas uppgiften som radiemått. Det betyder att ett verktyg t.ex. blir "längre" med det angivna måttet, vilket leder till en motsvarande dubbelt så stor ändring i arbetsstycksdiametern. Måttuppgifterna hänför sig till grundsystemet för styrningen (inch resp. metriskt).	
		Datatyp:	REAL[3]

2	<Comp>:	Verkygskomponent(er)			
		Datatyp:		STRING	
		Teckenföljden består av två delstringar, som är skilda av en dubbelpunkt från varandra.			
		Allm. form: "<SubStr_1> [: <SubStr_2>]"			
		<SubStr_1>:	Den första delstringen måste alltid finnas och kan bestå antingen av en eller två tecken. Därvid står det första eller enda tecknet för den 1:a komponenten (Val ₁) och det andra tecknet för den 2:a komponenten (Val ₂), som bearbetas i enlighet med de följande parametrarna <CorComp> och <CorMode>.		
			Tecken:	C	Adaptermått eller basmått (det av de båda alternativt existerande komponenterna, som är aktivt för det använda verktyget)
				E	Inställningskompenseringar
				G	Geometri
				S	Summakompenseringar
				W	Förslitning
<SubStr_2>:	Den andra delstringen är optional. Den kan alternativt bestå av den (enda) bokstaven "W" eller "T".				
	Tecken:	W	Är den andra delstringen tom eller innehåller den bokstaven "W", beräknas kompenseringsvärdena så som om de har mätts i arbetsstyckskoordinatsystemet (WKS).		
		T	Innehåller den andra delstringen bokstaven "T", beräknas kompenseringsvärden som om de har mätts i verktygskoordinatsystemet (Tool Coordinate System, TCS).		
Skrivsättet för tecknen i stringen (stor eller liten) är valfritt. Valfritt många mellanlag eller tabulatorstecken (white spaces) kan infogas.					

3	<CorComp>:	Specificerar komponente(n)(erna) i det verktygsdatablock, som ska beskrivas (option)		
		Datatyp:	INT	
		Värde:	0	Kompenseringsvärdet <CorVal>[0] hänför sig till den i parametern <GeoAx> överlämnade geometriaxeln i arbetsstyckskoordinatsystemet eller i verktygskoordinatsystemet (se för detta beskrivningen för parametern <Comp>). Dvs. kompenseringssvärdet måste i de betecknade verktygskomponenterna räknas in så, att under hänsynstagande till alla parametrar, som kan påverka verktygslängdsberäkningen, en ändring av den totala verktygslängden i den angivna axelriktningen med det föreskrivna värdet resulterar. Denna ändring ska uppnås genom kompenseringsvärdet av de under <Comp> angivna komponenterna och den under <CorMode> (se följande parameter) angivna symboliska räkneföreskriften. Den resulterande kompenseringsvärdet kan därför ha inverkan i alla tre riktningarna.
		1	Om "0" dock vektoriell. Innehållet i vektorn <CorVal> hänför sig till abskissa, ordinata och applikata i arbetsstyckskoordinatsystemet eller verktygskoordinatsystemet (se därtill beskrivningen för parametern <Comp>). Den efterföljande parametern <GeoAx> utvärderas inte.	
		2	Vektoriell kompenseringsvärdet, dvs. L1, L2 och L3 kan förändras simultant. I motsats till varianterna "0" och "1" hänför sig de i <CorVal> ingående kompenseringssvärdena dock till koordinaterna för Val ₁ -komponenten (se följande parameter <CorMode>) för verktyget. En eventuellt förekommande snedställning av verktyget gentemot arbetsstyckskoordinatsystemet har inget inflytande på kompenseringsvärdet.	
		3 - 5	Kompenseringsvärdet av verktygslängden L1 till L3 (\$TC_DP3 till \$TC_DP5) resp. de motsvarande värdena vid förslitning, inställnings- eller summakompenseringsvärdet. Kompenseringsvärdet ingår i <CorVal>[0]. Det har mätts i koordinaterna till Val ₁ -komponenten (se följande parameter <CorMode>) för verktyget. En eventuellt förekommande snedställning av verktyget gentemot arbetsstyckskoordinatsystemet har inget inflytande på kompenseringsvärdet.	
		6	Kompenseringsvärdet av verktygsradien (\$TC_DP6) resp. de motsvarande värdena vid förslitning, inställnings- eller summakompenseringsvärdet. Det tas hänsyn till bits 10 och 11 (värdering av diameter- resp. diameterförslitningsuppgift valfritt som radie- eller diameteruppgift) i maskindatum MD20360 \$MC_TOOL_PARAMETER_DEF_MASK.	
		7 – 11	Kompenseringsvärdet av \$TC_DP7 till \$TC_DP11 resp. de motsvarande värdena vid förslitning, inställnings- eller summakompenseringsvärdet. Dessa parametrar behandlas som verktygsradien.	
		Är denna parameter inte angiven, är dess värde "0".		

4	<CorMode>:	Specificerar typen av den skrivoperation som ska utföras (option)	
		Datatyp:	INT
		Värde:	0 $Val_{1ny} = <CorVal>$
			1 $Val_{1ny} = Val_{1gam} + <CorVal>$
			2 $Val_{1ny} = <CorVal>$ $Val_{2ny} = 0$
5	<GeoAx>:		3 $Val_{1ny} = Val_{1gam} + Val_{2gam} + <CorVal>$ $Val_{2ny} = 0$
		Skrivsättet $Val_{1gam} + Val_{2gam}$ ska förstås symboliskt. Värderas de båda komponenterna (på grund av tillståndet för <Stat>) olika, dvs. är en vridning verksam mellan de båda komponenterna, transformeras Val_{2gam} så före additionen, att den resulterande verktygslängden efter raderingen av Val_{2ny} och före additionen av <CorVal> förblir oförändrad.	
		<CorVal> hänför sig alltid till Val_1 . <CorVal> är ett värde, som beroende av den andra delen av parametern <Comp> mäts i arbetsstyckskoordinatsystemet (WKS) eller i verktygskoordinatsystemet (TCS). Den är därför gentemot den verktygskomponent, i vilken den ska inräknas, eventuellt redan transformerad. Den kan alltså inte genast inberäknas med det sparade värdet, utan måste före additionen till Val_1 resp. Val_2 återtransformeras. Detta kan ha till följd att kompenseringen verkar på en annan än den genom <CorComp> definierade axeln, resp. att denna verkar på flera axlar.	
		För den händelse <CorComp> = 0, dvs. om <CorVal> inte innehåller någon vektor, utan endast ett enskilt värde, utförs de beskrivna operationerna i de koordinater, i vilka <CorVal> mättes (WKS / TCS). Speciellt gäller detta också för nollsättning av Val_{2ny} i varianterna 2 och 3. Detta resultat återtransformeras sedan till koordinaterna för verktyget. Det kan ha till följd att inget av koordinatvärdena som ska nollställas (L_1, L_2, L_3) blir noll resp. att koordinatvärdena, som tidigare var noll, nu är ej lika med noll. Utförs de motsvarande operationerna dock successivt för alla tre geometriaxlarna, är garanterat att också alla tre koordinatvärdena för de komponenter som ska raderas blir noll. Är verktyget inte vridet gentemot arbetsstyckskoordinatsystemet, eller så vridet att alla verktygskomponenter förblir parallella till koordinataxlarna (axelbyten), är garanterat att varje gång endast en verktygskoordinat ändrar sig.	
		Det successiva utförande av samma operation (<CorMode>) med <CorComp> = 0 för alla tre koordinataxlarna i valfri ordningsföljd är identiskt med utförandet en gång av samma operation med <CorComp>=1.	
5	<GeoAx>:	För parametervärdena "0" och "1" måste parametern <Comp> innehålla ett och för parametervärdena "2" och "3" två tecken.	
		Exempel:	
		<Comp> innehåller stringen "ES", <CorMode> värdet "2"	
		$\Rightarrow$ Inställningskompensering _{ny} = <CorVal>, summakompensering _{ny} = 0	
		Är parametern <CorMode> inte angiven, är dess värde "0".	
5	<GeoAx>:	Anger index för den geometriaxel, i vilken kompenseringensvärde <CorVal>[0] mäts (option)	
		Datatyp:	INT
		Värdeområde:	0 ... 2
		Indexen 0 till 2 hänför sig till abskissa, ordinata och applikata i det verksamma planet (G17/G18/G19) för den aktuella verktygsomgivningen.	
		Innehållet i denna parameter utvärderas endast, när parametern <CorComp> har värdet "0".	

6	<Stat>:	Namn för datablocket till beskrivning av en verktygsomgivning (option)	
		Datatyp:	STRING
		Är värdet för denna parameter nollstringen (""), eller anges den inte, används det aktuella tillståndet. Är inget verktyg specificerat, används det aktuella verktyget.	
7	<T>:	Internt T-nummer för verktyget (option)	
		Datatyp:	INT
		Är denna parameter inte angiven eller är dess värde "0", används det i <Stat> lagrade verktyget. Är värdet för denna parameter "-1", använd T-numret för det aktiva verktyget. Det är också tillåtet, att explicit ange numret för det aktiva verktyget. Observera: Är <Stat> inte angivet, används det aktuella tillståndet som verktygsomgivning. Eftersom det med <T> = 0 hänvisas till det i verktygsomgivningen sparade T-numret, används i denna det aktiva verktyget, dvs. uppgifterna <T> = 0 och <T> = -1 betyder i detta specialfall detsamma.	
8	<D>:	Skärnummer för verktyget (option)	
		Datatyp:	INT
		Är denna parameter inte angiven eller är dess värde "0", rättar sig den använda D-numret efter källan till T-numret. Används T-numret från verktygsomgivningen, läses också D-numret från verktygsomgivningen, i annat fall det från det aktuellt aktiva verktyget.	
9	<TL>:	Nummer för den platsberoende kompenseringen (option)	
		Datatyp:	INT
		Är denna parameter inte angiven, rättar sig det använda DL-numret efter källan för T-numret. Används T-numret från verktygsomgivningen, läses också D-numret från verktygsomgivningen, i annat fall det från det aktuellt aktiva verktyget. Specificerar T, D och DL ett verktyg utan platsberoende kompenseringar, får i parametern <Comp> ingen summa- eller inställningskompensering specificeras (felkod i <Status>).	

Märk

Inte alla möjliga kombinationer av de tre parametrarna <Comp>, <CorComp> och <CorMode> är meningsfulla. Exempelvis kräver räknepöreskriften 3 i <CorComp> uppgiften för två tecken i <Comp>. Anges en otillåten parameterkombination, returneras i <Status> en motsvarande felkod.

Exempel

Exempel 1

Programkod	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräsverktyg
N30 \$TC_DP3[1,1]=10.0	; Geometri L1
N40 \$TC_DP12[1,1]=1.0	; Förslitning L1
N50 _CORVAL[0]=0.333	
N60 T1 D1 G17 G0	

Programkod	Kommentar
N70 R1=SETTCOR(_CORVAL,"G",0,0,2)	
N80 T1 D1 X0 Y0 Z0	; ==> MKS-position X0.000 Y0.000 Z1.333
N90 M30	

<CorComp> är "0", därför ersätts det i Z-riktningen verkande koordinatvärdet för geometri-komponenten med kompenseringsvärdet 0,333.

Den resulterande totala verktygslängden är därmed: $L1 = 0,333 + 1,000 = 1,333$

Exempel 2

Programkod	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräsverktyg
N30 \$TC_DP3[1,1]=10.0	; Geometri L1
N40 \$TC_DP12[1,1]=1.0	; Förslitning 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	; ==> MKS-position X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> är "1", därför adderas det i Z-riktningen verkande kompenseringsvärdet på 0,333 till förslitningsvärdet på 1,0.

Den resulterande totala verktygslängden är därmed: $L1 = 10,0 + 1,333 = 11,333$

Exempel 3

Programkod	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräsverktyg
N30 \$TC_DP3[1,1]=10.0	; Geometri L1
N40 \$TC_DP12[1,1]=1.0	; Förslitning 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	; ==> MKS-position X0.000 Y0.000 Z0.333
N90 M30	

<CorComp> är "2", därför införs den i Z-riktningen verkande kompenseringen i geometri-komponenten (det gamla värdet skrivs över), och förslutningsvärdet raderas.

Den resulterande totala verktygslängden är därmed: $L1 = 0,333 + 0,0 = 0,333$

Exempel 4

Programkod	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräsverktyg
N30 \$TC_DP3[1,1]=10.0	; Geometri L1

Programkod	Kommentar
N40 \$TC_DP12[1,1]=1.0	; Förslitning 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	; ==> MKS-position X0.000 Y0.000 Z11.333
N90 M30	

<CorComp> är "3", därför adderas förslitningsvärde och kompenseringensvärde till geometri-komponenten och förslitningskomponenten raderas.

Den resulterande totala verktygslängden är därmed: $L1 = 11,333 + 0,0 = 11,333$.

Exempel 5

Programkod	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=120	; Fräsverktyg
N30 \$TC_DP3[1,1]=10.0	; Geometri L1
N40 \$TC_DP12[1,1]=1.0	; Förslitning 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	; ==> MKS-position X0.333 Y0.000 Z11.000
N90 M30	

<CorComp> är "3" som i det tidigare exemplet, kompenseringen verkar dock nu på geometriaxeln med index "0" (X-axel), som på grund av G17 är tillordnat verktygskomponenten L3 för ett fräsverktyg. Anropet av SETTCOR påverkar därför inte verktygsparameter \$TC_DP3 och \$TC_DP12. Kompenseringensvärdet förs i stället in i \$TC_DP5.

Exempel 6

Programkod	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=500	; Svarvverktyg
N30 \$TC_DP3[1,1]=10.0	; Geometri L1
N40 \$TC_DP4[1,1]=15.0	; Geometri L2
N50 \$TC_DP12[1,1]=10.0	; Förslitning L1
N60 \$TC_DP13[1,1]=0.0	; Förslitning L2
N70 _CORVAL[0]=5.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=SETTCOR(_CORVAL,"GW",0,3,1)	
N110 T1 D1 X0 Y0 Z0	; ==> MKS-position X24.330 Y0.000 Z17.500
N120 M30	

Verktynet är ett svarvverktyg. I block N80 aktiveras en framevridning, så att baskoordinatsystemet (BKS) vrids gentemot arbetsstyckskoordinatsystemet (WKS). Kompenseringensvärdet (N70) verkar i WKS på geometriaxeln med index "1", dvs. eftersom G18

är aktivt, på X-axeln. Eftersom $\langle \text{CorMode} \rangle = 3$ gäller, måste verktygsförslitningen i X-axelns riktning i WKS bli noll efter utförandet av N100.

Innehållet i den relevanta verktygsparametern är därför i programslutet:

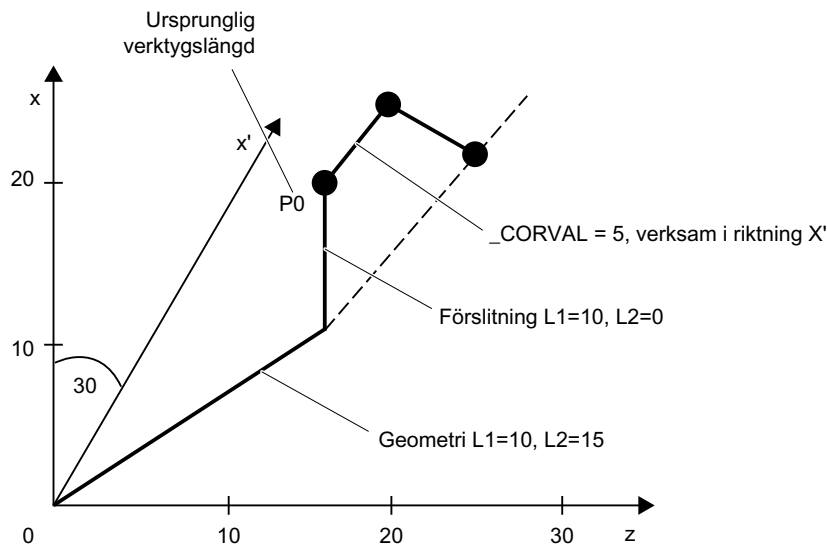
\$TC_DP3[1,1] : 21.830 ; Geometri L1

\$TC_DP4[1,1] : 21.830 ; Geometri L2

\$TC_DP12[1,1] : 2.500 ; Förslitning L1

\$TC_DP13[1,1] : -4.330 ; Förslitning L2

De geometriska förhållandena gestaltas i den följande bilden. Den totala förslitningen inklusive $_CORVAL$ projiceras på X' -riktningen i WKS. Det ger punkten P2. Koordinaterna för denna punkt (mätt i X-Y-koordinater) införs i verktygets geometrikomponent. I förslitningen blir differensvektorn $P_2 - P_1$ över. Därmed har förslitningen inte längre någon komponent i riktningen för $_CORVAL$.



Fortsätter man exempelprogrammet enligt N110 med följande anvisningar, övertas den resterande förslitningen fullständigt i geometrin, eftersom kompenseringen nu verkar i Z' -axeln (parameter $\langle \text{GeoAx} \rangle = 0$):

```
N120 _CORVAL[0]=0.0
N130 R1=SETTCOR(_CORVAL,"GW",0,3,0)
N140 T1 D1 X0 Y0 Z0 ; ==> MKS-Position X24.330 Y0.000 Z17.500
```

Eftersom det nya kompenseringsvärdet är "0", får den totala verktygslängden och därmed också den i N140 uppsökta positionen inte förändras. Skulle $_CORVAL$ i N120 vara skilt från "0", skulle en ny total verktygslängd och därmed också en förändrad position i N140 resultera, förslitningsandelen i verktygslängden skull dock i varje fall vara noll, dvs. den totala verktygslängden ingår sedan i varje fall i verktygets geometriandel.

Samma resultat som med två gånger anrop av funktionen SETTCOR med parametern <CorComp> = 0 uppnår man också genom en gång anrop med <CorComp> = 1 (vektoriell kompensering):

Programkod	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=500	; Svarvverktyg
N30 \$TC_DP3[1,1]=10.0	; Geometri L1
N40 \$TC_DP4[1,1]=15.0	; Geometri L2
N50 \$TC_DP12[1,1]=10.0	; Förslitning L1
N60 \$TC_DP13[1,1]=0.0	; Förslitning 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	; ==> MKS-position X24.330 Y0.000 Z17.500
N120 M30	

I detta fall är alla verktygets förslitningskomponenter genast efter det första anropet av SETTCOR i N100 lika med noll.

Exempel 7

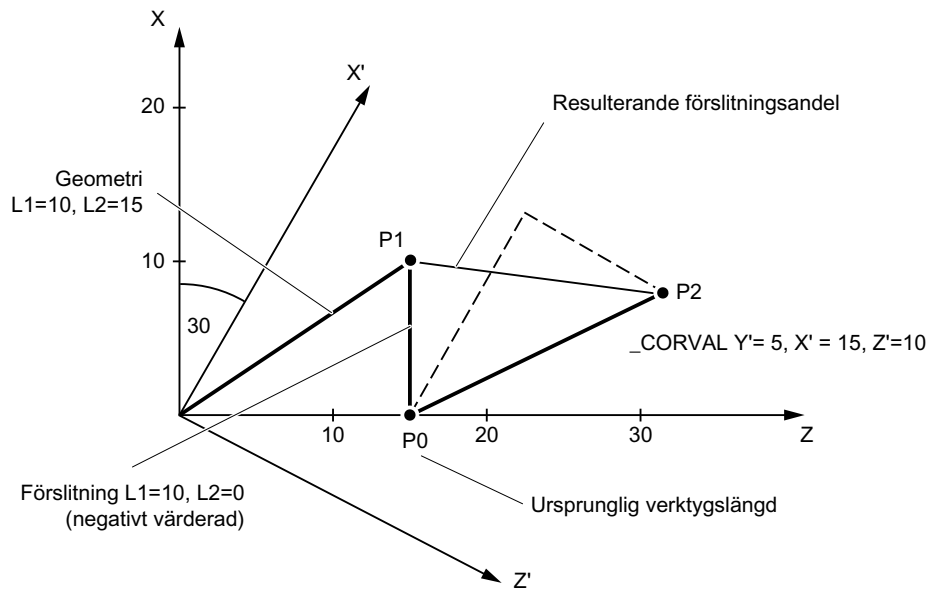
Programkod	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=500	; Svarvverktyg
N30 \$TC_DP3[1,1]=10.0	; Geometri L1
N40 \$TC_DP4[1,1]=15.0	; Geometri L2
N50 \$TC_DP12[1,1]=10.0	; Förslitning L1
N60 \$TC_DP13[1,1]=0.0	; Förslitning L2
N70 _CORVAL[0]=5.0	
N80 ROT Y-30	
N90 T1 D1 G18 G0	
N100 R1=SETTCOR(_CORVAL,"GW",3,3)	
N110 T1 D1 X0 Y0 Z0	; ==> MKS-position X25.000 Y0.000 Z15.000
N120 M30	

Jämfört med exempel 6 är här parametern <CorComp> = 3, därför kan uppgiften för parametern <GeoAx> utgå. Det i _CORVAL[0] ingående värdet verkar nu omedelbart på verktygskomponenten L1, vridningen i N80 har inget inflytande på resultatet, förslitningskomponenterna i \$TC_DP12 övertas tillsammans med _CORVAL[0] i geometriandelen, så att på grund av \$TC_DP13 den totala verktygsländan redan efter första anropet av SETTCOR i N100 står i verktygets geometridel.

Exempel 8

Programkod	Kommentar
N10 DEF REAL _CORVAL[3]	
N20 \$TC_DP1[1,1]=500	; Svarvverktyg
N30 \$TC_DP3[1,1]=10.0	; Geometri L1
N40 \$TC_DP4[1,1]=15.0	; Geometri L2
N50 \$TC_DP5[1,1]=20.0	; Geometri L3
N60 \$TC_DP12[1,1]=10.0	; Förslitning L1
N70 \$TC_DP13[1,1]=0.0	; Förslitning L2
N80 \$TC_DP14[1,1]=0.0	; Förslitning 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	; ==> MKS-position X7.990 Y25.000 Z31.160
N170 M30	

I N90 har settingdatum SD42930 \$SC_WEAR_SIGN satts, dvs. förslitningen måste värderas med negativt förtecken. Kompenseringen är vektoriell (<CorComp> = 1), och kompenseringsvektorn måste adderas till förslitningen (<CorMode> = 1). De geometriska förhållandena i Z-X-planet gestaltas i den följande bilden.



På grund av <CorMode> = 1 förblir geometriandelen för verktyget oförändrad. Det i WKS (rotation runt y-axeln) definierade kompenseringsvärdet måste övertas så i förslitningsandelen att den totala verktygslängden i bild 3 hänvisar till punkt P₂. Den resulterande förslitningsandelen för verktyget är därför given genom avståndet mellan de båda punkterna

P_1 och P_2 .

Men eftersom förslitningen på grund av settingdatumet SD42930 \$SC_WEAR_SIGN värderas negativt, måste den så fastställda kompenseringen föras in med negativt förtecken i kompenseringsminnet. Innehållet i den relevanta verktygsparametern är i programslutet därför:

\$TC_DP3[1,1] : 10.000 ; Geometri L1 (oförändrad)

\$TC_DP4[1,1] : 15.000 ; Geometri L2 (oförändrad)

\$TC_DP5[1,1] : 10.000 ; Geometri L3 (oförändrad)

\$TC_DP12[1,1] : 2.010 ; Förslitning L1 ($= 10 - 15 * \cos(30) + 10 * \sin(30)$)

\$TC_DP13[1,1] : -16.160 ; Förslitning L2 ($= -15 * \sin(30) - 10 * \cos(30)$)

\$TC_DP14[1,1] : -5.000 ; Förslitning L3

På L3-komponenten i Y-riktningen låter sig verkan av settingdatumet SD42930

\$SC_WEAR_SIGN identifieras utan den ytterligare kompliceringen genom framevridningen.

Ytterligare informationer

Svarv- / slipverktyg: Beräkning av verktygslängden i beroende av MD20360 \$MC_TOOL_PARAMETER_DEF_MASK

Hur förslitningen resp. verktygslängden för svarv- / slipverktyg ska värderas i en eventuellt förefintlig diameteraxel, fastläggs i maskindatumet:

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK.<Bit> = <Wert>

<Bit>	<Wert>	Betydelse
0	0	För svarv- / slipverktyg inberäknas förslitningsparametern för planaxeln i radievärdet :
	1	För svarv- / slipverktyg inberäknas förslitningsparametern för planaxeln som diametervärde :
1	0	För svarv- / slipverktyg inberäknas verktygslängskomponenten för planaxeln som radievärde :
	1	För svarv- / slipverktyg inberäknas verktygslängskomponenten för planaxeln som diametervärde :

Har de ifrågavarande bits ställts in, värderas den tillhörande posten med faktorn 0,5.

Kompenseringen med hjälp av SETTCOR utförs så att den totala effektiva verktygslängdändringen är lika med det till <CorVal> överlämnade värdetvärde. Värderas en längd på grund av maskindatumet MD20360 \$MC_TOOL_PARAMETER_DEF_MASK vid längdberäkningen med faktorn 0,5 måste kompenseringen för denna komponent därför göras med det dubbla överförda värdet.

Exempel

MD20360 \$MC_TOOL_PARAMETER_DEF_MASK = 2 (verktygslängd måste utvärderas i diameteraxeln med faktorn 0,5)

Axel X är diameteraxeln.

Programkod	Kommentar
N10 DEF REAL _LEN[11]	
N20 DEF REAL _CORVAL[3]	

Programkod	Kommentar
N30 \$TC_DP1[1,1]=500	; Verkygstyp
N40 \$TC_DP2[1,1]=2	; Skärläge
N50 \$TC_DP3[1,1]=3.	; Geometri - längd 1
N60 \$TC_DP4[1,1]=4.	; Geometri - längd 2
N70 \$TC_DP5[1,1]=5.	; Geometri - längd 3
N80 _CORVAL[0]=1.	
N90 _CORVAL[1]=1.	
N100 _CORVAL[2]=1.	
N110 T1 D1 G18 G0 X0 Y0 Z0	; ==> MKS-position X1.5 Y5 Z4
N120 R1=SETTCOR(_CORVAL,"G",1,1)	
N130 T1 D1 X0 Y0 Z0	; ==> MKS-position 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	

Kompenseringen av verktygslängden ska i varje axel vara 1 mm (N80 till N100). I längderna L2 L3 adderas därför till originallängden 1 mm för var och en. Till originalverktygslängden L1 adderas däremot det dubbla kompenseringsvärdet (2 mm), så att den totala verktygslängden ändrar sig enligt begäran med 1 mm. I jämförelse med de uppsökta positionerna i blocken N110 och N130 ser man att varje axelposition har förändrat sig med 1 mm.

11.13 Läs tillordningen av verktygslängderna L1, L2, L3 till koordinataxlarna (LENTOAX)

Funktionen LENTOAX levererar informationer över tillordningen av verktygslängderna L1, L2 och L3 för det **aktiva** verktyget till abskissa, ordinata och applikata. Tillordningen av abskissa, ordinata och applikata till geometriaxlarna påverkas av frames och det aktiva planet (G17 - G19).

Därvid betraktas endast geometriandelen för ett verktyg (\$TC_DP3[<t>,<d>] till \$TC_DP5[<t>,<d>]), dvs. en eventuellt därifrån avvikande axeltillordning för ytterligare komponenter (t. ex. förslitning) har inget inflytande på resultatet.

Syntax

```
<Status> = LENTOAX(<AxInd>, <Matrix>[, <Coord>])
```

Princip

LENTOAX(...):		Fördefinierad funktion för läsning av tillordningen av verktygslängderna L1, L2 och L3 för det aktiva verktyget till koordinataxlarna	
		Ensam i blocket:	ja
<Status>:		Funktionens returvärde. Negativa värden indikerar feltilstånd.	
		Datatyp:	INT
		Värde:	0 Funktion OK Information i <AxInd> räcker till för beskrivning (alla verktygslängdkomponenterna är parallella till geometriaxlarna).
			1 Funktion är OK, men till en korrekt beskrivning måste innehållet i <Matrix> utvärderas (verktygslängdkomponenterna är inte parallella till geometriaxlarna).
			-1 Ogiltig string i parametern <Coord>.
			-2 Inget verktyg aktivt.
Parameter			
1	<AxInd>:	Är verktygslängdkomponenterna parallella till geometriaxlarna, returneras axelindexen, som är tillordnade längdkomponenterna L1 till L3, till fältet <AxInd>:	
		• <AxInd> [0]: Abskissa • <AxInd> [1]: Ordinata • <AxInd> [2]: Applikata	
		Datatyp:	INT[3]
		Värde:	0 Ingen tillordning finns (axel existerar inte)
			1 ... 3 Nummer för den längd, som verkar i den motsvarande koordinataxeln.
			-1 ... -3 Förtecknet är negativt, när verktygslängdkomponenten pekar i negativ koordinatriktning.
		Är inte alla längdkomponenter parallella resp. antiparallella till geometriaxlarna, returneras alltid index för den axel till <AxInd> som innehåller den största andelen av en verktygslängdkomponent. I detta fall (om inte funktionen av ett annat skäl levererar ett fel) har returvärdet <Status> = 1. Avbildningen av verktygslängdkomponenterna L1 till L3 på geometriaxlarna 1 till 3 beskrivs sedan fullständigt genom innehållet i den 2:a parameters <Matrix>.	

11.13 Läs tillordningen av verktygslängderna L1, L2, L3 till koordinataxlarna (LENTOAX)

2	<Matrix>:	Matris, som avbildar vektorn för verktygslängderna (L1=1, L2=1, L3=1) i vektorn för koordinataxlarna (abskissa, ordinata, applikata), dvs. spalterna är tillordnade verktygslängdkomponenterna i ordningsföljden L1, L2, L3, raderna axlarna i ordningsföljden abskissa, ordinata, applikata.	
		Datotyp:	REAL
		I matrisen är alltid alla element giltiga, även om den till en koordinataxel hörande geometriaxeln inte finns, dvs. när den tillhörande posten i <AxInd> är noll.	
3	<Coord>:	Koordinatsystem, för vilket tillordningen gäller (option)	
		Datotyp:	STRING
		Tecken:	MCS M
			Avbildning av verktygslängden i maskinkoordinatsystemet
			BCS B
			Avbildning av verktygslängden i baskoordinatsystemet
			WCS W
			Avbildning av verktygslängden i arbetsstyckskoordinatsystemet (default)
		TCS T	KCS K
			Avbildning av verktygslängden i verktygskoordinatsystemet för den kinematiska transformationen
		TCS T	Avbildning av verktygslängden i verktygskoordinatsystemet
			Skrivsättet för tecknen i stringen (stor eller liten) är valfritt.
		Ange inte parametern <Coord> används WKS (default).	

Märk

I TCS är alltid alla verktygslängdkomponenter parallella eller antiparallella till axlarna.

Antiparallella kan komponenterna endast vara när spegling är aktiv och det följande settingdatumet har satts:

SD42900 \$SC_MIRROR_TOOL_LENGTH (byte av förtecken verktygslängd vid spegling)

Exempel

Standardfall fräsverktyg vid G17.

L1 verkar i Z (applikata), L2 verkar i Y (ordinata), L3 verkar i X (abskissa).

Anrop av funktionerna i formen:

```
<Status>=LENTOAX (<AxInd>, <Matrix>, "WCS")
```

Resultatparametern <AxInd> innehåller sedan värdena:

```
<AxInd>[0] = 3
```

```
<AxInd>[1] = 2
```

```
<AxInd>[2] = 1
```

eller kort: (3, 2, 1)

Den tillhörande matrisen (<Matrix>) är i detta fall:

11.13 Läs tillordningen av verktyglängderna L1, L2, L3 till koordinataxlarna (LENTOAX)

$$\langle \text{Matrix} \rangle = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

Ett byte från G17 till G18 eller G19 ändrar inget i resultatet, eftersom tillordningen av längdkomponenterna till geometriaxlarna ändrar sig på samma sätt som tillordningen av abskissa, ordinata och applikata.

Nu programmeras vid aktiv G17 en framevridning runt Z på 60 grader, t. ex.:

ROT Z60

Riktningen för applikatan (Z-riktning) förblir oförändrad, huvudandelen av L2 ligger nu i riktningen för den nya X-axeln, huvudandelen av L1 i riktningen för den negativa Y-axeln. Returvärdet (<Status>) är därför "1", <AxInd> innehåller värdena (2, -3, 1).

Den tillhörande matrisen (<Matrix>) är i detta fall:

$$\langle \text{Matrix} \rangle = \begin{pmatrix} 0 & \sin 60^\circ & \cos 60^\circ \\ 0 & \cos 60^\circ & -\sin 60^\circ \\ 1 & 0 & 0 \end{pmatrix}$$

Banbeteende

12.1 Tangentialstyrning

12.1.1 Definiera koppling (TANG)

Via den fördefinierade proceduren TANG(...) definieras en tangentialkoppling mellan en roterande axel som följdaxel och två geometriaxlar som styraxlar. Följdaxeln rikas därvid kontinuerligt upp mot bantangenten till styraxlarna.

Märk

Kopplingsfaktor

En kopplingsfaktor på 1 måste inte explicit programmeras.
Genom kopplingsfaktorn -1 vrids riktningen för tangentialaxeln.

Syntax

TANG(<Folgeachse>, <Leitachse_1>, <Leitachse_2>, <Koppelfaktor>, <Koordinatensystem>, <Optimierung>)

Betydelse

TANG (...):	Definiering av en tangentialkoppling		
<Folgeachse>:	Axelnamn för följdaxeln (roterande axel)		
	Datatyp:	AXIS	
	Värdeområde:	Kanalaxelnamn	
<Leitachse_1>	Axelnamn för styraxlarna (geometriaxlar) ¹⁾		
<Leitachse_2>:	Datatyp:	AXIS	
	Värdeområde:	Geometriaxelnamn för kanalen.	
<Koppelfaktor>:	Faktor n för vinkeländringen hos följdaxeln för ändringa av bantangenten för styraxlarna: Vinkeländring _{Följdaxel} = Vinkeländring _{Bantangent} * n		
	Datatyp:	REAL	
	Defaultvärde:	1,0	
<Koordinatensystem>:	Verksamt koordinatsystem ²⁾		
	Datatyp:	CHAR	
	Värde:	"B":	Baskoordinatsystem (defaultvärde)
		"W":	Arbetsstyckskoordinatsystem (inte disponibelt)

<Optimierung>:	Optimeringstyp	
	Datatyp:	CHAR
	Värde:	"S": Standard (defaultvärde) Dynamiken hos den roterande axeln har inga återverkningar på styraxlarna. Är dynamiken för den roterande axeln högre än nödvändigt för medföljningen, är detta förfarande tillräckligt noggrant. Är dynamiken för den roterande axeln inte hög nog, för att följa ändringen av bantangenten, avviker uppriktningen av den roterande axeln längs en inte definierad överläpningsväg från böruppriktningen. "P": Det tas hänsyn till dynamiken för den roterande axeln vid banplaneringen för styraxlarna. För detta måste vid tillkopplingen av tangentialkopplingen med TANGON() två ytterligare parametrar anges: • Översläpningsväg • Vinkeltolerans Siehe Kapitel "Koppla till koppling (TANGON) (Sida 499)" Observera I samband med kinematiska transformationer rekommenderas att använda optimeringstyp "P".
Observera Defaultvärden måste inte programmeras explicit. ¹⁾ Observera Som styraxlar till tangentialkopplingen måste de geometriaxlar användas, som relaterade till maskinens grundposition, kör den programmerade banan i maskinkoordinatsystemet (MKS). Används t. ex. på en fräsmaskin med svänghuvud svängcykeln CYCLE800, görs, beroende på cykelns konfiguration, interpoleringen i WKS t. ex. med geometriaxlarna X och Y. Men tangentialkopplingen måste definieras med geometriaxlarna som styraxlar, som kör den programmerade banan i MKS. För detta ska geometriaxlarna användas som styraxlar när maskinen befinner sig i icke svängt tillstånd. ²⁾ Observera Baskoordinatsystem (BKS) får inte vara vridet i förhållande till MKS. Vrids t. ex. BKS med kommandot ROT eller via svängcykeln CYCLE800, arbetar tangentialstyrningen inte längre korrekt.		

12.1.2 Koppla till mellanblocksgenerering (TLIFT)

Överskrider tangentändringen hos följdaxeln på ett ställe på styraxlarnas programmerade banan det i maskindatum MD37400 \$MA_EPS_TLIFT_TANG_STEP parameterade gränsvärdet, sker den fortsatta banplaneringen beroende på det inställda hörnbeteendet. Utan användning av den fördefinierade proceduren TLIFT(...) körs i enlighet med det i samband med TANG(...) (Sida 497) och TANGON(...) (Sida 499) programmerade överläpningsbeteendet.

Koppla till mellanblocksgenerering

Med programmeringen av TLIFT(...) i anslutning till TANG(...) infogas av förkörningen ett av styrningen automatiskt genererat mellanblock vid identifieringen av ett hörn på detta ställe på banan.

Vid genomarbetningen av programmet stoppas sedan styraxlarna när mellanblocket uppnås. I mellanblocket vrids följdaxeln med maximal axeldynamik i riktningen för bantangenten i det efterföljande blocket. Därefter förflyttas styraxlarna vidare på den programmerade banan.

Koppla från mellanblocksgenerering

För fränkoppling av mellanblocksgenereringen måste tangentialkopplingen definieras på nytt med hjälp av TANG(...), men utan efterföljande tillkoppling av mellanblocksgenereringen med hjälp av TLIFT(...).

Syntax

TLIFT(<Folgeachse>)

Betydelse

TLIFT(...):	Tillkoppling av hörnbeteende med mellanblocksgenerering	
<Folgeachse>:	Axelnamn för följdaxeln (roterande axel)	
	Datatyp:	AXIS
	Värdeområde:	Kanalaxelnamn

Vridhastighet för följdaxeln**Banaxel**

Förflyttades följdaxeln före aktiveringen av tangentialkopplingen redan som banaxel, sker vridrörelsen i mellanblocket som banaxel.

Genom att referensradien föreskrivs med FGREF[<Achse>]=0.001, sker vridrörelsen med den parametrerade maximala axelhastigheten:

MD32000 \$MA_MAX_AX_VELO[<Folgeachse>]

Positioneringsaxel

Förflyttades följdaxeln före aktiveringen av tangentialkopplingen ännu inte som banaxel, sker vridningen i mellanblocket som positioneringsaxel.

Vridrörelsen sker därvid med den parametrerade positioneringshastigheten:

MD32060 \$MA_POS_AX_VELO[<Folgeachse>]

12.1.3 Koppla till koppling (TANGON)

Via den fördefinierade proceduren TANGON(...) kopplas en tidigare med TANG(...) (Sida 497) definierad tangentialkoppling till. Följdaxeln riktas sedan vid den efterföljande förflyttningen av styraxlarna kontinuerligt upp mot bantangenten.

Vinkel för följdaxeln

Den vinkel, som följdaxeln intar vid referens till bantangenten, är beroende av i TANG(...) föreskrivet utväxlingsförhållande, den i maskindatum MD37402 \$MA_TANG_OFFSET parameterade offsetvinkeln och den därtill additivt verkande vid TANGON(...) föreskrivna offsetvinkeln.

Optimering "P"

Angavs vid definitionen av tangentialkopplingen (TANG(...)) som optimeringsparameter värdet "P", måste vid tillkopplingen av kopplingen parametern "översläpningsväg" och som tillval parametern "vinkeltolerans" anges.

Föreskrivs som vinkeltolerans värdet 0 verkar endast parametern "översläpningsväg".

Föreskrivs som vinkeltolerans ett värde större än 0 resulterar den verksamma översläpningsvägen ur minimet för den parameterade översläpningsvägen och översläpningsvägen på grund av den parameterade vinkeltoleransen.

Är dynamiken för följdaxeln inte tillräcklig, för att följa de parameterade villkoren, reduceras banhastigheten för styraxlarna motsvarande.

Syntax

TANGON(<Folgeachse>, <Offsetwinkel>, <Überschleifweg>, <Winkeltoleranz>)

Betydelse

TANGON(...):	Tillkoppling av en tangentialkoppling	
<Folgeachse>:	Axelnamn för följdaxeln (roterande axel)	
	Datotyp:	AXIS
	Värdeområde:	Kanalaxelnamn
<Offsetwinkel>:	Offsetvinkel för följdaxeln till bantangenten Referenspunkt är nollpunkten för den roterande axeln.	
	Datotyp:	REAL
<Überschleifweg>:	Maximalt tillåtet översläpningsväg Skulle översläpningsvägen bli större på grund av dynamikvillkoren, förminskas styraxlarnas banhastighet.	
	Datotyp:	REAL
<Winkeltoleranz>:	Maximalt tillåten tolerans beträffande den förskrivna vinkeln mellan följdaxel-nolläge och bantangenten	
	Datotyp:	REAL

12.1.4 Koppla från koppling (TANGOF)

Via den fördefinierade proceduren TANGOF(...) kopplas en med TANG(...) (Sida 497) definierad och med TANGON(...) (Sida 499) tillkopplad tangentialkoppling från. Följdaxeln rikas sedan inte längre upp mot bantangenten till styraxeln. Kopplingen av följdaxeln till styraxlarna består dock även efter fränkopplingen, vilket t.ex. följande funktioner förhindrar:

- Planväxling
- Geometriaxelomkoppling
- Definition av en ny tangentialkoppling för följdaxeln

Det fullständiga upphävandet av förbindelsen av följdaxeln till styraxlarna sker först efter raderingen av kopplingen med TANGDEL(...) (Sida 501).

Programmering

TANGOF(<Folgeachse>)

Betydelse

TANGOF(...):	Fränkoppling av en tangentialkoppling	
<Folgeachse>:	Axelnamn för följdaxeln (roterande axel)	
	Datotyp:	AXIS
	Värdeområde:	Kanalaxelnamn

12.1.5 Radera koppling (TANGDEL)

En med TANG(...) (Sida 497) definierad tangentialkoppling förblir bestående även efter fränkopplingen av tangentialkopplingen med TANGOF(...) (Sida 501). Den bestående tangentialkopplingen förhindrar sedan fortfarande t.ex. följande funktioner:

- Planväxling
- Geometriaxelomkoppling
- Definition av en ny tangentialkoppling för följdaxeln

Med den fördefinierade proceduren TANGDEL(...) raderas efter fränkopplingen av tangentialkopplingen med TANGOF(...) den bestående tangentialkopplingen.

Syntax

TANGDEL(<Folgeachse>)

Betydelse

TANGDEL(...):	Radera en med TANG() definierad tangentialkoppling	
	Verkan:	Blockvis

12.1 Tangentialstyrning

<Folgeachse>:	Axelnamn för följdaxeln, vars tangentialkoppling ska raderas	
	Datatyp:	AXIS
	Värdeområde:	Kanalaxelnamn

Exempel

Styraxelväxling

Innan en ny tangentialkoppling med en annan styraxel kan definieras för följdaxeln, måste den bestående tangentialkopplingen först raderas.

Programkod	Kommentar
N10 TANG (A, X, Y, 1)	; Definiera tangentialkoppling för följdaxel A: A till X och Y
N20 TANGON(A)	; Koppla till tangentialkoppling för följdaxel A
N30 X10 Y20	
...	
N80 TANGOF(A)	; Koppla från tangentialkoppling för följdaxel A
N90 TANGDEL (A)	; Radera tangentialkoppling för följdaxel A
...	
N120 TANG (A, X, Z)	; Definiera ny tangentialkoppling för följdaxel A
N130 TANGON(A)	; Koppla till ny tangentialkoppling för följdaxel A
...	

Geometriaxelomkoppling

Innan en geometriaxelomkoppling för en bestående koppling kan göras, måste kopplingen först raderas.

Programkod	Kommentar
N10 GEOAX(2, Y1)	; 2:a Geometriaxel = maskinaxel Y1
N20 TANG(A, X, Y)	; Definiera tangentialkoppling för följdaxel A
N30 TANGON(A, 90)	; Koppla till tangentialkoppling för följdaxel A
N40 G2 F8000 X0 Y0 I0 J50	; Rörelseblock
N50 TANGOF(A)	; Koppla från tangentialkoppling för följdaxel A
N60 TANGDEL (A)	; Radera tangentialkoppling för följdaxel A
N70 GEOAX (2, Y2)	; 2:a Geometriaxel = maskinaxel Y2
N80 TANG(A, X, Y)	; Definiera ny tangentialkoppling för följdaxel A
N90 TANGON(A, 90)	; Koppla till ny tangentialkoppling för följdaxel A
...	

12.2 Matningsförlopp (FNORM, FLIN, FCUB, FPO)

För flexiblares föreskrift av matningsförloppet kompletteras matningsprogrammeringen enligt DIN 66025 med linjära och kubiska förlopp.

De kubiska förloppen kan programmeras direkt eller som interpolerande splines. Härigenom låter sig - beroende på böjningen på det arbetsstycke som ska bearbetas - kontinuerligt jämna hastighetsförlopp programmeras.

Dessa hastighetsförlopp möjliggör jerkfria accelerationsändringar och härigenom tillverkning av likformiga arbetsstycksytor.

Syntax

F... FNORM
 F... FLIN
 F... FCUB
 F=FPO (... , ... , ...)

Betydelse

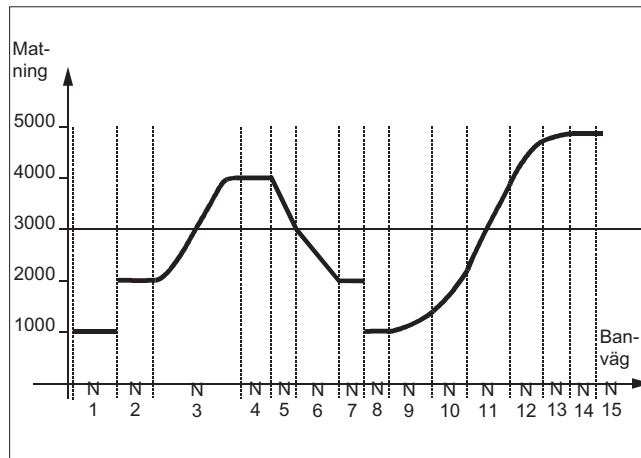
FNORM:	Grundinställning Matningsvärdet föreskrivs via blockets banväg och gäller därefter som modalt värde.
FLIN:	Banhastighetsprofil linjär : Matningsvärdet körs in linjärt från det aktuella värdet i blockbörjan till blockslutet via banvägen och gäller därefter som modalt värde. Detta beteende kan kombineras med G93 och G94.
FCUB:	Banhastighetsprofil kubisk : De blockvis programmerade F-värdena - relaterade till blockslutpunkten - förbinds genom en spline. Splinen börjar och slutar tangentiellt till den föregående resp. efterföljande matningsuppgiften och verkar med G93 och G94. Saknas F-adressen i ett block så används härför det sist programmerade F-värdet.
F=FPO... :	Banhastighetsprofil via polynom : F-adressen betecknar matningsförloppet via ett polynom från det aktuella värdet till blockslutet. Slutvärdet gäller därefter som modalt värde.

Matningsoptimering vid böjda banstycken

Matningspolynom $F=FPO$ och matningsspline $FCUB$ bör alltid köras med konstant skärhastighet CFC . Härigenom låter sig en accelerationskontinuerlig börmatningsprofil skapas.

Exempel: Olika matningsprofiler

I detta exempel finns programmeringen och grafisk framställning av olika matningsprofiler.



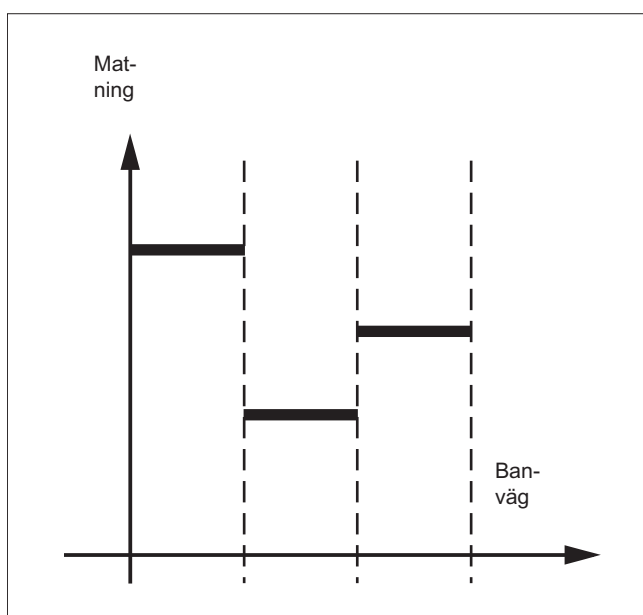
Programkod	Kommentar
N1 F1000 FNORM G1 X8 G91 G64	; Konstant matningsprofil, kedjemåttsuppgift
N2 F2000 X7	; Språngartad börshastighetsändring
N3 F=FPO(4000, 6000, -4000)	; Matningsprofil via polynom med matning 4000 i blockslutet.
N4 X6	; Polynommatning 4000 gäller som modalt värde.
N5 F3000 FLIN X5	; Linjär matningsprofil
N6 F2000 X8	; Linjär matningsprofil
N7 X5	; Linjär matning gäller som modalt värde
N8 F1000 FNORM X5	; Konstant matningsprofil med språngartad accelerationsändring.
N9 F1400 FCUB X8	; Alla följande blockvis programmerade F-värden förbinds med splines.
N10 F2200 X6	
N11 F3900 X7	
N12 F4600 X7	
N13 F4900 X5	; Koppla från splineprofil.
N14 FNORM X5	
N15 X20	

Ytterligare information

FNORM

Matningsadressen F betecknar banmatningen som konstant värde enligt DIN 66025.

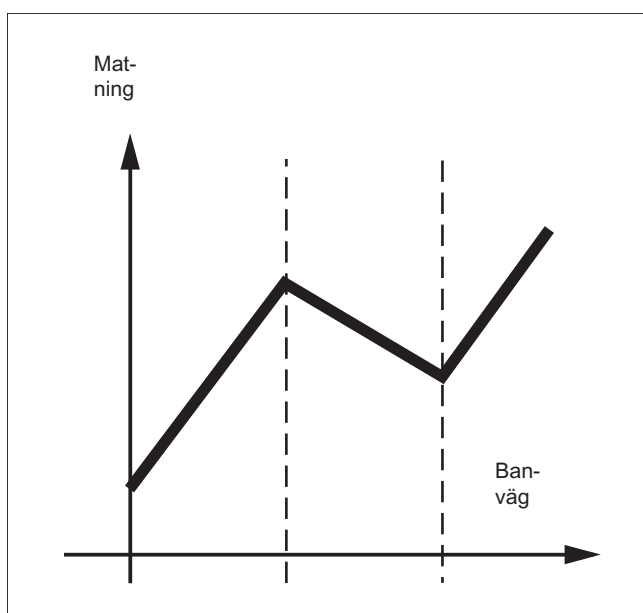
Mer informationer finns i programmeringshandboken "Grunder".

**FLIN**

Matningsförloppet körs in linjärt från det aktuella matningsvärdet till det programmerade F-värdet fram till blockslut.

Exempel:

```
N30 F1400 FLIN X50
```

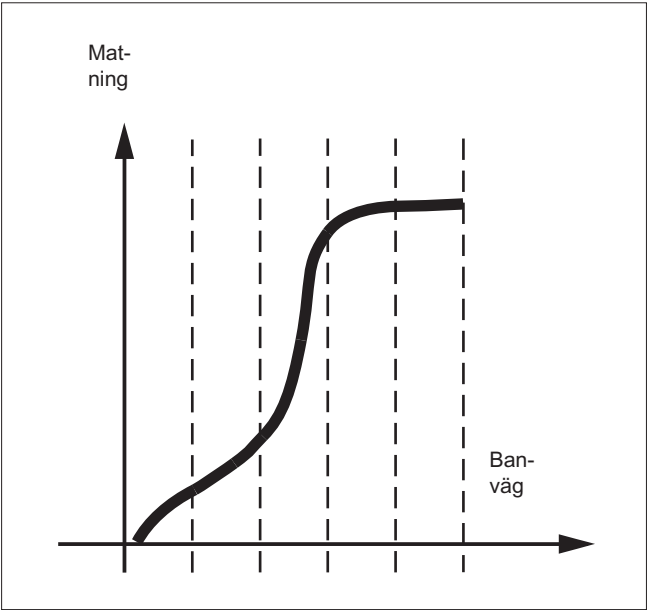
**FCUB**

Matningen körs in i kubiskt förlopp från det aktuella matningsvärdet till det programmerade F-värdet fram till blockslut. Styrningen förbinder alla med aktiv FCUB blockvis programmerade

matningsvärden genom splines. Matningsvärdena tjänar här som stödpunkter för beräkningen av splineinterpoleringen.

Exempel:

```
N50 F1400 FCUB X50
N60 F2000 X47
N70 F3800 X52
```



F=FPO(.....)

Matningsförloppet programmeras direkt via ett polynom. Uppgiften om polynomkoefficienterna sker analogt till polynominterpoleringen.

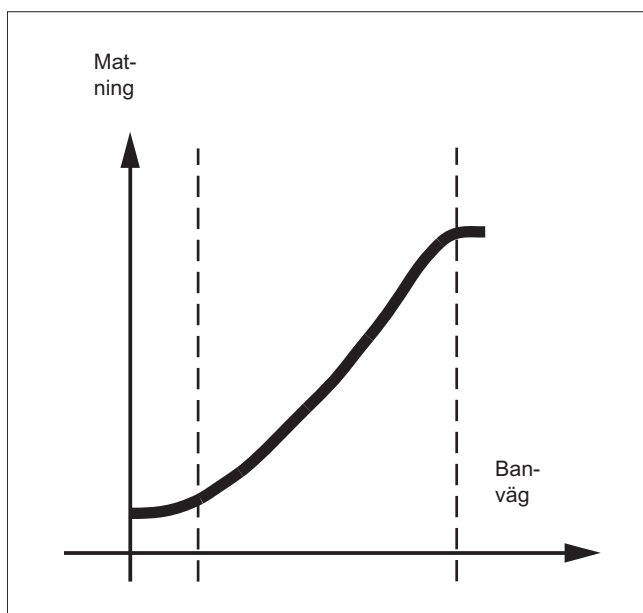
Exempel:

```
F=FPO(endfeed, quadf, cubf)
```

endfeed, quadf och cubf är tidigare definierade variabler.

endfeed:	Matning till blockslut
quadf:	Kvadratisk polynomkoefficient
cubf:	Kubisk polynomkoefficient

Vid aktiv `FCUB` ansluter splinen i blockbörjan och blockslut tangentiellt till det via `FPO` fastlagda förloppet.



Randvillkor

- Oberoende av det programmerade matningsförloppet gäller funktionerna för programmeringen av bankörningsbeteendet.
- Det programmerbara matningsförloppet gäller principiellt absolut - oberoende av G90 eller G91.
- Matningsförloppet FLIN och FCUB verkar med G93 och G94, **inte** vid G95, G96/G961 och G97/G971.
- Vid aktiv kompressor COMPON och sammanfattning av flera block till ett splinesegment gäller följande:

FNORM:	För splinesegmentet gäller F-ordet för det sista tillhörande blocket.
FLIN:	För splinesegmentet gäller F-ordet för det sista tillhörande blocket. Det programmerade F-värdet gäller till slutet av segmentet och uppsöks sedan linjärt.
FCUB:	Den skapade matningssplinen avviker maximalt med det i maskindatum MD20172 \$MC_COMPRESS_VELO_TOL definierade värdet från de programmerade slutpunkterna.
F=FPO (... , ... , ...) :	Dessa block komprimeras inte.

12.3 Accelerationsbeteende

12.3.1 Accelerationsmode (BRISK, BRISKA, SOFT, SOFTA, DRIVE, DRIVEA)

För programmeringen av accelerationsmode står följande detaljprogramkommandon till förfogande:

- "BRISK, BRISKA"
Singelaxlarna resp. banaxlarna kör med maximal acceleration tills de uppnår den programmerade matningshastigheten (**Acceleration utan jerkbegränsning**).
- "SOFT, SOFTA"
Singelaxlarna resp. banaxlarna kör med kontinuerlig acceleration tills de uppnår den programmerade matningshastigheten (**Acceleration med jerkbegränsning**).
- "DRIVE, DRIVEA"
Singelaxlarna resp. banaxlarna kör med maximal acceleration upp till en projekterad hastighetsgräns (MD-inställning). Därefter följer en accelerationsreduktion (MD-inställning!) tills den programmerade matningshastigheten uppnås.

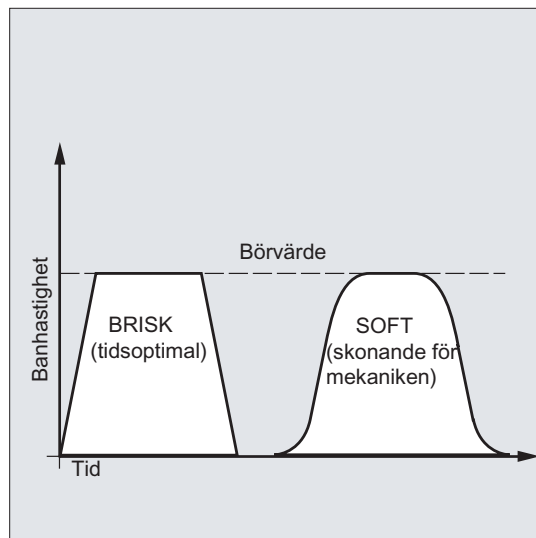


Bild 12-1 Förlöpp för banhastigheten vid BRISK och SOFT

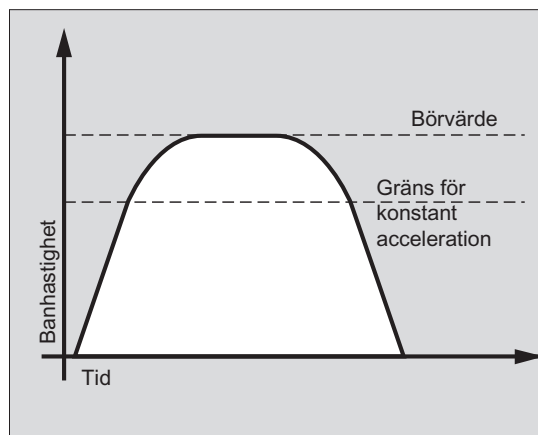


Bild 12-2 Förlopp för banhastigheten vid DRIVE

Syntax

```
BRISK
BRISKA (<Achse1>,<Achse2>,...)
SOFT
SOFTA (<Achse1>,<Achse2>,...)
DRIVE
DRIVEA (<Achse1>,<Achse2>,...)
```

Betydelse

BRISK:	Kommando för tillkoppling av "Acceleration utan jerkbegränsning" för banaxlarna.
BRISKA:	Kommando för tillkoppling av "Acceleration utan jerkbegränsning" för singelaxelrörelser (JOG, JOG/INC, positioneringsaxel, pendelaxel, etc.).
SOFT:	Kommando för tillkoppling av "Acceleration med jerkbegränsning" för banaxlarna.
SOFTA:	Kommando för tillkoppling av "Acceleration med jerkbegränsning" för singelaxelrörelser (JOG, JOG/INC, positioneringsaxel, pendelaxel, etc.).
DRIVE:	Kommando för tillkoppling av reducerad acceleration ovanför en projekterad hastighetsgräns (MD35220 \$MA_ACCEL_REDUCTION_SPEED_POINT) för banaxlarna.
DRIVEA:	Kommando för tillkoppling av den reducerade accelerationen ovanför en projekterad hastighetsgräns (MD35220 \$MA_ACCEL_REDUCTION_SPEED_POINT) för singelaxelrörelser (JOG, JOG/INC, positioneringsaxel, pendelaxel, etc.).
(<Achse1>,<Achse2>,...):	Singelaxlar för vilka det anropade accelerationsmodet ska gälla.

Randvillkor

Byte av accelerationsmode under bearbetning

Om accelerationsmode växlas i ett detaljprogram under bearbetningen (BRISK ↔ SOFT), då görs också vid banstyrningsdrift vid övergången ett blockbyte med precisionsstopp vid blockslut.

Exempel

Exempel 1: SOFT och BRISKA

Programkod
N10 G1 X... Y... F900 SOFT
N20 BRISKA (AX5, AX6)
...

Exempel 2: DRIVE och DRIVEA

Programkod
N05 DRIVE
N10 G1 X... Y... F1000
N20 DRIVEA (AX4, AX6)
...

Litteratur

Funktionshandbok Grundfunktioner; Acceleration (B2)

12.3.2 Påverkan av accelerationen vid följdaxlar (VELOLIMA, ACCLIMA, JERKLIMA)

Vid axelkopplingar (tangentiell medföljning, medsläpning, ledvärdeskoppling, elektronisk växellåda; siehe "Axelkopplingar (Sida 543)") förflyttas följdaxlar/-spindlar beroende på en eller flera styraxlar/-spindlar.

Dynamikbegränsningarna för följdaxlar/-spindlar kan påverkas med funktionerna VELOLIMA, ACCLIMA och JERKLIMA från detaljprogrammet eller från synkronaktioner även vid redan aktiv axelkoppling.

Märk

Funktionen JERKLIMA är inte disponibel för alla kopplingstyper.

Litteratur:

- Funktionshandbok Specialfunktioner; Axelkopplingar (M3)
 - Funktionshandbok Tilläggsfunktioner; Synkronspindel (S3)
-

Märk**Disponibilitet vid SINUMERIK 828D**

Funktionerna VELOLIMA, ACCLIMA och JERKLIMA kan vid SINUMERIK 828D endasst användas i förbindelse med funktionen "Medsläpning"!

Syntax

VELOLIMA (<Achse>)=<Wert>
 ACCLIMA (<Achse>)=<Wert>
 JERKLIMA (<Achse>)=<Wert>

Betydelse

VELOLIMA:	Kommando för korrigering av den parameterade maximalhastigheten
ACCLIMA:	Kommando för korrigering av den parameterade maximalaccelerationen
JERKLIMA:	Kommando för korrigering av den parameterade maximaljerken
<Achse>:	Följdaxel vars dynamikbegränsningar ska korrigeras
<Wert>:	Procentuellt kompenseringssvärde

Exempel**Exempel 1: Korrigering av dynamikbegränsningarna för en följdaxel (AX4)**

Programkod	Kommentar
...	
VELOLIMA[AX4]=75	; Begränsningskorrigering på 75% av den i maskindatum lagrade axiala maximalhastigheten.
ACCLIMA[AX4]=50	; Begränsningskorrigering på 50% av den i maskindatum lagrade axiala maximalaccelerationen.
JERKLIMA[AX4]=50	; Begränsningskorrigering på 50% av den i maskindatum lagrade axiala maximaljerken vid banrörelse.
...	

Exempel 2: Elektronisk växel

Axel 4 kopplas via en koppling "Elektronisk växel" till axel X. Accelerationsförmågan för följdaxeln begränsas till 70 % av den maximala accelerationen. Den maximalt tillåtna hastigheten begränsas till 50 % av den maximala hastigheten. Efter utförd tillkoppling av kopplingen sätts den maximalt tillåtna hastigheten åter på 100 %.

Programkod	Kommentar
...	
N120 ACCLIMA[AX4]=70	; Reducerad maximal acceleration.
N130 VELOLIMA[AX4]=50	; Reducerad maximal hastighet.
...	
N150 EGON(AX4,"FINE",X,1,2)	; Tillkoppling av EG-koppling.

Programkod	Kommentar
...	
N200 VELOLIMA[AX4]=100	; Full maximalhastighet.
...	

Exempel 3: Påverka styrvärdeskoppling per statisk synkronaktion

Axel 4 kopplas med hjälp av ledvärdeskoppling till X. Accelerationsbeteendet begränsas per statisk synkronaktion 2 från och med position 100 till 80 %.

Programkod	Kommentar
...	
N120 IDS=2 WHENEVER \$AA_IM[AX4] >	; Synkronaktion
100 DO ACCLIMA[AX4]=80	
N130 LEADON(AX4, X, 2)	; Styrvärdeskoppling till
...	

12.3.3 Aktivering av teknologispecifika dynamikvärden (DYNNORM, DYNPOS, DYNROUGH, DYNSEMIFIN, DYNFINISH)

Med hjälp av G-gruppen "Teknologi" kan för 5 olika teknologiska bearbetningssteg den därtill passande dynamiken aktiveras.

Dynamikvärden och G-kommandon är projekterbara och därmed beroende av maskindatainställningar (→ maskintillverkare!).

Litteratur:

Funktionshandbok Grundfunktioner; Banskrytningsdrift, precisionsstopp, LookAhead (B1)

Syntax**Aktivera dynamikvärden:**

DYNNORM
DYNPOS
DYNROUGH
DYNSEMIFIN
DYNFINISH

Märk

Dynamikvärden blir verksamma redan i det block i vilket det tillhörande G-kommandot programmeras. Det följer inget bearbetningsstopp.

Läsa eller skriva bestämt fältelement:

R<m>=\$MA... [n, X]
\$MA... [n, X]=<Wert>

Betydelse

DYNNORM:	G-kommando för aktivering av den normala dynamiken		
DYNPOS:	G-kommando för aktiveringen av dynamiken för positioneringsdrift, gängtappning		
DYNROUGH:	G-kommando för aktiveringen av dynamiken för grovbearbetning		
DYNSEMIFIN:	G-kommando för aktiveringen av dynamiken för finbearbetning		
DYNFINISH:	G-kommando för aktiveringen av dynamiken för mycket fin bearbetning		
R<m>:	Räkneparameter med nummer <m>		
\$MA...[n,X]:	Maskindatum med dynamikbestämmande fältelement		
<n>:	Fältindex		
	Värdeområde:		0 ... 4
	0	Normal dynamik (DYNNORM)	
	1	Dynamik för positioneringsdrift (DYNPOS)	
	2	Dynamik för grovbearbetning (DYNROUGH)	
	3	Dynamik för finbearbetning (DYNSEMIFIN)	
	4	Dynamik för mycket fin bearbetning (DYNFINISH)	
<X> :	Axeladress		
<Wert>:	Dynamikvärde		

Exempel

Exempel 1: Aktivera dynamikvärden

Programkod	Kommentar
DYNNORM G1 X10	; Grundläge
DYNPOS G1 X10 Y20 Z30 F...	; Positioneringsdrift, gängtappning
DYNROUGH G1 X10 Y20 Z30 F10000	; Grovbearbetning
DYNSEMIFIN G1 X10 Y20 Z30 F2000	; Finbearbetning
DYNFINISH G1 X10 Y20 Z30 F1000	; Finbearbetning

Exempel 2: Läs eller skriv bestämt fältelement

Maximal acceleration för grovbearbetning, axel X.

Programkod	Kommentar
R1=\$MA_MAX_AX_ACCEL[2,X]	; Läs
\$MA_MAX_AX_ACCEL[2,X]=5	; Skriva

12.4 Köra med förstyrning (FFWON, FFWOF)

Genom förstyrningen reduceras den hastighetsberoende medföljningsvägen vid bankörning mot noll. Körning med förstyrning möjliggör högre bannoggrannhet och därmed bättre tillverkningsresultat.

Syntax

FFWON

FFWOF

Betydelse

FFWON:	Kommando för till koppling av förstyrningen
FFWOF:	Kommando för från koppling av förstyrningen

Märk

Via maskindata fastläggs typen av förstyrning och vilka banaxlar som ska förflyttas med förstyrning.

Standard: Hastighetsberoende förstyrning

Option: Accelerationsberoende förstyrning

Exempel

Programkod

N10 FFWON

N20 G1 X... Y... F900 SOFT

12.5 Programmerbar konturprecision (CPRECON, CPRECOF)

Funktionen "Programmerbar konturprecision" reducerar banfelen på böjda konturer genom automatisk anpassning av hastigheten.

Den konturprecision som ska respekteras föreskrivs beroende på projekteringen av maskinen (MD20470 \$MC_MC_CPREC_WITH_FFW; se uppgifter från maskintillverkaren) antingen via settingdatum \$SC_CONTPREC eller via den programmerade konturtoleransen CTOL. Ju mindre värdet och ju mindre K_v -faktorn för geometriaxlarna är desto mer sänks banmatningen på böjda konturer.

Funktionen "Programmerbar konturprecision" kopplas till resp. från via anvisningarna CPRECON och CPRECOF i NC-programmet.

Syntax

CPRECON

...

CPRECOF

Betydelse

CPRECON:	G-kommandoanrop: Koppla till "Programmerbar konturprecision"	
	Verkan:	modal
CPRECOF:	G-kommandoanrop: Koppla från "Programmerbar konturprecision"	
	Verkan:	modal

CPRECON och CPRECOF bildar tillsammans G-funktionsgrupp 39 (Programmerbar konturprecision).

Märk

Via settingdatum \$SC_MINFEED (minsta banmatning vid CPRECON) kan användaren föreskriva en minsta hastighet för banmatningen.

Matningen begränsas inte under detta värde, för såvida inte ett lägre F-värde programmerades eller de dynamiska begränsningarna av axlarna tvingar fram en lägre banhastighet.

Märk

Funktionen "Programmerbar konturprecision" betraktar endast geometriaxlarna till banan. Den har inget inflytande på positioneraxlarnas hastigheter.

Exempel

Programkod	Kommentar
N10 G0 X0 Y0	
N20 CPRECON	; Tillkoppling av "Programmerbar konturprecision".

12.5 Programmerbar konturprecision (CPRECON, CPRECOF)

Programkod	Kommentar
N30 G1 G64 X100 F10000	; Bearbetning med 10 m/min i banstyrningsdriften.
N40 G3 Y20 J10	; Automatisk matningsbegräsning i cirkelblocket.
N50 G1 X0	; Matning åter utan begränsning (10 m/min).
...	
N100 CPRECOF	; Frånkoppling av "Programmerbar konturprecision".
N110 G0 ...	

Litteratur

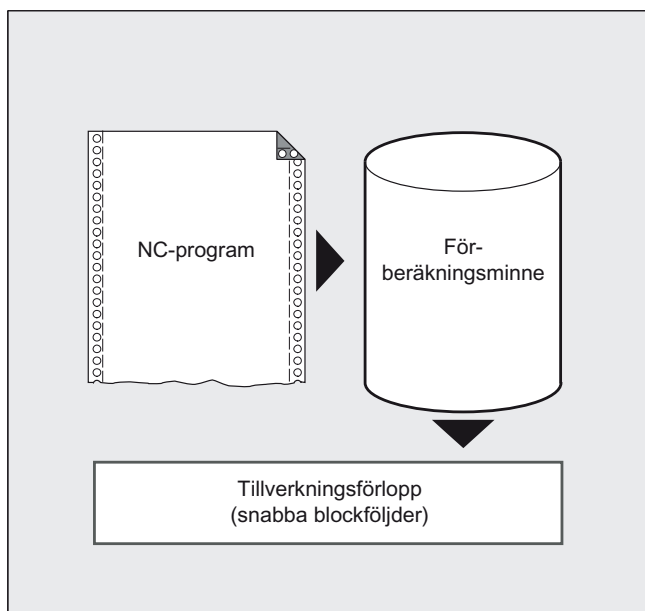
För programmering av CTOL se "Programmera kontur-/orienteringtolerans (CTOL, OTOL, ATOL) (Sida 538)"

För utförliga informationer till funktionen "Programmerbar konturprecision" se:

Funktionshandbok Specialfunktioner; Konturtunnel-övervakning (K6), Kapitel: "Programmerbar konturprecision"

12.6 Programförlopp med förberäkningsminne (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE)

Beroende på utbyggnadssteg förfogar styrningen över en viss mängd så kallade förberäkningsminnen, som sparar färdigt förberedda block före genomarbetningen och matar ut dem under tillverkningsförloppet som snabba blockföljder. Härigenom låter sig korta vägar köras med höga hastigheter. Så långt som styrningens resttid tillåter blir förberäkningsminnet principiellt fyllt.



Känneteckna bearbetningsavsnitt

Bearbetningsavsnittet som ska sparas intermediärt i förberäkningsminnet, kännetecknas i detaljprogrammet i början med "STOPFIFO" och i slutet med "STARTFIFO". Genomarbetningen av de förberedda och mellanlagrade blocken börjar först efter kommandot "STARTFIFO" eller när förberäkningsminnet är fullt.

Automatisk styrning för förberäkningsminne

Den automatiska styrningen för förberäkningsminnet anropas med kommandot "FIFOCTRL". "FIFOCTRL" verkar först precis om "STOPFIFO". Vid varje programmering väntas tills förberäkningsminnet är fullt, sedan börjar genomarbetningen. Däremot är beteendet olika vid tomkörning av förberäkningsminnet: med "FIFOCTRL" reduceras banhastigheten alltmer från och med en fyllningsnivå på 2/3, för att förhindra en komplett tomkörning och en bromsning till stillestånd.

Fördekoderingsstopp

Blockförberedelsen och -mellanlagringen stoppas, när kommandot "STOPRE" programmeras i blocket. Det efterföljande blocket utförs först när alla tidigare förberedda och sparade block har genomarbetats fullständigt. Det föregående blocket stoppas i precisionsstopp (som G9).

OBSERVERA**Programavbrott**

Vid tillkopplad verktygskompensering och vid spline-interpoleringar bör inget "STOPRE" programmeras, eftersom annars sammanhörande blockföljder blir åtskilda.

Syntax

Tabell 12-1 Känneteckna bearbetningsavsnitt:

```
STOPFIFO
...
STARTFIFO
```

Tabell 12-2 Automatisk styrning för förberäkningsminne:

```
...
FIFCTRL
...
```

Tabell 12-3 Fördekoderingsstopp:

```
...
STOPRE
...
```

Märk

Kommandona "STOPFIFO", "STARTFIFO", "FIFCTRL" och "STOPRE" måste programmeras i det egna blocket.

12.6 Programförlopp med förberäkningsminne (STOPFIFO, STARTFIFO, FIFOCTRL, STOPRE)

Betydelse

STOPFIFO:	"STOPFIFO" kännetecknar början på ett bearbetningsavsnitt, som ska mellanlagras i förberäkningsminnet. Med "STOPFIFO" stoppas bearbetningen och förberäkningsminnet fylls tills: • "STARTFIFO" eller "STOPRE" identifieras eller • förberäkningsminnet är fullt eller • programslut har uppnåtts.
STARTFIFO:	Med "STARTFIFO" startar den snabba genomarbetningen av bearbetningsavsnittet, parallellt därtill sker påfyllningen av förberäkningsminnet
FIFOCTRL:	Tillkoppling av den automatiska styrningen för förberäkningsminnet
STOPRE:	Stoppa förkörning

Märk

Påfyllningen av förberäkningsminnet utförs inte resp. avbryts, när bearbetningsavsnittet innehåller kommandon, som tvingar fram en obuffrad drift (referenspunktkörning, mätfunktioner, ...).

Märk

Vid åtkomst till maskinens tillståndsdata (\$SA...) skapar styrningen internt fördekoderingsstopp.

Exempel: Stoppa förkörning

Programkod	Kommentar
...	
N30 MEAW=1 G1 F1000 X100 Y100 Z50	; Mätblock med mätprobe till den första mätingången och linjärinterpolering.
N40 STOPRE	; Fördekoderingsstopp.
...	

12.7 Definiera Stop-Delay-områden (DELAYFSTON, DELAYFSTOF)

För definition av ett villkorligt stoppbart område i detaljprogrammet (Stop-Delay-område) tjänar de fördefinierade procedurerna DELAYFSTON och DELAYFSTOF.

Märk

I synkronaktioner är DELAYFSTON och DELAYFSTOF **inte** tillåtna!

Syntax

```
DELAYFSTON
...
DELAYFSTOF
```

Betydelse

DELAYFSTON:	Definiera början på ett Stop-Delay-område	
	Ensam i blocket:	ja
DELAYFSTOF:	Definiera slutet på ett Stop-Delay-område	
	Ensam i blocket:	ja

Programmeringsexempel

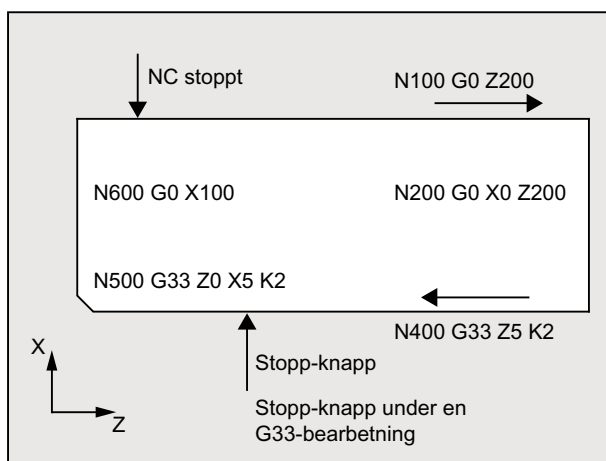
I en slinga upprepas följande programblock:

Programkod

```
...
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
...
```

I den följande bilden syns att användaren trycker "Stopp" i Stop-Delay-området och NC börjar bromsförloppet utanför Stop-Delay-området dvs. i block N100. Därmed kommer NC i det främre området av N100 till stillestånd.

12.7 Definiera Stop-Delay-områden (DELAYFSTON, DELAYFSTOF)



Ytterligare informationer

Underprogramslut

Med slutet på det underprogram, i vilket DELAYFSTON anropades, aktiveras DELAYFSTOF implicit.

Kapslingsdjup

Anropar underprogram 1 i ett Stop-Delay-område underprogram 2, så är underprogram 2 komplett ett Stop-Delay-område. Speciellt är DELAYFSTOF utan verkan i underprogram 2.

Exempel:

Programkod	Kommentar
N10010 DELAYFSTON	; Block med N10xxx programnivå 1.
N10020 R1 = R1 + 1	
N10030 G4 F1	; Stop-Delay-område börjar.
...	
N10040 Unterprogramm2	
...	
...	; Interpretering av underprogram 2.
N20010 DELAYFSTON	; Overksam, upprepad början, 2:a nivån.
...	
N20020 DELAYFSTOF	; Overksam, slut i andra nivån.
N20030 RET	
N10050 DELAYFSTOF	; Stop-Delay-områdesslut i samma plan.
...	
N10060 R2 = R2 + 2	
N10070 G4 F1	; Stop-Delay-område slutar. Stopp verkar från och med nu omedelbart.

Systemvariabler

Om detaljprogrambearbetningen vid den aktuella tidpunkten befinner sig i ett Stop-Delay-område, kan avfrågas med hjälp av följande systemvariabler:

- i detaljprogrammet med \$P_DELAYFST
- i synkronaktionerna med \$AC_DELAYFST

Värde	Betydelse
0	Stop-Delay-område inte aktivt
1	Stop-Delay-område aktivt

12.8 Förhindra programställe för SERUPRO (IPTRLOCK, IPTRUNLOCK)

För bestämda komplicerade mekaniska situationer på maskinen är det erforderligt att förhindra blocksökningen SERUPRO.

Med en programmerbar stoppningsvisare finns ingripningsmöjligheten att vid "Sökning till stoppstället" sätta framför det sökningsodugliga stället.

Det kan också definieras sökningsodugliga områden i detaljprogramområden i vilka NC ännu inte åter kan stiga in. Med programavbrott märker NC det sist genomarbetade blocket, på vilket kan sökas via användargränssnittet HMI.

Syntax

IPTRLOCK
IPTRUNLOCK

Kommandona står ensamma i en detaljprogramrad och möjliggör en programmerbar stoppvisare

Betydelse

IPTRLOCK:	Början på det sökodugliga programavsnittet
IPTRUNLOCK:	Slut på det sökodugliga programavsnittet

Båda kommandona är tillåtna endast i detaljprogram men **inte** i synkronaktioner.

Exempel

Kapsling sökodugliga programavsnitt i två programnivåer med implicit "IPTRUNLOCK". Det implicita "IPTRUNLOCK" i underprogram 1 avslutar det sökodugliga området.

Programkod	Kommentar
N10010 IPTRLOCK()	
N10020 R1 = R1 + 1	
N10030 G4 F1	; Stoppblock, det sökodugliga programavsnittet börjar.
...	
N10040 Unterprogramm2	
...	; Interpretering av underprogram 2.
N20010 IPTRLOCK ()	; Överksam, upprepad början.
...	
N20020 IPTRUNLOCK ()	; Överksam, slut i andra nivån.
N20030 RET	
...	
N10060 R2 = R2 + 2	
N10070 RET	; Slut på det sökodugliga programavsnittet.
N100 G4 F2	; Huvudprogram fortsätts.

Ett stopp på 100 levererar då åter stoppvisaren.

Ytterligare information

Registrera och söka sökodugliga områden

De sökodugliga programavsnitten markeras med språkkommandona "IPTRLOCK" och "IPTRUNLOCK".

Kommandot "IPTRLOCK" fryser stoppvisaren på ett i huvudkörningen utförbart enkelblock (SBL1). Detta block betecknas i det följande som stoppblock. Uppträder efter "IPTRLOCK" ett programavbrott, så kan på användargränssnittet HMI sökas efter detta så kallade stoppblock.

Åter ställa på det aktuella blocket

Stoppvisaren kommer med "IPTRUNLOCK" för det efterföljande programavsnittet att sättas på det aktuella blocket till stoppunkt.

Efter ett hittat sökmål kan med själva stoppblocket ett nytt sökmål upprepas.

En av användaren redigerad stoppvisare måste åter avlägsnas via HMI.

Regler vid kapsling

Följande punkter reglerar samspelet mellan språkkommandona "IPTRLOCK" och "IPTRUNLOCK" med kapslingar och underprogramslutet:

1. Med slutet på det underprogram, i vilket "IPTRLOCK" anropades, aktiveras "IPTRUNLOCK" implicit.
2. "IPTRLOCK" i ett sökodugligt område förblir utan verkan.
3. Anropar Unterprogramm1 i ett sökodugligt område Unterprogramm2, så förblir Unterprogramm2 fullständigt sökodugligt. Speciellt är "IPTRUNLOCK" utan verkan i Unterprogramm2.

Ytterligare informationer härtill se

/FB1/ Funktionshandbok Grundfunktioner; BAG, Kanal, Programdrift (K1).

Systemvariabel

Ett sökodugligt område kan identifieras med "\$P_IPTRLOCK" i detaljprogrammet.

Automatisk stoppvisare

Funktionern automatisk stoppvisare fastlägger automatiskt en tidigare fastlagd kopplingstyp som sökoduglig. Med hjälp av maskindatum blir för

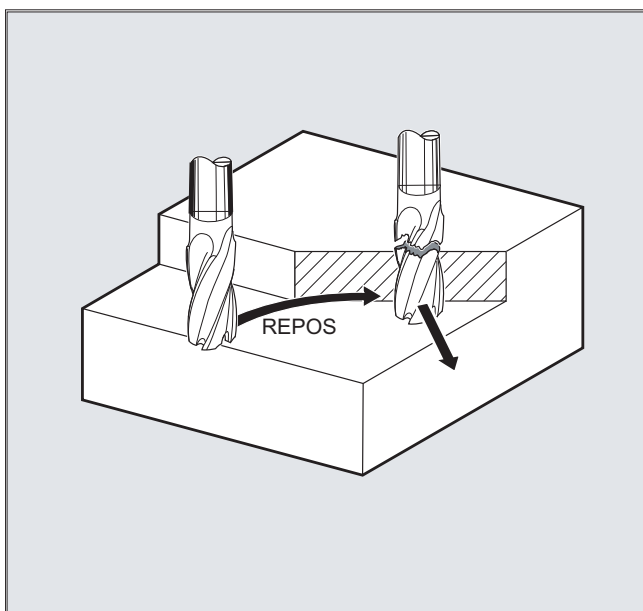
- Elektronisk växel vid "EGON"
- Axial styrvärdeskoppling vid "LEADON"

den automatiska stoppvisaren aktiverad. Överlappar den programmerade och den via maskindatum aktiverbara automatiska stoppvisaren varandra så bildas det största möjliga sökodugliga området.

12.9 Återstart på kontur (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)

Om du under bearbetningen stoppar det pågående programmet och frikör verktyget – t. ex. på grund av verktygsbrott eller för mätning av arbetsstycket – kan du programstyrt åter uppsöka konturen vid en valbar punkt.

Kommandot `REPOS` verkar i en ASUP som ett underprogram-återhopp (t.ex. M17). Efterföljande block utförs inte längre. För stopp i programkörningen se även "Interruptutin (ASUP) (Sida 124)".



Syntax

```
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=...
```

Betydelse**Val av framkörningsväg**

REPOSA:	Återstart på konturen med geometriaxlarna på en rät linje. Alla andra kanalaxlar blir också repositionerade.
REPOSL:	Återstart på konturen med geometriaxlarna på en rät linje. Alla andra kanalaxlar måste explicit programmeras.
REPOSQ DISR=... :	Återstart på konturen med geometriaxlarna på en fjärdedels cirkel med radien DISR. Alla andra kanalaxlar måste explicit programmeras.
REPOSQA DISR=... :	Återstart på konturen med geometriaxlarna på en fjärdedels cirkel med radien DISR. Alla andra kanalaxlar blir också repositionerade.
REPOSH DISR=... :	Återstart på konturen med geometriaxlarna på en halvcirkel med diametern DISR. Alla andra kanalaxlar måste explicit programmeras.
REPOSHA DISR=... :	Återstart på konturen med geometriaxlarna på en halvcirkel med radien DISR. Alla andra kanalaxlar blir också repositionerade.

Val av återstartpunkt

RMIBL:	Uppsöka stoppunkt
RMIBL DISPR=...:	Inträdeshpunkt på avståndet DISPR i mm/inch före stoppunkt
RMBBL:	Uppsöka blockbegränsningspunkt
RMEBL:	Uppsöka blockslutpunkt
RMEBL DISPR=... :	Uppsöka blockslutpunkt på avståndet DISPR före slutpunkten
RMNBL:	Uppsöka den närmast liggande banpunkten
A0 B0 C0 :	Axlar i vilka ska uppsökas

Märk**Kompatibilitet**

För att förbli kompatibel med äldre softwareversioner kan du även i fortsättningen programmera REPOS-startmode vid de modala kommandona RMI, RMB, RME och RMN. Vid användning inom en ASUP ska denna föras med attributet SAVE i PROC-anvisningen. I annat fall verkar det i ASUP använda modala REPOS-startmode, när det avviker från förinställningen RMI, även i efterföljande REPOS-förlopp.

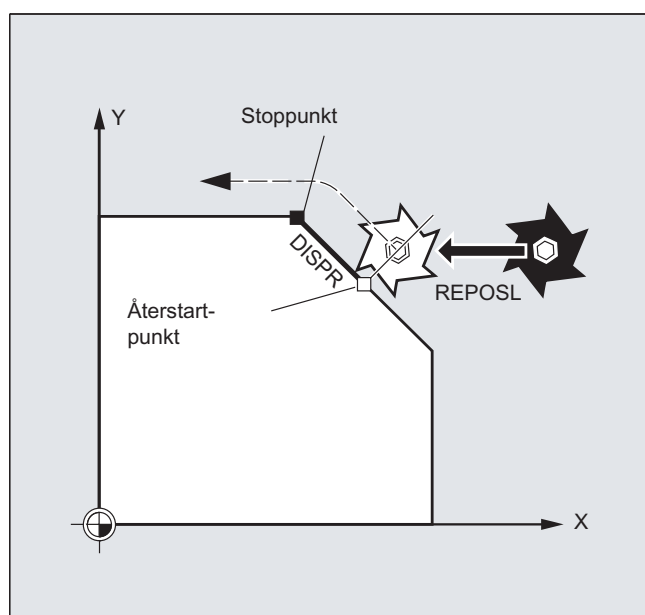
Återuppsökning av konturen på en rät linje, REPOSA, REPOSL

Verktöget uppsöker återstartpunkten direkt på en rät linje.

Exempel

```
REPOSL RMIBL DISPR=6 F400
```

12.9 Återstart på kontur (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)

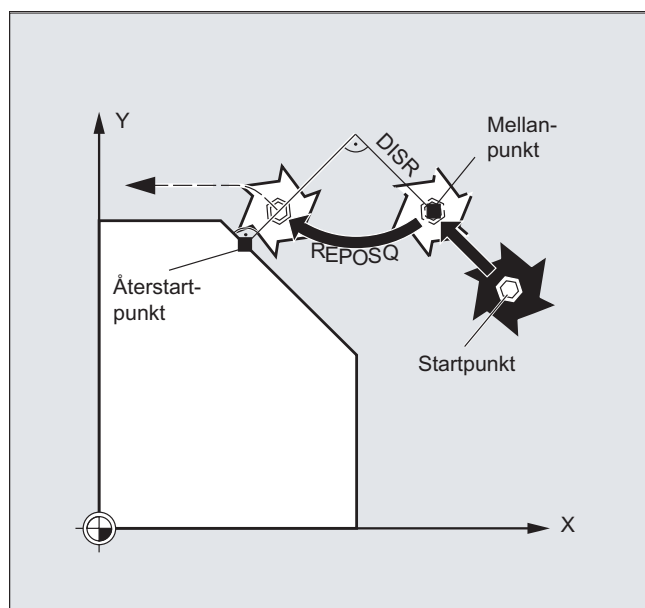


Återuppsökning av konturen med fjärdedels cirkel, REPOSQ, REPOSQA

Verktuget uppsöker återstartpunkten på en fjärdedels cirkel med radie $DISR = \dots$. Den nödvändiga mellanpunkten mellan start- och återstartpunkt beräknar styrningen automatiskt.

Exempel

REPOSQ RMIBL DISR=10 F400

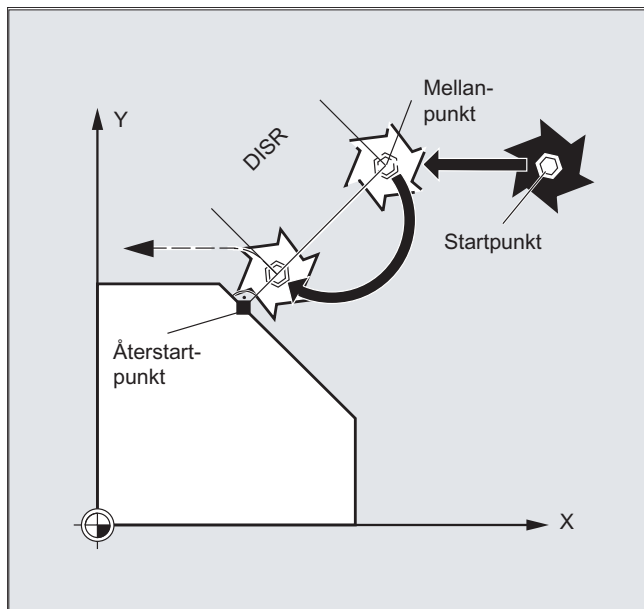


Återuppsökning av konturen med halvcirkel, REPOSH, REPOSHA

Verktøget uppsöker återstartpunkten på en halvcirkel med radie $DISR = \dots$. Den nödvändiga mellanpunkten mellan start- och återstartpunkt beräknar styrningen automatiskt.

Exempel

REPOSH RMIBL DISR=20 F400

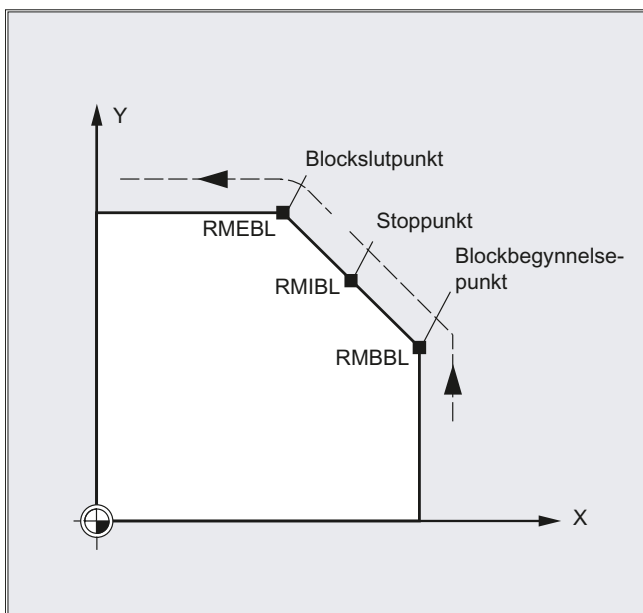


Fastlägga återstartpunkt (inte för SERUPRO start med RMNBL)

Hänförd till det NC-block i vilket programförloppet stoppades kan du välja mellan tre återstartpunkter:

- RMIBL, stoppunkt
- RMBBL, blockbegränsningspunkt resp. sista slutpunkt
- RMEBL, blockslutpunkt

12.9 Återstart på kontur (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)



Med RMIBL DISPR=... resp. med RMEBL DISPR=... kan du fastlägga en återstartpunkt, som ligger före stoppunkten resp. före blockslutpunkten.

Med DISPR=... beskriver du den konturväg i mm/inch, med vilken återstartpunkten ligger före stopp- resp. slutpunkten. Denna punkt kan - också för större värden - maximalt ligga i blockbegynnelspunkten.

Programmeras ingen DISPR=... gäller DISPR=0 och därmed stoppunkten (vid RMIBL) resp. blockslutpunkten (vid RMEBL).

Förtecken till DISPR

Förtecknet till DISPR utvärderas. Vid positivt förtecken är beteendet som hittills.

Vid negativt förtecken sätts åter på bakom stoppunkten resp. vid RMBBL bakom startpunkten.

Avståndet stoppunkt-påställningspunkt resulterar ur beloppet för DISPR. Denna punkt kan också för beloppsmässigt större värden maximalt ligga i blockslutpunkten.

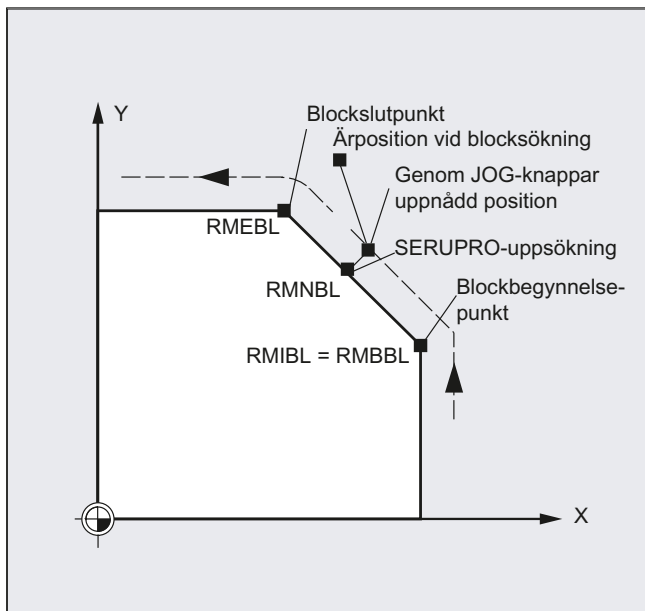
Användningsexempel:

Genom en sensor identifieras närmandet av ett spännidon. En ASUP utlöses med vilken spännidonet kan rundas.

Sedan repositioneras med negativ DISPR på en punkt bakom spännidonet och programmet fortsätts.

SERUPRO-start med RMNBL

Blir för bearbetningen på ett godtyckligt ställe ett avbrott framtvingat, då uppsöks med SERUPRO-start under RMNBL den kortaste vägen från avbrottsstället för att sedan genomarbete endast restvägen. Därtill startar användaren ett SERUPRO-förlopp på stoppblocket och positionerar med JOG-knappen före det skadade stället på målblocket.



Märk

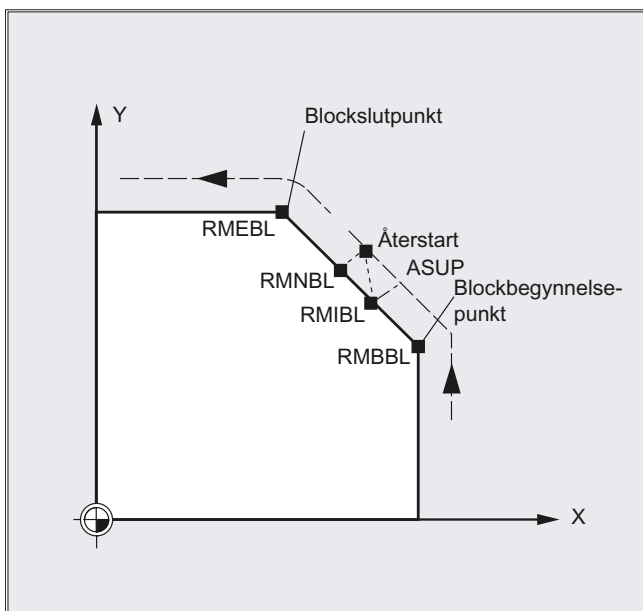
SERUPRO

För SERUPRO är RMIBL och RMBBL identiska. RMNBL är inte begränsad till bara SERUPRO utan allmänt giltig.

12.9 Återstart på kontur (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)

Start från närmast liggande banpunkt RMNBL

Vid interpreteringstidpunkten för REPOSA blir efter ett stopp återstartblocket med RMNBL inte än en gång komplett börjat, utan endast restvägen genomarbetas. Den banpunkt som ligger närmast det stoppade blocket uppsöks.



Status för giltigt REPOS-mode

Den giltiga REPOS-moden för det stoppade blocket kan läsas via synkronaktioner med variabeln \$AC_REPOS_PATH_MODE:

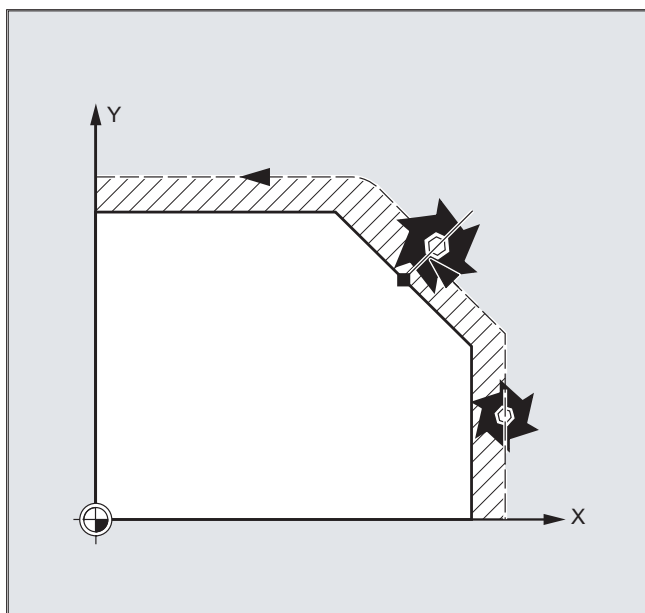
- 0 Uppsökning inte definierad
- 1 RMBBL: Uppsökning av början
- 2 RMIBL: Uppsökning av stoppunkten
- 3 RMEBL: Uppsökning av blockslutpunkten
- 4 RMNBL: Uppsökning av den närmast liggande banpunkten i det stoppade blocket

Start med nytt verktyg

Om programförloppet stoppades på grund av verktygsbrott:

Med programmeringen av det nya D-numret fortsätts programmet från återstartpunkten med de ändrade verktygskompenseringsvärdena.

Vid ändrade verktygskompenseringsvärden kan stoppunkten eventuellt inte längre uppsökas. I detta fall uppsöks den punkt som ligger närmast stoppunkten på den nya konturen (eventuellt modifierad med DISPR).



Uppsöka kontur

Den rörelse med vilken verktyget åter uppsöker konturen är programmerbar. Adresserna för de axlar som ska förflyttas anger du med värdet noll.

Med kommandona REPOSA, REPOSQA och REPOSHA repositioneras automatiskt alla axlar. Ingen axeluppgift är nödvändig.

Vid programmering av REPOSL, REPOSQ och REPOSH startar alla geometriaxlar automatiskt, alltså också utan uppgift i kommandot. Alla andra axlar måste anges i kommandot.

För cirkelrörelserna REPOSH och REPOSQ gäller:

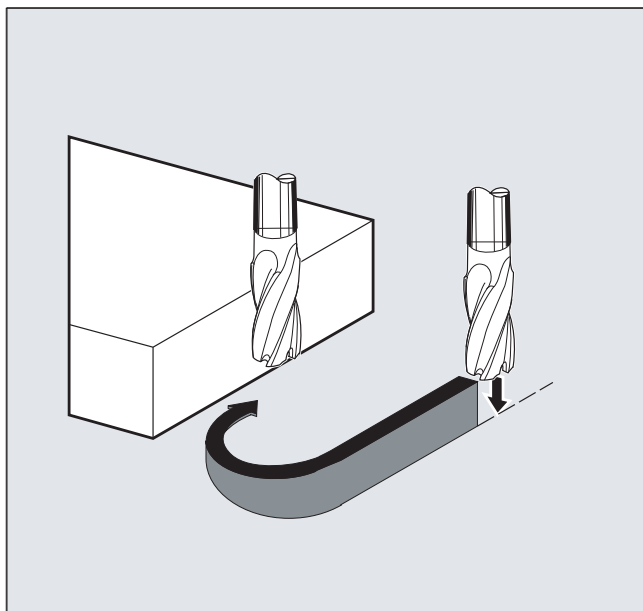
Cirkeln uppsöks i det angivna arbetsplanet G17 till G19.

Om du i startblocket anger den tredje geometriaxeln (ansättningsriktning) uppsöks återstartpunkten, i fall verktygsposition och programmerad position i ansättningsriktning inte överensstämmer, längs en skruvlinje.

12.9 Återstart på kontur (REPOSA, REPOSL, REPOSQ, REPOSQA, REPOSH, REPOSHA, DISR, DISPR, RMIBL, RMBBL, RMEBL, RMNBL)

I följande fall kopplas automatiskt om till linjär uppsökning REPOSL:

- Du har inte angivit något värde för DISR.
- Det finns ingen definierad startriktning (programstopp i ett block utan förflyttningsinformation).
- Vid startriktning vinkelrätt mot det aktuella arbetsplanet.



12.10 Påverkan av rörelsestyrningen

12.10.1 Procentuell jerkkompensering (JERKLIM)

Med NC-kommandot "JERKLIM" kan den med maskindatum inställda maximalt möjliga jerken för en axel reduceras eller höjas vid banrörelse i kritiska programavsnitt.

Förutsättning

Accelerationsmode SOFT måste vara aktiv.

Verkan

Funktionen verkar:

- i AUTOMATIK-driftlägen.
- endast på banaxlar.

Syntax

JERKLIM[<Achse>]=<Wert>

Betydelse

JERKLIM:	Kommando för jerkkompensering	
<Achse>:	Maskinaxel vars jerkgränsvärde ska anpassas.	
<Wert>:	Procentuellt kompeneringsvärde relaterat till den projekterade maximala axeljerken vid banrörelse (MD32431 \$MA_MAX_AX_JERK).	
	Värdeområde:	1 ... 200
	Värdet 100 har inget inflytande på jerken.	

Märk

Beteendet hos JERKLIM vid detaljprogramslut och kanal-reset projekteras med Bit 0 i maskindatum MD32320 \$MA_DYN_LIMIT_RESET_MASK:

- Bit 0 = 0:
Det programmerade värdet för JERKLIM återställs med kanal-reset/M30 till 100 %.
- Bit 0 = 1:
Det programmerade värdet för JERKLIM bibehålls utöver kanal-reset/M30.

Exempel

Programkod	Kommentar
...	

Programkod	Kommentar
N60 JERKLIM[X]=75	; Axelsliden i X-riktningen ska accelereras/fördröjas endast med maximalt 75% av den för axeln tillåtna jerken.
...	

12.10.2 Procentuell hastighetskompensering (VELOLIM)

Med kommandot VELOLIM kan i detaljprogrammet eller synkronaktion den via maskindatum inställda maximalt möjliga hastigheten för en axel resp. det maximalt möjliga växelstegsberoende varvtalet för en spindel reduceras.

Verkan

Funktionen verkar:

- i AUTOMATIK-driftlägen.
- på ban- och positioneringsaxlar.
- på spindlar i spindel-/axeldrift

Syntax

VELOLIM[<Achse/Spindel>]=<Wert>

Betydelse

VELOLIM:	Kommando för hastighetskompensering	
<Achse/Spindel>:	Axel eller spindel vars hastighets- eller varvtalsgränsvärde ska anpassas. VELOLIM för spindlar Via maskindatum (MD30455 \$MA_MISC_FUNCTION_MASK, Bit 6) kan för programmeringen i detaljprogrammet ställas in om "VELOLIM" oberoende av den aktuella användningen verkar som spindel eller axel (Bit 6 = 1) eller ska vara programmerbar separat för varje driftsläge (Bit 6 = 0). Är en åtskild verkan projekterad då görs valet via beteckningarna vid programmeringen: • Spindelbeteckning S<n> för spindel driftslägen • Axelbeteckning, t. ex. "C", för axeldriften	
<Wert>:	Procentuellt kompenseringsvärde Kompenseringsvärdet hänför sig: • vid axlar / spindlar i axeldrift (MD30455, bit 6 == 0): till den projekterade maximala axelhastigheten (MD32000 \$MA_MAX_AX_VELO). • vid spindlar i spindel- eller axeldrift (MD30455 bit 6 == 1): till maximalvarvtalet för det aktiva växelsteget (MD35130 \$MA_GEAR_STEP_MAX_VELO_LIMIT[<n>])	
	Värdeområde:	1 ... 100
	Värdet 100 har inget inflytande på hastighet resp. varvtal.	

Märk**Beteende vid detaljprogramslut och kanal-reset**

Beteendet hos "VELOLIM" vid detaljprogramslut och kanal-reset kan ställas in via maskindatum: MD32320 \$MA_DYN_LIMIT_RESET_MASK, Bit 0

Identifikation av en aktiv varvtalsbegränsning i spindeldriften

En varvtalsbegränsning genom "VELOLIM" (mindre än 100 %) kan i spindeldriften identifieras via följande systemvariabel:

- \$AC_SMAXVELO (maximalt möjligt spindelvarvtal)
- \$AC_SMAXVELO_INFO (kännetecken för den varvtalsbegränsande orsaken)

Exempel**Exempel 1: Hastighetsbegränsning maskinaxel**

Programkod	Kommentar
...	
N70 VELOLIM[X]=80	; Axelsliden i X-riktningen ska förflyttas endast med maximalt 80 % av den för axeln tillåtna hastigheten.
...	

Exempel 2: Varvtalsbegränsning spindel

Programkod	Kommentar
N05 VELOLIM[S1]=90	Begränsning av maximalvarvtalet för spindel 1 till 90 % av 1000 varv/min.
...	
N50 VELOLIM[C]=45	Begränsning av varvtalet till 45 % av 1000 varv/min, C är axelbeteckningen för S1.
...	

Maskindatainställningar för spindel 1 (AX5)

- Maximalt varvtal för växelsteg 1 = 1000 varv/min:
MD35130 \$MA_GEAR_STEP_MAX_VELO_LIMIT[1, AX5] = 1000
- Programmeringen av "VELOLIM" verkar gemensamt för spindel- och axeldrift oberoende av programmerad beteckning:
MD30455 \$MA_MISC_FUNCTION_MASK[AX5], Bit 6 = 1

12.10.3 Programexempel för JERKLIM och VELOLIM

Det följande programmet framställer ett användningsexempel för den procentuella jerk- och hastighetsbegränsningen:

Programkod	Kommentar
N1000 G0 X0 Y0 F10000 SOFT G64	
N1100 G1 X20 RNDM=5 ACC[X]=20	
ACC[Y]=30	
N1200 G1 Y20 VELOLIM[X]=5	; Axelsliden i X-riktningen ska förflyttas endast med max. 5% av den för axeln tillåtna hastigheten.
JERKLIM[Y]=200	; Axelsliden i Y-riktningen kan accelereras/fördröjas med max. 200% av den för axeln tillåtna jerken.
N1300 G1 X0 JERKLIM[X]=2	; Axelsliden i X-riktningen ska accelereras/fördröjas endast med max. 2% av den för axeln tillåtna jerken.
N1400 G1 Y0	
M30	

12.11 Programmera kontur-/orienteringtolerans (CTOL, OTOL, ATOL)

Med adresserna CTOL, OTOL och ATOL kan de via maskin- och settingdata parametrerade bearbetningstoleranserna för kompressor-funktionerna, översläpning och orienteringsjämnning i detaljprogrammet anpassas.

De programmerade toleransvärdena gäller tills de programmeras på nytt eller raderas genom tillordning av ett negativt värde. Dessutom raderas de vid programslut resp. reset. Efter raderingen är de parametrerade toleransvärdena åter verksamma.

Syntax

```
CTOL=<Value>
OTOL=<Value>
ATOL[<Axis>]=<Value>
```

Betydelse

CTOL:	Adress för programmering av konturtoleransen		
	Användningsområde:	• alla kompressorfunktioner • alla översläpningstyper utom G641 och G644	
	Fördekoderingsstopp:	nej	
	Verkan:	modal	
	<Value>:	Värdet för konturtoleransen är en längduppgift.	
		Typ:	REAL
		Enhet:	Inch/mm (beroende på den aktuella inställningen för måttuppgiften)
		Värdeområde:	≥ 0: Toleransvärde < 0: Radera det programmerade toleransvärdet ⇒ Det gäller åter det i maskin- resp. settingdatum parametrerade toleransvärdet
OTOL:	Adress för programmering av orienteringtoleransen		
	Användningsområde:	• alla kompressorfunktioner • orienteringsjämnning ORISON • alla översläpningstyper utom G641, G644 och OSD	
	Fördekoderingsstopp:	nej	
	Verkan:	modal	
	<Value>:	Värdet för orienteringtoleransen är en vinkeluppgift.	
		Typ:	REAL
		Enhet:	grader
		Värdeområde:	≥ 0: Toleransvärde < 0: Radera det programmerade toleransvärdet ⇒ Det gäller åter det i maskin- resp. settingdatum parametrerade toleransvärdet

12.11 Programmera kontur-/orienteringstolerans (CTOL, OTOL, ATOL)

ATOL:	Adress för programmering av en axelspecifik tolerans		
	Användningsområde:	• alla kompressorfunktioner • orienteringsjämning ORISON • alla översläpningstyper utom G641, G644 och OSD	
	Fördekoderingsstopp:	nej	
	Verkan:	modal	
	<Axis>:	Namn på den kanalaxel, för vilken den programmerade toleransen ska verka	
	<Value>:	Värdet för axeltoleransen är beroende på axeltyp (linjär eller roterande axel) en längd- eller vinkeluppgift.	
		Typ:	REAL
		Enhet:	för linjäraaxlar: Inch/mm (beroende på den aktuella inställningen för måttuppgiften)
			för roterande axlar: grader
		Värdeområde:	≥ 0: Toleransvärde
			< 0: Radera det programmerade toleransvärdet ⇒ Det gäller åter det i maskin- resp. settingdatum parameterade toleransvärdet

Märk

De med CTOL och OTOL programmerade kanalspecifika toleransvärdena har högre prioritet än de med ATOL programmerade axelspecifika toleransvärdena.

Märk**Skaländrande frames**

Skaländrande frames verkar på de programmerade toleranserna på samma sätt som på axelpositionerna dvs. den relativa toleransen förblir lika.

Exempel

Programkod	Kommentar
COMPCAD G645 G1 F10000	; Aktivera kompressor-funktion COMPCAD.
X... Y... Z...	; Här verkar maskin- och settingdata.
X... Y... Z...	
X... Y... Z...	
CTOL=0.02	; Från och med här verkar en konturtolerans på 0,02 mm.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	
ASCALE X0.25 Y0.25 Z0.25	; Från och med här verkar en konturtolerans på 0,005 mm.
X... Y... Z...	

Programkod	Kommentar
X... Y... Z...	
X... Y... Z...	
CTOL=-1	; Från och med här verkar åter maskin- och setting-data.
X... Y... Z...	
X... Y... Z...	
X... Y... Z...	

Systemvariabler

Läsa med fördekoderingsstopp

Via följande systemvariabler kan de aktuellt verksamma toleranserna läsas i detaljprogrammet och synkronaktioner:

- **\$AC_CTOL**
Kanalspecifik konturtolerans, som var verksam vid förberedelsen av det aktuella huvudkörningsblocket
Om ingen konturtolerans är verksam levererar \$AC_CTOL roten ur summan på kvadraterna för toleranserna till geometriaxlarna.
- **\$AC_OTOL**
Kanalspecifik orienteringstolerans, som var verksam vid förberedelsen av det aktuella huvudkörningsblocket.
Om ingen orienteringstolerans är verksam levererar \$AC_OTOL under en aktiv orienteringstransformation roten ur summan av kvadraterna för toleranserna till orienteringsaxlarna, annars värdet "-1".
- **\$AA_ATOL[<Achse>]**
Axelspecifik konturtolerans, som var verksam vid förberedelsen av det aktuella huvudkörningsblocket
Om en konturtolerans är aktiv levererar \$AA_ATOL[<Geometrieachse>] konturtoleransen delad genom roten ur antalet geometriaxlar.
Om en orienteringstolerans och en orienteringstransformation är aktiva levererar \$AA_ATOL[<Orientierungsachse>] orienteringstoleransen delad genom roten ur antalet orienteringsaxlar.

Märk

Om inga toleransvärden programmerades, då är \$A-variablerna inte differentierade nog, för att skilja på toleranserna för de enskilda funktionerna.

Sådana fall kan uppträda när maskin- och settingdata ställer in olika toleranser för kompressorfunktioner, översläpning och orienteringsjämning. Systemvariablerna levererar då det största värdet som uppträder vid de just aktiva funktionerna. När t.ex. en kompressorfunktion med orienteringstoleransen 0,1° och en orienteringsjämning ORISON med 1° är aktiv, levererar variabeln \$AC_OTOL värdet "1". Kopplas orienteringsjämningen från, levererar \$AC_OTOL värdet "0,1".

Läsa utan fördekoderingsstopp

Via följande systemvariabler kan de aktuellt verksamma toleranserna läsas i detaljprogrammet:

- \$P_CTOL
Aktuellt verksam kanalspecifik konturtolerans.
- \$P_OTOL
Aktuellt verksam kanalspecifik orienteringstolerans.
- \$PA_ATOL
Aktuellt verksam axelspecifik konturtolerans.

Randvillkor

De med CTOL, OTOL och ATOL programmerade toleranserna verkar också på funktioner, som indirekt beror på dessa toleranser:

- Begränsningen av kordafelet vid börvärdesberäkningen
- Grundfunktionerna för friformyta-modus

Följande översläpningsfunktioner blir **inte** påverkade av programmeringen av CTOL, OTOL och ATOL:

- Översläpning av orienteringen med OSD
OSD använder ingen tolerans, utan en distans till blockövergången.
- Översläpning med G644
G644 tjänar inte bearbetningen utan optimeringen av verktygsväxlingen och andra rörelser utan bearbetning.
- Översläpning med G645
G645 förhåller sig nästan alltid som G642 och använder på så sätt de programmerade toleranserna. Endast i tangentkontinuerliga blockövergångar med ett böjningshopp, t. ex. en tangential cirkel-rätlinje-övergång, används toleransvärdet från maskindatumet MD33120 \$MA_PATH_TRANS_POS_TOL. Översläpningsvägen kan på dessa ställen nämligen också befinna sig på yttersidan av den programmerade konturen, där många användningar är mindre toleranta. Dessutom räcker som regel en liten, fast inställd tolerans för utjämning av böjningsändringarna, vilka NC-programmeraren inte behöver bekymra sig om.

12.12 Blockbytesbeteende vid aktiv koppling (CPBC)

Med kommandot CPBC anges det blockbyteskriterium som måste vara uppfyllt, för att ett blockbyte genomförs vid aktiv koppling i detaljprogrammet.

Syntax

```
CPBC[<Folgeachse>] = <Kriterium>
```

Betydelse

CPBC:	Blockbyteskriterium vid aktiv koppling	
<Folgeachse>:	Axelbeteckning gör följdaxeln	
<Kriterium>:	Blockbyteskriterium	
	Typ:	STRING
	Värde	Betydelse: Blockbyte sker
	"NOC"	oberoende av kopplingstillstånd
	"IPOSTOP"	vid synkronkörning på börvärdessidan
	"COARSE"	vid synkronkörning "Grov" på ärvärdessidan
	"FINE"	vid synkronkörning "Fin" på ärvärdessidan

Exempel

Programkod

```
; Blockbyte sker vid:
; - koppling till följdaxel X2 == aktiv
; - synkronkörning på börvärdessidan == aktiv
CPBC[X2]="IPOSTOP"
```

Axelkopplingar

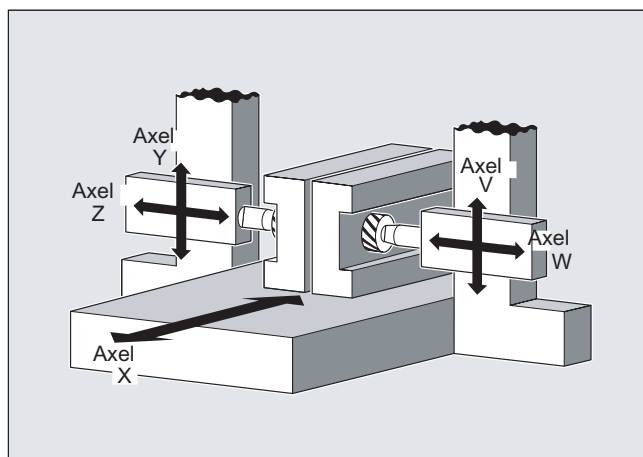
13.1 Medsläpning (TRAILON, TRAILOF)

Vid förflyttning av en definierad styraxel kör medsläpningsaxlar (= följdaxlar) som är tillordnade styraxeln de av denna härledda förflyttningarsvägarna under hänsynstagande till en kopplingsfaktor.

Styraxel och följdaxlar bildar tillsammans ett medsläpningsförband.

Användningsområden

- Förflyttning av en axel genom en simulerad axel. Styraxeln är en simulerad axel och medsläpningsaxeln en reell axel. Därmed kan den reella axeln förflyttas med hänsynstagande till kopplingsfaktorn.
- Tvåsidesbearbetning med 2 medsläpningsförband:
1:a styraxel Y, medsläpningsaxel V
2:a styraxel Z, medsläpningsaxel W



Syntax

```
TRAILON(<Folgeachse>,<Leitachse>,<Koppelfaktor>)
TRAILOF(<Folgeachse>,<Leitachse>,<Leitachse 2>)
TRAILOF(<Folgeachse>)
```

Betydelse

TRAILON:	Kommando för tillkoppling och definiering av ett medsläpningsförband	
	Verkan:	modal
<Folgeachse>:	Parameter 1: Axelbeteckning för medsläpningsaxel Observera: En medsläpningsaxel kan också vara styraxel för ytterligare medsläpningsaxlar. På detta sätt kan olika medsläpningsförband byggas upp.	

13.1 Medsläpning (TRAILON, TRAILOF)

<Leitachse>:	Parameter 2: Axelbeteckning för styraxeln		
<Koppelfaktor>:	Parameter 3: Kopplingsfaktor		
	Kopplingsfaktorn anger det önskade förhållandet mellan vägarna för medsläpningsaxeln och styraxeln: <Koppelfaktor> = väg för medsläpningsaxel / väg för styraxel		
	Typ:	REAL	
	Förinställning:	1	
	Inmatningen av ett negativt värde förorsakar en förflyttningsrörelse hos styr- och medsläpningsaxeln i motsatt riktning. Anges kopplingsfaktorn inte vid programmeringen så gäller automatiskt kopplingsfaktor 1.		
TRAILOF:	Kommando för fränkoppling av ett medsläpningsförband		
	Verkan:	modal	
	TRAILOF med 2 parametrar kopplar bara från kopplingen till den angivna styraxeln: TRAILOF(<Folgeachse>,<Leitachse>)		
	Har en medsläpningsaxel 2 styraxlar, kan anropas för fränkoppling av det båda kopplingarna TRAILOF med 3 parametrar: TRAILOF(<Folgeachse>,<Leitachse>,<Leitachse 2>)		
	Samma resultat levererar programmeringen av TRAILOF utan angivande av en styraxel: TRAILOF(<Folgeachse>)		

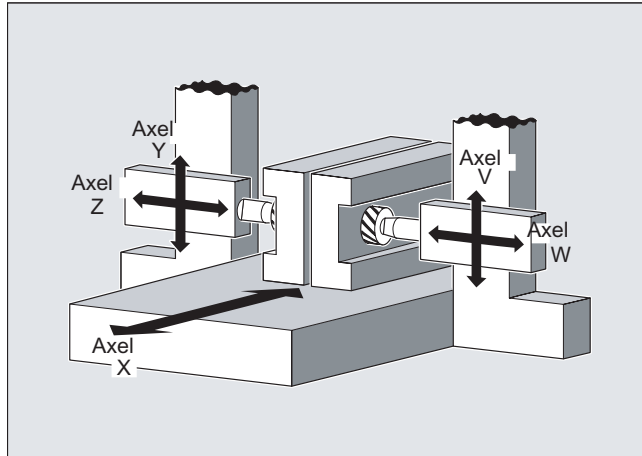
Märk

Medsläpningen sker alltid i baskoordinatsystemet (BKS).

Antalet samtidigt aktiverbara medsläpningsförband begränsas endast av kombinationsmöjligheterna för de på maskinen förefintliga axlarna.

Exempel

Arbetsstycket ska bearbetas tvåsidigt med den visade axelkonstellationen. Därtill bildar du 2 medsläpningsförband.



Programkod	Kommentar
...	
N100 TRAILON(V,Y)	; Tillkoppling av det 1:a medsläpningsförbandet
N110 TRAILON(W,Z,-1)	; Tillkoppling av det 2:a medsläpningsförbandet. Kopplingsfaktor negativ: Medsläpningsaxel går alltid i motsatt riktning till styraxeln.
N120 G0 Z10	; Ansättning av Z- och W-axeln i motsatt axelriktning.
N130 G0 Y20	; Ansättning av Y- och V-axeln i samma axelriktning.
...	
N200 G1 Y22 V25 F200	; Överlagring av en beroende och oberoende rörelse hos medsläpningsaxeln V.
...	
TRAILOF(V,Y)	; Frånkoppling av det 1:a medsläpningsförbandet.
TRAILOF(W,Z)	; Frånkoppling av det 2:a medsläpningsförbandet.

Ytterligare informationer

Axeltyper

Ett medsläpningsförband kan bestå av valfria kombinationer av linjära och roterande axlar. En simulerad axel kan därvid också definieras som styraxel.

Medsläpningsaxlar

En medsläpningsaxel kan samtidigt vara tillordnad till maximalt 2 styraxlar. Tillordningen sker i olika medsläpningsförband.

En medsläpningsaxel kan programmeras med alla till förfogande stående rörelsekommandon (G0, G1, G2, G3, ...). Förutom de oberoende definierade vägarna går medsläpningsaxeln de med kopplingsfaktorerna ur styraxlarna härledda vägarna.

Dynamikbegränsning

Dynamikbegränsningen är beroende av typen av aktivering av medsläpningsförbandet:

- **Aktivering i detaljprogrammet**
Görs aktiveringen i detaljprogrammet och är alla styraxlar programaxlar i den aktiverande kanalen, tas vid förflyttningen av styraxlarna hänsyn till dynamiken för alla medsläpningsaxlar så att ingen medsläpningsaxel blir överbelastad.
Görs aktiveringen i detaljprogrammet med styraxlar, som inte är aktiva som programaxlar i den aktiverande kanalen (\$AA_TYP ≠ 1), tas vid förflyttningen av styraxlarna ingen hänsyn till dynamiken för medsläpningsaxeln. Därigenom kan det för medsläpningsaxlar, med en dynamik lägre än den för kopplingen nödvändiga, leda till en överbelastning.
- **Aktivering i synkronaktion**
Görs aktiveringen i en synkronaktion tas vid förflyttningen av styraxlarna ingen hänsyn till dynamiken för medsläpningsaxlarna. Därigenom kan det för medsläpningsaxlar, med en dynamik lägre än den för kopplingen nödvändiga, leda till en överbelastning.



SE UPP

Axelöverbelastning

Aktiveras ett medsläpningsförband

- i synkronaktioner
- i detaljprogrammet med styraxlar som inte är programaxlar i medsläpningsaxelns kanal, då är det uttryckligen användarens/maskintillverkarens ansvar att planera lämpliga åtgärder så att styraxelns förflyttningsrörelser inte förorsakar en överbelastning av medsläpningsaxlarna.

Kopplingsstatus

Kopplingsstatusen för en axel kan avfrågas i detaljprogrammet med systemvariablerna:

\$AA_COUP_ACT[<Achse>]

Värde	Betydelse
0	Ingen koppling aktiv
8	Medsläpning aktiv

Restvägsindikering av medsläpningsaxeln vid roterande modulo-axlar

Är styr- och medsläpningsaxeln roterande modulo-axlar summeras förflyttningsrörelserna i styraxeln från $n \cdot 360^\circ$ med $n = 1, 2, 3, \dots$, i restvägsindikeringen för medsläpningsaxeln tills kopplingen kopplas från.

Exempel: Programavsnitt med TRAILON och styraxel B och följdaxel C

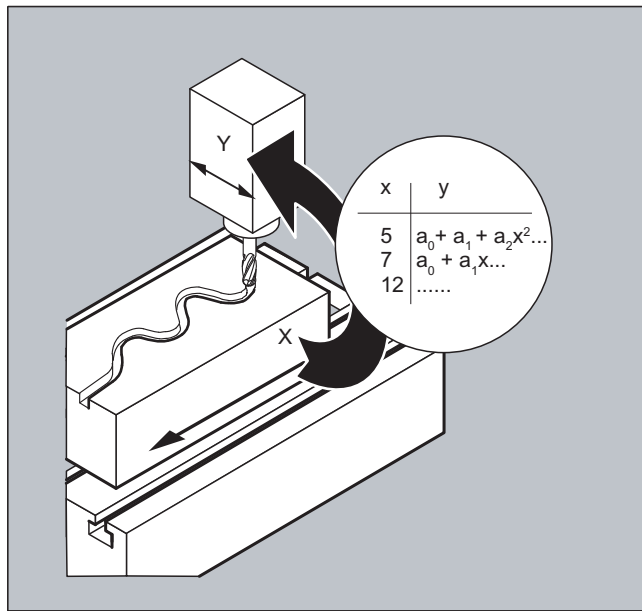
Programkod	Kommentar
TRAILON(C,B,1)	; Koppla till koppling
G0 B0	; Utgångsposition
	; Restvägsindikering till blockbörjan:
G91 B360	; B=360, C=360
G91 B720	; B=720, C=1080
G91 B360	; B=360, C=1440

13.2 Kurvtabeller (CTAB)

Med hjälp av kurvtabeller kan positions- och hastighetsrelationer mellan två axlar (styr- och följdaxel) programmeras. Kurvtabeldefinitionen görs i detaljprogrammet.

Användning

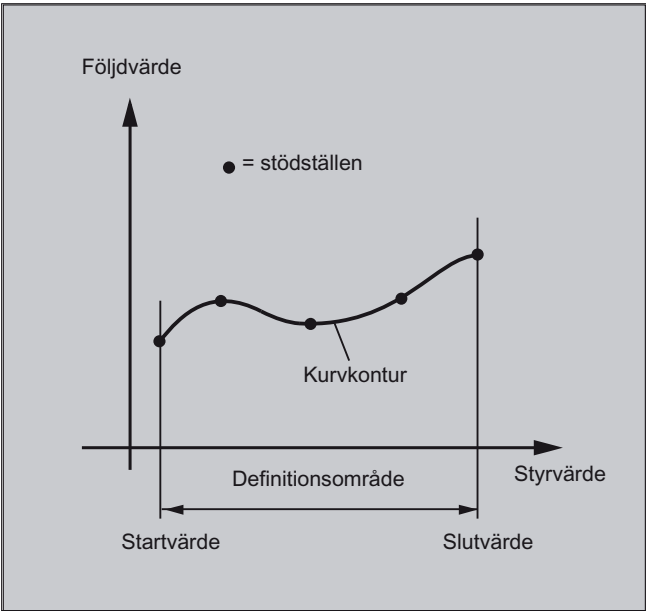
Kurvtabellerna ersätter mekaniska kurvskivor. Kurvtabellen utgör därvid grunden för den axiella styrvärdeskopplingen genom att den skapar det funktionella sambandet mellan styr- och följdvärde: Styrningen beräknar vid motsvarande programmering ett polynom ur till varandra tillordnade positioner för styr- och följdaxel, som motsvarar kurvskivan.



13.2.1 Definiera kurvtabeller (CTABDEF, CATBEND)

En kurvtabell föreställer ett detaljprogram eller ett detaljprogramavsnitt, vilket är kännetecknat av framförställande av CTABDEF och det avslutande kommandot CATBEND.

Inom detta detaljprogramavsnitt tillordnas genom rörelseanvisningar för enskilda positioner för styraxeln entydiga följdaxelpositioner, som tjänar till stödstellen för beräkningen av en kurvkontur i form av ett polynom upp till maximalt 5:e graden.



Förutsättning

För definitionen av kurvtabeller måste genom motsvarande MD-projektering minnesplats vara reserverad (→ maskintillverkaren!).

Syntax

```
CTABDEF(<Folgeachse>,<Leitachse>,<n>,<Periodizität>[,<Speicherort>])
...
CTABEND
```

Betydelse

CTABDEF ():	Början på kurvtabelldefinition	
CTABEND:	Slut på kurvtabelldefinition	
<Folgeachse>:	Axel vars rörelse ska beräknas via kurvtabellen	
<Leitachse>:	Axel som levererar styrvärdena för beräkningen av följdaxelrörelsen	
<n>:	Nummer (ID) på kurvtabellen Numret för en kurvtabel är entydigt och oberoende av minnesplatsen. Det kan inte ligga några tabeller med samma nummer i det statiska och dynamiska NC-minnet.	
<Periodizität>:	Tabellperiodicitet	
	0	Tabellen är inte periodisk (genomarbetas endast en gång, även vid roterande axlar)
	1	Tabellen är periodisk vad beträffar styraxeln
	2	Tabellen är periodisk vad beträffar styraxeln och följdaxel

<Speicherort>:	Uppgift för minnesplatsen (tillval)	
	"SRAM"	Kurvtabellen lagras i det statiska NC-minnet.
	"DRAM"	Kurvtabellen lagras i det dynamiska NC-minnet.
Observera: Om inget värde programmeras för denna parameter, då används den med MD20905 \$MC_CTAB_DEFAULT_MEMORY_TYPE inställda standard-minnesplatsen.		

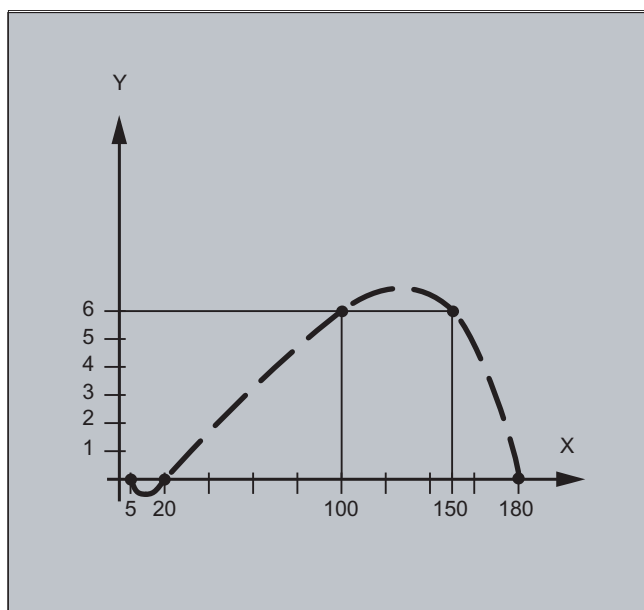
Märk**Överskrivning**

En kurvtabell skrivs över så snart som vid en ny tabelldefinition dess nummer (<n>) används (Undantag: kurvtabellen är aktiv i en axelkoppling eller spärrad med CTABLOCK). **Vid överskrivningen matas ingen tillhörande varning ut!**

Exempel**Exempel 1: Programavsnitt som kurvtabelldefinition**

Ett programavsnitt ska användas oförändrat för definition av en kurvtabell. Det däri uppträdande kommandot till fördekoderingsstopp STOPRE kan stå kvar och blir genast åter aktivt, så snart som programavsnittet inte längre används till tabelldefinition och CTABDEF och CTABEND avlägsnades.

Programkod	Kommentar
...	
CTABDEF (Y,X,1,1)	; Definition av en kurvtabell.
...	
IF NOT (\$P_CTABDEF)	
STOPRE	
ENDIF	
...	
CTABEND	

Exempel 2: Definition av en icke periodisk kurvtabell

Programkod	Kommentar
N100 CTABDEF(Y,X,3,0)	; Början av definitionen för en icke periodisk kurvtabell med nummer 3.
N110 X0 Y0	; 1:a rörelseranvisning, fastlägger startvärde och 1:a stödstället: Styrvärde: 0, följdvärde: 0
N120 X20 Y0	; 2:a stödstället: Styrvärde: 0...20, följdvärde: startvärde...0
N130 X100 Y6	; 3:e stödstället: Styrvärde: 20...100, följdvärde: 0...6
N140 X150 Y6	; 4:e stödstället: Styrvärde: 100...150, följdvärde: 6...6
N150 X180 Y0	; 5:e stödstället: Styrvärde: 150...180, följdvärde: 6...0
N200 CTABEND	; Slut på definitionen. Kurvtabellen skapas i sin interna framställning som polynom av maximalt 5:e graden. Beräkningen av kurvkonturen med de angivna stödställena är beroende av den modalt valda interpoleringstypen (cirkel-, linjär-, spline-interpolering). Detaljprogramtillståndet blir återupprättat före början av definitionen.

Exempel 3: Definition av en periodisk kurvtabell

Definition av en periodisk kurvtabell med nummer 2, styrvärdesområde från 0 till 360, följdaxelrörelse från 0 till 45 och tillbaka till 0:

Programkod	Kommentar
N10 DEF REAL DEPPOS	

Programkod	Kommentar
N20 DEF REAL GRADIENT	
N30 CTABDEF(Y,X,2,1)	; Början på definitionen.
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	; Slut på definitionen.
;Test av kurvan genom koppling från Y till 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)	; Läs tabellfunktionen vid styr- värdet 75.0.
N180 G0 X75 Y=DEPPOS	; Positionering av styr- och följ- daxel.
;Efter tillkoppling av kopplingen är ingen synkronisering av följdaxeln nödvändig.	
N190 LEADON(Y,X,2)	
N200 G1 X110 F1000	
N210 LEADOF(Y,X)	
N220 M30	

Ytterligare informationer

Start- och slutvärde för kurvtabellen

Som startvärde för början av definitionsområdet för kurvtabellen gäller den första uppgiften av sammanhörande axelpositioner (den första rörelseanvisningen) inom kurvtabeldefinitionen. Slutvärdet för definitionsområdet för kurvtabellen bestäms motsvarande genom det sista förflyttningskommandot.

Tillgänglig språkomfattning

Inom definitionen för kurvtabellden står den totala NC-språkomfattningen till förfogande.

Märk

Följande uppgifter är inte tillåtna i kurvtabelldefinitioner:

- Fördekoderingsstopp
- Hopp i styraxelrörelsen (t.ex. vid byte av transformationer)
- Rörelseanvisning bara för följdaxeln
- Rörelsevändning för styraxeln, dvs. styraxelns position måste alltid vara entydig
- CTABDEF- och CTABEND-anvisning i olika programnivåer.

Verkan av modala anvisningar

Samtliga modala verksamma anvisningar, som görs inom kurvtabelldefinitionen, är ogiltiga vid avslutningen av tabelldefinitionen. Detaljprogrammet i vilket tabelldefinitionen görs befinner sig därmed före och efter tabelldefinitionen i samma tillstånd.

Tillordningar till R-parametrar

Tillordningar till R-parametrar inom tabelldefinitionen återställs efter CTABEND.

Exempel:

Programkod	Kommentar
...	
R10=5 R11=20	; R10=5
...	
CTABDEF	
G1 X=10 Y=20 F1000	
R10=R11+5	; R10=25
X=R10	
CTABEND	
...	; R10=5

Aktivering av ASPLINE, BSPLINE, CSPLINE

Aktiveras inom en kurvtabelldefinition CTABDEF ... CTABEND en ASPLINE, BSPLINE eller CSPLINE så bör minst en startpunkt programmeras före denna spline-aktivering. En omedelbar aktivering efter CTABDEF bör undvikas eftersom splinen annars beror av den aktuella axelpositionen före kurvtabelldefinitionen.

Exempel:

Programkod
...
CTABDEF(Y,X,1,0)
X0 Y0
ASPLINE
X=5 Y=10

Programkod

```
X10 Y40
...
CTABEND
```

Upprepad användning av kurvtabeller

Det via kurvtabellen beräknade funktionella sambandet för styr- och följdaxel bibehålls under det valda tabellnumret utöver detaljprogramslutet och utöver POWER OFF, om tabellen är lagrad i det statiska NC-minnet (SRAM).

En tabell som lagrades i det dynamiska minnet (DRAM), raderas vid POWER ON och måste eventuellt skapas ännu en gång.

Den en gång upprättade kurvtabellen låter sig användas till valfria axelkombinationer för styr- och följdaxel och är oberoende av vilka axlar som användes för upprättandet av kurvtabellen.

Överskrivning av kurvtabeller

En kurvtabell skrivs över så snart som vid en förnyad tabelldefinition dess nummer används.

Undantag: En kurvtabell är aktiv i en axelkoppling eller spärrad med CTABLOCK.

Märk

Vid överskrivningen av kurvtabeller matas ingen tillhörande varning ut!

Kurvtabelldefinition aktiv?

Med systemvariabeln `$P_CTABDEF` kan alltid avföras från ett detaljprogram, om en kurvtabelldefinition är aktiv.

Upphävande av kurvtabelldefinition

Detaljprogramavsnittet är efter borttagning av anvisningarna till kurvtabelldefinitionen åter användbart som reellt detaljprogram.

Laddning av kurvtabeller via "Genomarbetning från Extern"

Vid extern genomarbetning av kurvtabeller måste storleken på efterladdningsbufferten (DRAM) via MD18360 `$MN_MM_EXT_PROG_BUFFER_SIZE` väljas så att hela kurvtabelldefinitionen kan lagras samtidigt i efterladdningsbufferten.

Detaljprogrambearbetningen avbryts annars med ett larm.

Hopp i följdaxeln

Beroende på inställningen i maskindatum:

MD20900 `$MC_CTAB_ENABLE_NO_LEADMOTION`

kan hopp i följdaxeln tolereras när rörelse saknas i styraxeln.

13.2.2 Kontrollera om en kurvtabell finns (CTABEXISTS)

Med kommandot CTABEXISTS kan kontrolleras om ett visst kurvtabellnummer finns i NC-minnet.

Syntax

CTABEXISTS (<n>)

Betydelse

CTABEXISTS:	Kontrollerar om kurvtabellen med numret <n> finns i det statiska eller dynamiska NC-minnet	
	0	Tabell existerar inte
	1	Tabell existerar
<n>:	Nummer (ID) på kurvtabellen	

13.2.3 Radera kurvtabell (CTABDEL)

Med CTABDEL kan kurvtabeller raderas.

Märk

Kurvtabeller som är aktiva i en axelkoppling kan inte raderas.

Syntax

```
CTABDEL (<n>)
CTABDEL (<n>, <m>)
CTABDEL (<n>, <m>, <Speicherort>)
CTABDEL ()
CTABDEL (, , <Speicherort>)
```

Betydelse

CTABDEL:	Kommando för radering av kurvtabeller	
<n>:	Nummer (ID) på kurvtabellen som ska raderas Vid radering av ett kurvtabellområde CTABDEL (<n>, <m>) anges med <n> numret för den första kurvtabellen i området.	
<m>:	Vid radering av ett kurvtabellområde CTABDEL (<n>, <m>) anges med <m> numret för den sista kurvtabellen i området. <m> måste vara större än <n>!	
<Speicherort>:	Uppgift för minnesplatsen (tillval) Vid radering utan angivande av minnesplatsen raderas de angivna kurvtablerna i det statiska och dynamiska NC-minnet. Vid radering med angivande av minnesplatsen raderas endast de av de angivna kurvtablerna som ligger i det angivna minnet. De övriga blir bestående.	
	"SRAM"	Radering i det statiska NC-minnet
	"DRAM"	Radering i det dynamiska NC-minnet

Programmeras CTABDEL utan angivande av den kurvtabell som ska raderas, då raderas **alla** kurvtabeller resp. alla kurvtabeller i det angivna minnet:

CTABDEL () :	Raderar alla kurvtabeller i det statiska och dynamiska NC-minnet
CTABDEL (, , "SRAM") :	Raderar alla kurvtabeller i det statiska NC-minnet
CTABDEL (, , "DRAM") :	Raderar alla kurvtabeller i det dynamiska NC-minnet

Märk

Om vid flerfaldig radering CTABDEL (<n>, <m>) eller CTABDEL () åtminstone en av de kurvtabeller som ska raderas är aktiv i en koppling då utförs inte raderingskommandot dvs. **ingen** av de adresserade kurvtabellerna raderas.

13.2.4 Spärra kurvtabeller mot radering och överskrivning (CTABLOCK, CTABUNLOCK)

Kurvtabeller kan skyddas mot oavsiktlig radering och överskrivning genom att sätta spärrar. En spärr som satts kan alltid åter upphävas.

Syntax

Sätta spärr:

```
CTABLOCK (<n>)
CTABLOCK (<n>, <m>)
CTABLOCK (<n>, <m>, <Speicherort>)
CTABLOCK ( )
CTABLOCK ( , , <Speicherort>)
```

Upphäva spärr:

```
CTABUNLOCK (<n>)
CTABUNLOCK (<n>, <m>)
CTABUNLOCK (<n>, <m>, <Speicherort>)
CTABUNLOCK ( )
CTABUNLOCK ( , , <Speicherort>)
```

Betydelse

CTABLOCK:	Kommando för att sätta en spärr mot radering/överskrivning
CTABUNLOCK:	Kommando för att upphäva en spärr mot radering/överskrivning CTABUNLOCK ger de med CTABLOCK spärrade kurvtavellerna åter fria. Kurvtabeller som verkar i en aktiv koppling förblir fortfarande spärrade och kan inte raderas. Spärren med CTABLOCK är upphävd så snart som spärrningen upphävs av den aktiva kopplingen med inaktivering av kopplingen. Därmed kan denna tabell raderas. Ett ytterligare CTABUNLOCK-anrop är inte nödvändigt.

13.2 Kurvtabeller (CTAB)

<n>:	Nummer (ID) på kurvtabelen där spärr ska sättas/upphävas Vid sättandet/upphävandet av spärren i ett kurvtabellområde CTABDEL (<n>, <m>)/CTABUNLOCK (<n>, <m>) anges med <n> numret för den första kurvtabelen i området.
<m>:	Vid sättandet/upphävandet av spärren i ett kurvtabellområde CTABDEL (<n>, <m>)/CTABUNLOCK (<n>, <m>) anges med <m> numret för den sista kurvtabelen i området. <m> måste vara större än <n>!
<Speicherort>:	Uppgift för minnesplatsen (tillval) Vid sättande/upphävande av en spärr utan angivande av minnesplatsen sätts/upphävs spärren för de angivna kurvtabellerna i det statiska och dynamiska NC-minnet. Vid sättande/upphävande av en spärr med angivande av minnesplatsen sätts/upphävs spärren endast för de av de angivna kurvtabellerna som ligger i det angivna minnet. För de övriga sätts/upphävs inte spärren.
	"SRAM" Sätta/upphäva spärr i det statiska NC-minnet
	"DRAM" Sätta/upphäva spärr i det dynamiska NC-minnet

Programmeras CTABLOCK/CTABUNLOCK utan angivande av den kurvtabell där spärren ska sättas/upphävas då sätts/upphävs spärren för **alla** kurvtabeller resp. alla kurvtabeller i det angivna minnet:

CTABLOCK () :	Spärrar alla kurvtabeller i det statiska och dynamiska NC-minnet
CTABLOCK (, , "SRAM") :	Spärrar alla kurvtabeller i det statiska NC-minnet
CTABLOCK (, , "DRAM") :	Spärrar alla kurvtabeller i det dynamiska NC-minnet
CTABUNLOCK () :	Upphäver spärren för alla kurvtabeller i det statiska och dynamiska NC-minnet
CTABUNLOCK (, , "SRAM") :	Upphäver spärren för alla kurvtabeller i det statiska NC-minnet
CTABUNLOCK (, , "DRAM") :	Upphäver spärren för alla kurvtabeller i det dynamiska NC-minnet

13.2.5 Kurvtabeller: Fastställa tabellegenskaper (CTABID, CTABISLOCK, CTABMEMTYP, CTABPERIOD)

Med dessa kommandon kan viktiga egenskaper avfrågas för en kurvtabell (tabellnummer, spärrtillstånd, minnesplats, periodicitet).

Syntax

```
CTABID (<p>)
CTABID (<p>, <Speicherort>)
CTABISLOCK (<n>)
CTABMEMTYP (<n>)
TABPERIOD (<n>)
```


Betydelse

CTABID:	Levererar det tabellnummer , som i det angivna minnet är införd som den <p>:e kurvtabellen. Exempel: CTABID(1, "SRAM") levererar numret för den första kurvtabellen i det statiska minnet. Den första kurvtabellen motsvarar därvid kurvtabellen med det högsta tabellnumret. Observera: Ändras mellan på varandra följande anrop av CTABID ordningsföljden för kurvtabellerna i minnet, t.ex. genom radering av kurvtabeller med CTABDEL, kan CTABID(<p>, . . .) leverera en annan kurvtabell mot tidigare med samma nummer <p>.	
CTABISLOCK:	Ger tillbaka spärrtillståndet för kurvtabellen med nummer <n>:	
	0	Tabellen är inte spärrad
	1	Tabellen är spärrade genom CTABLOCK
	2	Tabellen är spärrad genom aktiv koppling
	3	Tabellen är spärrad genom CTABLOCK och aktiv koppling
	-1	Tabell existerar inte
CTABMEMTYP:	Levererar minnesplatsen för kurvtabellen med nummer <n>:	
	0	Tabell i det statiska NC-minnet
	1	Tabell i det dynamiska NC-minnet
	-1	Tabell existerar inte
CTABPERIOD:	Levererar periodiciteten för kurvtabellen med nummer <n>:	
	0	Tabellen är inte periodisk
	1	Tabellen är periodisk i styraxeln
	2	Tabellen är periodisk i styr- och följdaxel
	-1	Tabell existerar inte
<p>:	Postnummer i minnet	
<n>:	Nummer (ID) på kurvtabellen	
<Speicherort>:	Uppgift för minnesplatsen (tillval)	
	"SRAM"	Statiskt NC-minne
	"DRAM"	Dynamiskt NC-minne
	Observera: Om inget värde programmeras för denna parameter, då används den med MD20905 \$MC_CTAB_DEFAULT_MEMORY_TYPE inställda standard-minnesplatsen.	

13.2.6 Läs kurvtabellvärden (CTABTSV, CTABTEV, CTABTSP, CTABTEP, CTABSSV, CTABSEV, CTAB, CTABINV, CTABTMIN, CTABTMAX)

Följande kurvtabeller kan läsas i detaljprogrammet:

- Följdaxel- och styraxelvärden i början och slutet av en kurvtabell
- Följdaxelvärden i början och slutet av ett kurvsegment
- Följdaxelvärde till ett styraxelvärde

- Styraxelvärde till ett följdaxelvärde
- Minimal- och maximalvärde för följdaxeln
 - i hela definitionsområdet för kurvtabellen eller
 - i ett definierat kurvtabellintervall

Syntax

```
CTABTSV(<n>,<Gradient>[,<Folgeachse>])
CTABTEV(<n>,<Gradient>[,<Folgeachse>])
CTABTSP(<n>,<Gradient>[,<Leitachse>])
CTABTEP(<n>,<Gradient>[,<Leitachse>])
CTABSSV(<Leitwert>,<n>,<Gradient>[,<Folgeachse>])
CTABSEV(<Leitwert>,<n>,<Gradient>[,<Folgeachse>])
CTAB(<Leitwert>,<n>,<Gradient>[,<Folgeachse>,<Leitachse>])
CTABINV(<Folgewert>,<Näherungswert>,<n>,<Gradient>[,<Folgeachse>,<Leitachse>])
CTABTMIN(<n>[,<Folgeachse>])
CTABTMAX(<n>[,<Folgeachse>])
CTABTMIN(<n>,<a>,<b>[,<Folgeachse>,<Leitachse>])
CTABTMAX(<n>,<a>,<b>[,<Folgeachse>,<Leitachse>])
```

Betydelse

CTABTSV:	Läsa följdaxelvärde i början av kurvtabellen nr <n>
CTABTEV:	Läsa följdaxelvärde i slutet av kurvtabellen nr <n>
CTABTSP:	Läsa styraxelvärde i början av kurvtabellen nr <n>
CTABTEP:	Läsa styraxelvärde i slutet av kurvtabellen nr <n>
CTABSSV:	Läsa följdaxelvärde i början av det till det angivna styraxelvärdet (<Leitwert>) tillhörande kurvsegmentet
CTABSEV:	Läsa följdaxelvärde i slutet av det till det angivna styraxelvärdet (<Leitwert>) tillhörande kurvsegmentet
CTAB:	Läsa följdaxelvärdet till det angivna styraxelvärdet (<Leitwert>)
CTABINV:	Läsa styraxelvärdet till det angivna följdaxelvärdet (<Folgewert>)
CTABTMIN:	Bestämma minimalvärde för följdaxeln: • i hela definitionsområdet för kurvtabellen eller • i ett definierat intervall <a> ...
CTABTMAX:	Bestämma maximalvärde för följdaxeln: • i hela definitionsområdet för kurvtabellen eller • i ett definierat intervall <a> ...
<n>:	Nummer (ID) på kurvtabellen
<Gradient>:	I parametern <Gradient> returneras stigningen för kurvtabellfunktionen i den fastställda positionen
<Folgeachse>:	Axel vars rörelse ska beräknas via kurvtabellen (tillval)

<Leitachse>:	Axel som levererar styrvärdena för beräkningen av följdaxelrörelsen (tillval)
<Folgewert>:	Följdaxelvärde för att läsa det tillhörande styraxelvärdet vid CTABINV
<Leitwert>:	Styraxelvärde: - för att läsa det tillhörande följdaxelvärdet vid CTAB eller - för val av kurvsegment vid CTABSSV/CTABSEV
<Näherungswert>:	Tillordningen av ett styraxelvärde till ett följdaxelvärde måste alltid vara entydig vid CTABINV. CTABINV behöver därför ett approximativt värde som parameter för det väntade styraxelvärdet.
<a>:	Undre gräns för styrvärdesintervallet vid CTABTMIN/CTABTMAX
:	Övre gräns för styrvärdesintervallet vid CTABTMIN/CTABTMAX
	Observera: Styrvärdesintervallet <a> ... måste ligga inom definitionsområdet för kurvtabellen.

Exempel

Exempel 1:

Bestämna följdaxel- och styraxelvärden i början och slutet av kurvtabellen samt minimal- och maximalvärde för följdaxeln i hela definitionsområdet för kurvtabellen.

Programkod	Kommentar
N10 DEF REAL STARTPOS	
N20 DEF REAL ENDPOS	
N30 DEF REAL STARTPARA	
N40 DEF REAL ENDPARA	
N50 DEF REAL MINVAL	
N60 DEF REAL MAXVAL	
N70 DEF REAL GRADIENT	
...	
N100 CTABDEF(Y,X,1,0)	; Början på tabelldefinitionen
N110 X0 Y10	; Startposition 1:a tabellsegmentet
N120 X30 Y40	; Slutposition 1:a tabellsegmentet = startposition 2:a tabellsegmentet
N130 X60 Y5	; Slutposition 2:a tabellsegmentet = ...
N140 X70 Y30	
N150 X80 Y20	
N160 CTABEND	; Slut på tabelldefinitionen.
...	
N200 STARTPOS=CTABTSV(1,GRADIENT)	; Följdaxelvärde vid kurvtabellens början = 10
N210 ENDPOS=CTABTEV(1,GRADIENT)	; Följdaxelvärde vid kurvtabellens slut = 20
N220 STARTPARA=CTABTSP(1,GRADIENT)	; Styraxelvärde vid kurvtabellens början = 0
N230 ENDPARA=CTABTEP(1,GRADIENT)	; Styraxelvärde vid kurvtabellens slut = 80
N240 MINVAL=CTABTMIN(1)	; Minimalvärde för följdaxeln vid Y=5
N250 MAXVAL=CTABTMAX(1)	; Maximalvärde för följdaxeln vid Y=40

Exempel 2:

Bestämning av följdaxelvärdena i början och slutet av det kurvsegment som tillhör styraxelvärdet $X=30$.

Programkod	Kommentar
N10 DEF REAL STARTPOS	
N20 DEF REAL ENDPOS	
N30 DEF REAL GRADIENT	
...	
N100 CTABDEF(Y,X,1,0)	; Början på tabelldefinitionen.
N110 X0 Y0	; Startposition 1:a tabellsegmentet
N120 X20 Y10	; Slutposition 1:a tabellsegmentet = startposition 2:a tabellsegmentet
N130 X40 Y40	; Slutposition 2:a tabellsegmentet = ...
N140 X60 Y10	
N150 X80 Y0	
N160 CTABEND	; Slut på tabelldefinitionen.
...	
N200 STARTPOS=CTABSSV(30.0,1,GRADIENT)	; Startposition Y i det 2:a segmentet = 10
N210 ENDPOS=CTABSEV(30.0,1,GRADIENT)	; Slutposition Y i det 2:a segmentet = 40

Ytterligare informationer**Användning i synkronaktioner**

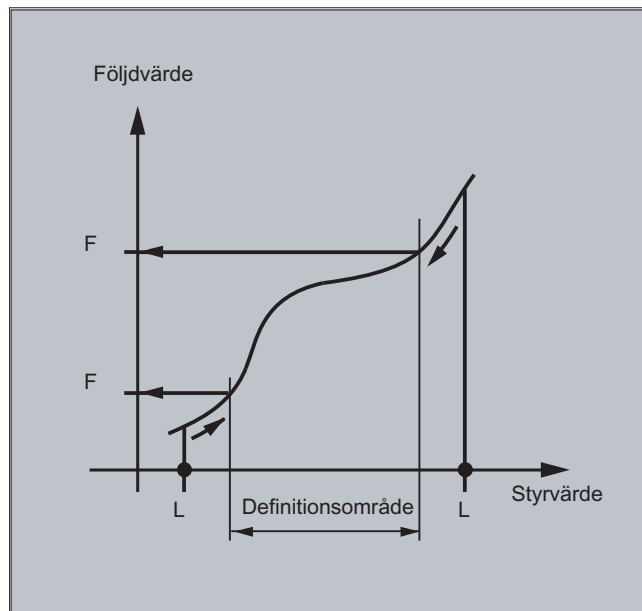
Alla kommandon för läsning av kurvtabellvärden kan också användas i synkronaktioner (se även Kapitel "Rörelsesynkronaktioner").

Vid användning av kommandona CTABINV, CTABTMIN och CTABTMAX ska ges akt på att:

- vid tidpunkten för utförandet tillräcklig NC-prestation är disponibel eller
- före anropet avfråga antalet segment i kurvtabellen för att eventuellt kunna dela upp den ifrågavarande tabellen

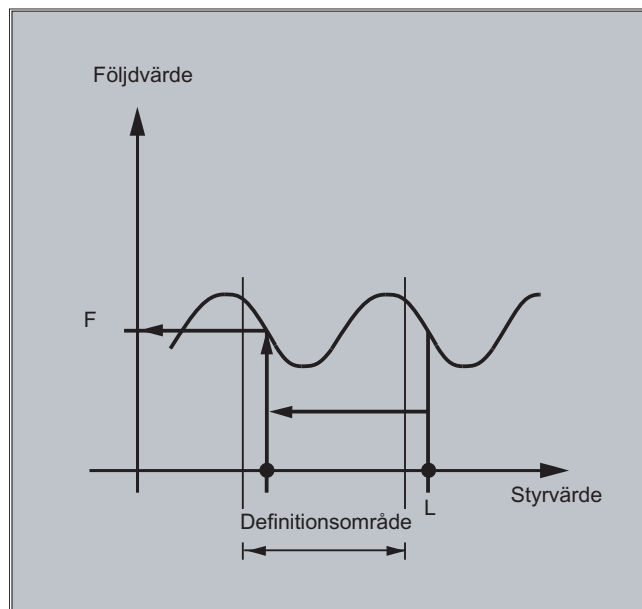
CTAB vid icke periodiska kurvtabeller

Ligger det angivna <Leitwert> utanför definitionsområdet, matas den övre resp. undre gränsen ut som följdvärde:



CTAB vid periodiska kurvtabeller

Ligger det angivna <Leitwert> utanför definitionsområdet, värderas styrvärdet Modulo i definitionsområdet och det motsvarande följdvärdet matas ut:



Approximativt värde för CTABINV

Kommandot CTABINV behöver ett approximativt värde för det väntade styrvärdet. CTABINV returnerar det styrvärde som ligger närmast det approximativa värdet. Det approximativa värdet kan t.ex. vara styrvärdet från den föregående interpolatortakten.

Stigning hos kurvtabellfunktionen

Utmatningen av stigningen (<Gradient>) gör det möjligt att beräkna hastigheten för styr- eller följdaxeln i den motsvarande positionen.

Angivande av styr- eller följdaxel

Tillvalsuppgiften om styr- och/eller följdaxel är viktig om styr- och följdaxel är projekterade i olika längdenheter.

CTABSSV, CTABSEV

Kommandona CTABSSV och CTABSEV är i följande fall **inte** lämpliga att, avfråga programmerade segment:

- Cirklar eller evolventer är programmerade.
- Faser resp. rundningar med CHF/RND är aktiva.
- Översläpning med G643 är aktiv.
- NC-block-kompression med COMPON/COMPCURV/COMPCAD är aktiv.

13.2.7**Kurvtabeller: Kontrollera resursanvändning (CTABNO, CTABNOMEM, CTABFNO, CTABSEGID, CTABSEG, CTABFSEG, CTABMSEG, CTABPOLID, CTABPOL, CTABFPOL, CTABMPOL)**

Med dessa kommandon har programmeraren möjligheten att informera sig aktuellt över beläggningen av resurserna för kurvtabeller, tabellsegment och polynom.

Syntax

```
CTABNO
CTABNOMEM(<Speicherort>)
CTABFNO(<Speicherort>)
CTABSEGID(<n>,<Speicherort>)
CTABSEG(<Speicherort>,<Segmentart>)
CTABFSEG(<Speicherort>,<Segmentart>)
CTABMSEG(<Speicherort>,<Segmentart>)
CTABPOLID(<n>)
CTABPOL(<Speicherort>)
CTABFPOL(<Speicherort>)
CTABMPOL(<Speicherort>)
```

Betydelse

CTABNO:	Bestämma det totala antalet definierade kurvtabeller (i det statiska och dynamiska NC-minnet)
CTABNOMEM:	Bestämma antalet definierade kurvtabeller i den angivna <Speicherort>
CTABFNO:	Bestämma antalet ännu möjliga kurvtabeller i den angivna <Speicherort>
CTABSEGID:	Bestämma antalet kurvsegment av den angivna <Segmentart> som används av kurvtabellen med nummer <n>

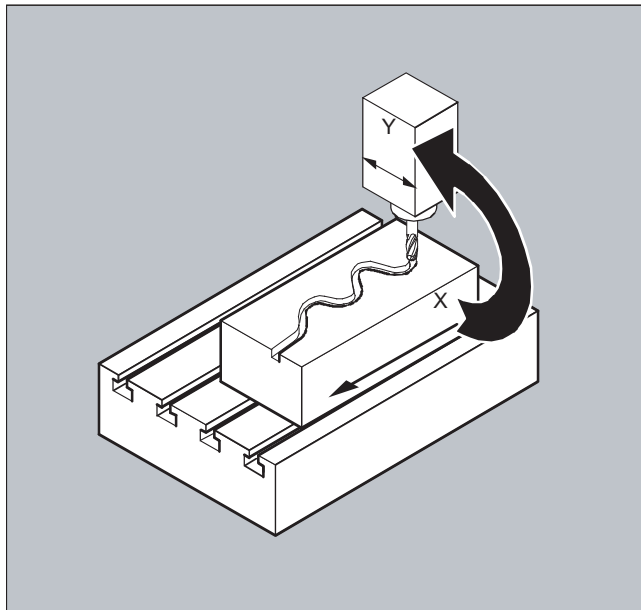
CTABSEG:	Bestämma antalet använda kurvsegment av den angivna <Segmentart> i den angivna <Speicherort>	
CTABFSEG:	Bestämma antalet ännu möjliga kurvsegment av den angivna <Segmentart> i den angivna <Speicherort>	
CTABMSEG:	Bestämma antalet maximalt möjliga kurvsegment av den angivna <Segmentart> i den angivna <Speicherort>	
CTABPOLID:	Bestämma antalet kurvpolytom som används av kurvtabellen med nummer <n>	
CTABPOL:	Bestämma antalet använda kurvpolytom i den angivna <Speicherort>	
CTABFPOL:	Bestämma antalet ännu möjliga kurvpolytom i den angivna <Speicherort>	
CTABMPOL:	Bestämma antalet maximalt möjliga kurvpolytom i den angivna <Speicherort>	
<n>:	Nummer (ID) på kurvtabellen	
<Speicherort>:	Uppgift för minnesplatsen (tillval)	
	"SRAM"	Statiskt NC-minne
	"DRAM"	Dynamiskt NC-minne
	Observera: Om inget värde programmeras för denna parameter, då används den med MD20905 \$MC_CTAB_DEFAULT_MEMORY_TYPE inställda standard-minnesplatsen.	
<Segmentart>:	Uppgift för segmenttypen (tillval)	
	"L"	Linjärt segment
	"P"	Polynomsegment
	Observera: Om inget värde programmeras för denna parameter, då matas summan av linjär- och polynomsegment ut.	

13.3 Axial styrvärdeskoppling (LEADON, LEADOF)

Märk

Denna funktion står för SINUMERIK 828D inte till förfogande!

Vid en axial styrvärdeskoppling förflyttas en styr- och en följdaxel synkront. Därvid är respektive position för följdaxeln via en kurvtabell resp. ett därur beräknat polynom entydigt tillordnat en - ev. simulerad - position för styraxeln.



Styraxel heter den axel som levererar ingångsvärdena för kurvtabellen. **Följdaxel** heter den axel som antar de via kurvtabellen beräknade positionerna.

Är- och börvärdeskoppling

Som styrvärden alltså utgångsvärden för positionsbestämning för följdaxeln kan användas:

- Ärvärden för styraxelpositionen: Ärvärdeskoppling
- Börvärden för styraxelpositionen: Börvärdeskoppling

Styrvärdeskopplingen gäller alltid i baskoordinatsystemet.

För upprättande av kurvtabeller se kapitel "Kurvtabeller".

Syntax

```
LEADON(<Folgeachse>,<Leitachse>,<n>)
LEADOF(<Folgeachse>,<Leitachse>)
```

eller fränkoppling utan angivande av styraxeln:

```
LEADOF(<Folgeachse>)
```

Styrvärdeskopplingen kan kopplas till och från både från detaljprogrammet och även under rörelsen utifrån synkronaktioner.

Betydelse

LEADON:	Koppla till styrvärdeskoppling
LEADOF:	Koppla från styrvärdeskoppling
<Folgeachse>:	Följdaxel
<Leitachse>:	Styraxel
<n>:	Kurvtabell-nummer
\$SA_LEAD_TYPE:	Omkoppling mellan bör- och ärvärdeskoppling

Koppla från styrvärdeskoppling, LEADOF

Med frångkopplingen av styrvärdeskopplingen blir följdaxeln åter till normal kommandoaxel!

Axial styrvärdeskoppling och olika drifttillstånd, RESET

Beroende på inställningen i maskindatum kopplas styrvärdeskopplingar från med RESET.

Exempel Styrvärdeskoppling från synkronaktion

Vid en pressanläggning ska en vanlig mekanisk koppling mellan en styraxel (stämpelaxel) och axlarna till ett transfersystem bestående av transferaxlar och hjälpaxlar ersättas av ett elektroniskt kopplingssystem.

Det demonstrerar hur vid en pressanläggning ett mekaniskt transfersystem ersätts av ett elektroniskt transfersystem. Kopplings- och urkopplingsförlopp är realiserade som **statiska synkronaktioner**.

Från styraxeln LW (stämpelaxel) styrs transferaxlar och hjälpaxlar som följdaxlar definierat via kurvtabeller.

Följdaxlar

X matnings- resp. längsaxel
YL stängning- resp. tväraxel
ZL lyftaxel
U valsmatning, hjälpaxel
V rikthuvud, hjälpaxel
W fettning, hjälpaxel

Aktioner

Som aktioner uppträder i synkronaktionerna t.ex.:

- Tillkoppling, LEADON(<Folgeachse>,<Leitachse>,<Kurventabellen-Nummer>)
- Frångkoppling, LEADOF(<Folgeachse>,<Leitachse>)
- Ärvärdesinställning, PRESETON(<Achse>,<Wert>)
- Sätta flagga, \$AC_MARKER[i] =<Wert>
- Kopplingstyp: reellt/virtuellt styrvärde
- Uppsökning av axelpositioner, POS[<Achse>]=<Wert>

Villkor

13.3 Axial styrvärdeskoppling (LEADON, LEADOF)

Som villkor utvärderas digitala snabba ingångar, realtidsvariabler \$AC_MARKER och positionsjämförelser förbundna med den logiska operatoren AND.

Märk

I det följande exemplet användas radbyte, indragningar och **fet** skrift uteslutande för att underlätta läsningen av programmeringen. För styrningen är allt som står under ett radnummer enradigt.

Kommentar

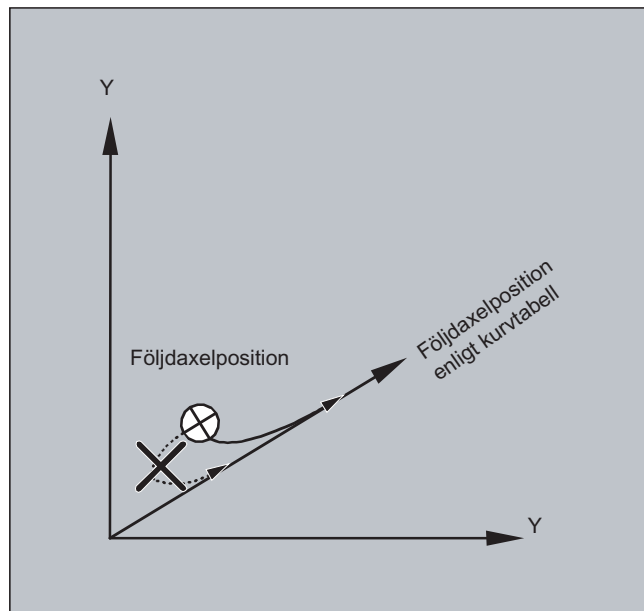
Programkod	Kommentar
	; Definierar samtliga statiska synkronaktioner.
	; ****Återställa flagga
N2 \$AC_MARKER[0]=0 \$AC_MARKER[1]=0 \$AC_MARKER[2]=0 \$AC_MARKER[3]=0 \$AC_MARKER[4]=0 \$AC_MARKER[5]=0 \$AC_MARKER[6]=0 \$AC_MARKER[7]=0	
	; **** E1 0=>1 koppling transfer TILL
N10 IDS=1 EVERY (\$A_IN[1]==1) AND (\$A_IN[16]==1) AND (\$AC_MARKER[0]==0) DO LEADON(X,LW,1) LEADON(YL,LW,2) LEADON(ZL,LW,3) \$AC_MARKER[0]=1	
	; **** E1 0=>1 koppling valsmatning TILL
N20 IDS=11 EVERY (\$A_IN[1]==1) AND (\$A_IN[5]==0) AND (\$AC_MARKER[5]==0) DO LEADON(U,LW,4) PRESETON(U,0) \$AC_MARKER[5]=1	
	; **** E1 0->1 koppling rikthuvud TILL
N21 IDS=12 EVERY (\$A_IN[1]==1) AND (\$A_IN[5]==0) AND (\$AC_MARKER[6]==0) DO LEADON(V,LW,4) PRESETON(V,0) \$AC_MARKER[6]=1	
	; **** E1 0->1 koppling fettning TILL
N22 IDS=13 EVERY (\$A_IN[1]==1) AND (\$A_IN[5]==0) AND (\$AC_MARKER[7]==0) DO LEADON(W,LW,4) PRESETON(W,0) \$AC_MARKER[7]=1	
	; **** E2 0=>1 koppling FRÅN
N30 IDS=3 EVERY (\$A_IN[2]==1) DO LEADOF(X,LW) LEADOF(YL,LW) LEADOF(ZL,LW) LEADOF(U,LW) LEADOF(V,LW) LEADOF(W,LW) \$AC_MARKER[0]=0 \$AC_MARKER[1]=0 \$AC_MARKER[3]=0 \$AC_MARKER[4]=0 \$AC_MARKER[5]=0 \$AC_MARKER[6]=0 \$AC_MARKER[7]=0 N110 G04 F01 N120 M30	

Beskrivning

Styrvärdeskopplingen kräver synkroniseringen av styr- och följdaxel. Denna synkronisering kan endast uppnås när följdaxeln vid tillkopplingen av styrvärdeskopplingen står inom toleransområdet för den kurvkontur som beräknats ur kurvtabellen.

Toleransområdet för följdaxelns läge är definierat via maskindatum MD 37200:
COUPLE_POS_POL_COARSE A_LEAD_TYPE .

Befinner sig följdaxeln vid tillkopplingen av styrvärdeskopplingen ännu inte på den motsvarande positionen, upprättas synkronkörningen automatiskt så snart som det beräknade positionsbörvärdet för följdaxeln närmar sig den verkliga följdaxelpositionen. Följdaxeln förflyttas därvid under synkroniseringsförloppet i den riktning som är definierad genom börhastigheten för följdaxeln (beräknas ur styraxelhastighet och enligt kurvtabell CTAB).

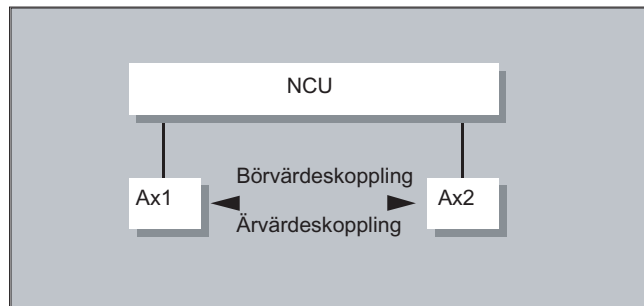


Ingen synkronkörning

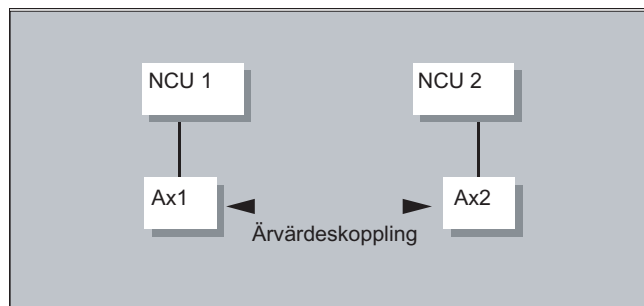
Avlägsnar sig den beräknade följdaxelbörpositionen vid tillkopplingen av styrvärdeskopplingen från den aktuella följdaxelpositionen upprättas ingen synkronkörning.

Är- och börvärdeskoppling

Börvärdeskopplingen levererar i jämförelse med ärvärdeskopplingen en bättre synkronkörning mellan styr- och följdaxel och är därför standardmässigt förinställd.



Börvärdeskoppling är endast möjlig när styr- och följdaxel interpoleras av samma NCU. Vid en extern styraxel kan följdaxeln endast kopplas via ärvärden till styraxeln.



En **omkoppling** är möjlig via settingdatum \$SA_LEAD_TYPE.

Omkopplingen mellan är- och börvärdeskoppling bör alltid göras när följdaxeln står still. Ty endast vid stillestånd synkroniseras på nytt efter omkopplingen.

Användningsexempel

Läsandet av ärvärdena kan inte göras felfritt vid stora maskinvibrationer. Vid användningen av styrvärdeskopplingen i presstransferen kan det därför vara nödvändigt i arbetsmomenten med de största vibrationerna att koppla om från ärvärdeskoppling till börvärdeskoppling.

Styrvärdessimulering vid börvärdeskoppling

Via maskindatum låter sig interpolatorn för styraxeln skiljas från servon. Därmed kan vid börvärdeskoppling börvärden skapas utan verklig rörelse av styraxeln.

De via börvärdeskopplingen skapade styrvärdena är läsbara från följande variabler vid användning t.ex. i synkronaktioner:

- \$AA_LEAD_P	Styrvärde position
- \$AA_LEAD_V	Styrvärde hastighet

Skapa styrvärden

Styrvärden kan valfritt skapas med andra själv programmerade beteenden. De så skapade styrvärdena skrivs i variabeln

- \$AA_LEAD_SP	Styrvärde position
- \$AA_LEAD_SV	Styrvärde hastighet

och läses ur dem. För användning av dessa variabler måste settingdatum \$SA_LEAD_TYPE = 2 ställas in.

Status för kopplingen

I NC-detaljprogrammet kan du avfråga kopplingsstatus med följande systemvariabler:

\$AA_COUP_ACT[Achse]
 0: Ingen koppling aktiv
 16: Styrvärdeskoppling aktiv

Status-förvaltning vid synkronaktioner

Bryt- och kopplingsförlopp förvaltas via realtidsvariabler:

\$AC_MARKER[i] = n
 förvaltas med:
 i Flagg-nummer
 n Statusvärde

13.4 Elektronisk växel (EG)

Med hjälp av funktionen "Elektronisk växel" är det möjligt att styra rörelsen hos en **följdaxel** enligt linjärt rörelseblock beroende av upp till fem **styraxlar**. Sambandet mellan styraxlarna och följdaxeln är definierat för var styraxel genom kopplingsfaktorn.

Den beräknade följdaxel-rörelseandelen bildas genom addition av de enskilda styraxel-rörelseandelarna multiplicerade med respektive kopplingsfaktor. Vid aktiveringen av ett EG-axelförband kan synkroniseringen av följdaxeln på en definierad position ombesörjas. Ett växelförband kan:

- definieras,
- kopplas till,
- kopplas från,
- raderas

från detaljprogrammet.

Följdaxelrörelsen kan valfritt härledas ur

- börvärden för styraxlarna samt
- ärvärden för styraxlarna.

Som komplettering kan också icke linjära samband mellan styraxlarna och följdaxeln realiserar via **kurvtabeller** (se Kapitel Banbeteende). Elektroniska växlar kan kaskaderas dvs. följdaxeln till en elektronisk växel kan vara styraxel för en ytterligare elektronisk växel.

13.4.1 Definiera elektronisk växel (EGDEF)

Ett EG-axelförband fastläggs genom angivande av följdaxeln och minst en, dock högst fem styraxlar med respektive kopplingstyp.

Förutsättning

Förutsättning för en definition av ett EG-axelförband:

För följdaxeln får ännu ingen axelkoppling vara definierad (ev. måste en som finns dessförinnan raderas med EGDEL).

Syntax

```
EGDEF (Folgeachse, Leitachse1, Kopplungstyp1, Leitachse2, Kopplungstyp2, .
..)
```

Betydelse

EGDEF:	Definition av en elektronisk växel
Folgeachse:	Axel som påverkas av styraxlar

13.4 Elektronisk växel (EG)

Leitachse1 Leitachse5	Axlar som påverkar följdaxeln	
Kopplungstyp1 Kopplungstyp5	Kopplingstyp Kopplingstypen måste inte vara lika för alla styraxlar och ska därför anges separat för varje styraxel.	
	Värde:	Betydelse:
	0	Följdaxeln blir påverkad av ärvvärdet för den motsvarande styraxeln.
	1	Följdaxeln blir påverkad av börvärdet för den motsvarande styraxeln.

Märk

Kopplingsfaktorerna förbeläggs med noll vid definitionen av EG-kopplingsförbandet.

Märk

EGDEF utlöser fördekoderingsstopp. Växeldefinitionen med EGDEF ska också användas oförändrad när vid systemen en eller flera styraxlar inverkar på följdaxeln via **kurvtabell**.

Exempel

Programkod	Kommentar
EGDEF(C,B,1,Z,1,Y,1)	; Definition av ett EG-axelförband. Styraxlarna B, Z, Y påverkar följdaxeln C via börvärdet.

13.4.2 Koppla till elektronisk växel (EGON, EGONSYN, EGONSYNE)

För tillkopplingen av ett EG-axelförband existerar 3 varianter.

Syntax

Variant 1:

EG-axelförbandet kopplas utan synkronisering selektivt till med:

```
EGON(FA, "Satzwechselmodus", LA1, Z1, N1, LA2, Z2, N2, ..., LA5, Z5, N5)
```

Variant 2:

EG-axelförbandet kopplas med synkronisering selektivt till med:

```
EGONSYN(FA, "Satzwechselmodus", SynPosFA, [, LAi, SynPosLAI, Zi, Ni])
```

Variant 3:

EG-axelförbandet kopplas med synkronisering selektivt till och uppsökningsmode föreskrivs med:

```
EGONSYN(FA, "Satzwechselmodus", SynPosFA, [, LAi, SynPosLAI, Zi, Ni])
```

Betydelse

Variant 1:

FA	Följdaxel
Satzwechselmodus:	Följande moder kan användas:
	"NOC" Blockbyte görs omedelbart
	"FINE" Blockbyte görs vid "Synkronkörning fin"
	"COARSE" Blockbyte görs vid "Synkronkörning grov"
	"IPOSTOP" Blockbyte görs vid börvärdessidig synkronkörning
LA1, ... LA5	Styraxlar
Z1, ... Z5	Täljare för kopplingsfaktorn i
N1, ... N5	Nämnare för kopplingsfaktorn i Kopplingsfaktor i = täljare i/nämnare i

Endast de styraxlar får programmeras som dessförinnan har specificerats med EGDEF. Minst en styraxel måste vara programmerad.

Variant 2:

FA	Följdaxel
Satzwechselmodus:	Följande moder kan användas:
	"NOC" Blockbyte görs omedelbart
	"FINE" Blockbyte görs vid "Synkronkörning fin"
	"COARSE" Blockbyte görs vid "Synkronkörning grov"
	"IPOSTOP" Blockbyte görs vid börvärdessidig synkronkörning
[, LAi, SynPosLAI, Zi, Ni]	(inte skriva hakparenteser) Minst 1, max. 5 följer av:
LA1, ... LA5	Styraxlar
SynPosLAI	Synkronposition för den i:e styraxeln
Z1, ... Z5	Täljare för kopplingsfaktorn i
N1, ... N5	Nämnare för kopplingsfaktorn i Kopplingsfaktor i = täljare i/nämnare i

Endast de styraxlar får programmeras som dessförinnan har specificerats med EGDEF. Genom de programmerade "synkronpositionerna" för följdaxeln (SynPosFA) och för styraxlarna (SynPosLA) definieras positioner, i vilka kopplingsförbandet gäller som *synkront* gilt. Såvida den elektriska växeln inte befinner sig i synkrontillstånd vid tillkopplingen, går följdaxeln till sin definierade synkronposition.

Variant 3:

Parametrarna motsvarar de i variant 2 och dessutom:

Anfahrmodus:	Följande moder kan användas:	
	"NTGT"	Uppsöka nästa tandlucka tidsoptimerat
	"NTGP"	Uppsöka nästa tandlucka vägoptimerat
	"ACN"	Förflytta roterande axel i negativ rotationsriktning absolut
	"ACP"	Förflytta roterande axel i positiv rotationsriktning absolut
	"DCT"	Tidsoptimerat till den programmerade synkronpositionen
	"DCP"	Vägoptimerat till den programmerade synkronpositionen

Variant 3 har endast inverkan på Modulo-följdaxlar, som är kopplade till Modulo-styraxlar. Tidsoptimering tar hänsyn till hastighetsgränserna för följdaxeln.

Ytterligare informationer**Beskrivning av tillkopplingsvarianterna**

Variant 1:

Positionerna för styraxlarna samt följdaxeln vid tidpunkten för tillkopplingen sparas som "Synkronpositioner". "Synkronpositionerna" kan läsas med systemvariabeln \$AA_EG_SYN .

Variant 2:

När moduloaxlar finns i kopplingsförbandet reduceras deras positionsvärden modulo. Därmed är garanterat att nästa möjliga synkronposition uppsöks (så kallad *relativ synkronisering*, t.ex. nästa tandlucka). När för följdaxeln inte "Frigivning följdaxelöverlagring" gränssnittssignal DB(30 +axelnummer), DBX 26 Bit 4 är given, körs inte till synkronpositionen. I stället stoppas programmet vid EGONSYN-blocket och det självutlösande larmet 16771 meddelas, ända tills den ovan nämnda signalen har ställts in.

Variant 3:

Tandavståndet (grader) resulterar ur: $360 \cdot Z_i/N_i$. För den händelse att följdaxeln står vid tidpunkten för anropet, levererar vägoptimerat samma beteende som tidsoptimerat.

Vid en följdaxel som redan förflyttar sig synkroniseras med NTGP oberoende av den aktuella hastigheten för följdaxeln till nästa tandlucka. Vid en följdaxel som redan förflyttar sig synkroniseras med NTGT beroende av den aktuella hastigheten för följdaxeln till nästa tandlucka. Axeln blir därtill ev. också bromsad.

Kurvtabeller

Används för en av styraxlarna en **kurvtabell** så måste:

- Ni nämnaren för kopplingsfaktorn för linjära kopplingar sätts på 0. (nämnaren 0 skulle vara otillåten för linjära kopplingar). Nämnaren noll är för styrningen kännetecknet att
- Zi ska interpretas som nummer för den kurvtabell som ska användas. Kurvtabellen med det angivna numret måste redan vara definierad vid tidpunkten för tillkopplingen.
- LAi Angivandet av styraxeln motsvarar styraxeluppgiften vid koppling via kopplingsfaktor (linjär koppling).

Ytterligare anvisningar över användningen av kurvtabeller och kaskaderingen av elektroniska växlar och deras synkronisering finns i:

Litteratur:

Funktionshandbok Specialfunktioner; Axelkopplingar och ESR (M3), Kapitel "Medsläpning och styrvärdeskoppling".

Beteende hos den elektroniska växeln vid Power On, RESET, driftslägesbyte, blocksökning

- Efter Power On är **ingen** koppling aktiv.
- Aktiva kopplingar bibehålls utöver RESET och driftslägesbyte.
- Vid blocksökning blir kommandon för koppling, radering, definiering av den elektroniska växeln inte utförda och inte samlade utan förbigångna.

Systemvariabler för den elektroniska växeln

Med hjälp av systemvariablerna för den elektroniska växeln kan detaljprogrammet fastställa aktuella tillstånd för ett EG-axelförband och ev. reagera därpå.

Systemvariablerna för den elektroniska växeln är kännetecknade på följande sätt:

\$AA_EG_ ...

eller

\$VA_EG_ ...

Litteratur:

Handbok för systemvariablerna

13.4.3 Koppla från elektronisk växel (EGOFS, EGOFC)

För fränkopplingen av ett aktivt EG-axelförband existerar 3 varianter.

Programmering

Variant 1:

Syntax	Betydelse
EGOFS (Folgeachse)	Den elektroniska växeln kopplas från. Följdaxeln bromsas till stillstånd. Anropet utlöser fördekoderingsstopp.

Variant 2:

Syntax	Betydelse
EGOFS (Folgeachse, Leitachse1, ..., Leitachse5)	Denna parametring av kommandot tillåter att selektivt undanröja inflytandet av enskilda styraxlar på följdaxelns rörelse.

Minst en styraxel måste anges. Inflytandet av de angivna styraxlarna på följdaxeln kopplas målinriktat från. Anropet utlöser fördekoderingsstopp. Förblir fortfarande aktiva styraxlar så fortsätter följdaxeln under deras inflytande. Är alla styraxelinflytanden fränkopplade på detta sätt så bromsas följdaxeln till stillstånd.

Variant 3:

Syntax	Betydelse
EGOFC (Folgespindel)	Den elektroniska växeln kopplas från. Följdspindeln fortsätter att gå med det(den) vid tidpunkten för fränkopplingen aktuella varvtalet/hastigheten. Anropet utlöser fördekoderingsstopp.

Märk

Denna variant är tillåten endast för spindlar.

13.4.4 Radera definition för en elektronisk växel (EGDEL)

Ett EG-axelförband måste vara fränkopplat, innan dess definition kan raderas.

Programmering

Syntax	Betydelse
EGDEL (Folgeachse)	Kopplingsdefinitionen för axelförbandet raderas. Det blir åter möjligt att på nytt definiera ytterligare axelförband med EGDEF tills det maximala antalet samtidigt aktiverade axelförband är uppnått. Anropet utlöser fördekoderingsstopp.

13.4.5 Varvmatning (G95) / elektronisk växel (FPR)

Med FPR-kommandot kan också följdaxeln för en elektronisk växel anges som matningsbestämmande axel för varvmatningen. För detta fall gäller följande beteende:

- Matningen är beroende av börhastigheten för följdaxeln till den elektroniska växeln.
- Börhastigheten beräknas ur hastigheterna för styrspindlarna och Modulo-styraxlarna (som inte är banaxlar) och deras tillordnade kopplingsfaktorer.
- Hastighetsandelarna för linjära resp. icke Modulo-styraxlar och överlagrade rörelser hos följdaxeln tas det ingen hänsyn till.

13.5 Synkronspindel

I synkron drift finns det en styrspindel (LS) och en följdspindel (FS), det så kallade **synkronspindelparet**. Följdspindeln följer vid aktiv koppling (synkron drift) rörelserna hos styrspindeln i enlighet med det fastlagda funktionssambandet.

Synkronspindelparen låter sig för varje maskin både med hjälp av kanalspecifika maskindata fast projekteras eller via CNC-detaljprogrammet användningsspecifikt definieras. Per NC-kanal kan upp till 2 synkronspindelpar drivas samtidigt.

Kopplingen kan

- definieras resp. ändras
- kopplas till
- kopplas från
- raderas

från detaljprogrammet.

Därutöver kan beroende av softwareversion

- väntas på synkronkörningsvillkor
- att blockbytesbeteendet förändras
- kopplingstyp antingen börvärdeskoppling eller ärvärdeskoppling väljas eller vinkelförskjutningen mellan styr- och följdspindel föreskrivas
- vid tillkopplingen av kopplingen ett tidigare programmering för följdspindeln övertas
- antingen en uppmätt eller en redan känd synkronkörningsavvikelse korrigeras

.

13.5.1 Synkronspindel: Programmering (COUPDEF, COUPDEL, COUPON, COUPONC, COUPOF, COUPOFS, COUPRES, WAITC)

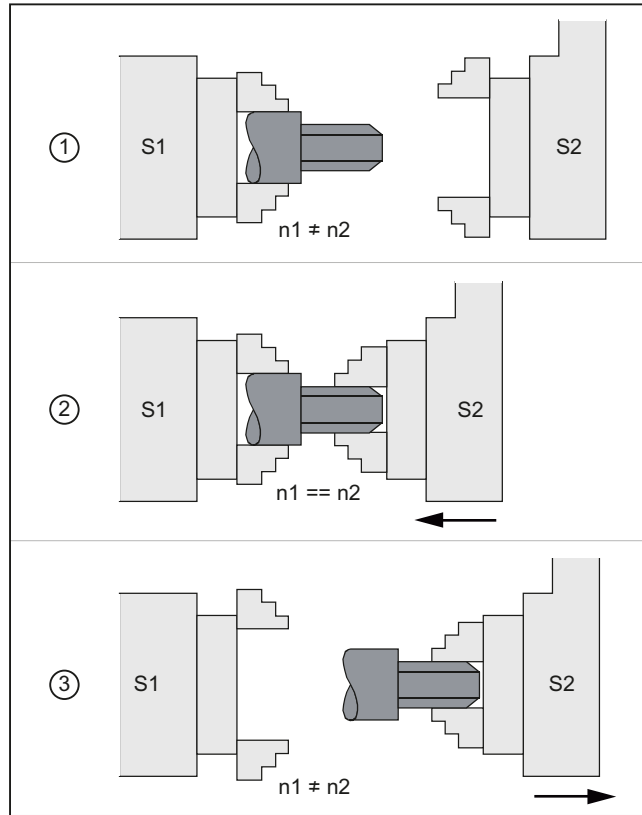
Funktionen "Synkronspindel" möjliggör en varvtalsynkron förflyttning av följd- (FS) och styrspindel (LS) med programmerbart utväxlingsförhållande.

Funktionen erbjuder följande moder:

- Vartalssynkronitet ($n_{FS} = n_{LS}$)
- Lägessynkronitet ($\phi_{FS} = \phi_{LS}$)
- Lägessynkronitet med vinkelförskjutning ($\phi_{FS} = \phi_{LS} + \Delta\phi$)

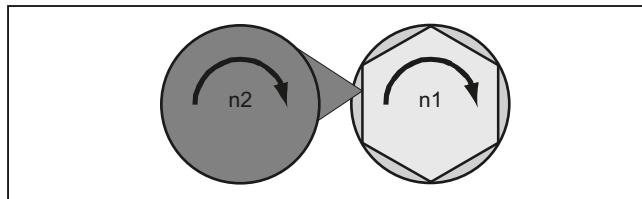
Användningsexempel:

- Flygande överlämning av arbetsstycken t.ex. till baksidebearbetning, utväxlingsförhållande: 1:1



- ① Synkronisera varvtal
- ② Överlämna arbetsstycke
- ③ Bearbeta baksida

- Flerkantsbearbetning (polygonsvarvning), varvtalssynkronitet, utväxlingsförhållande: $n_1:n_2$



Syntax

```
COUPDEF (<FS>, <LS>, <ZFS>, <NLS>, <Satzwechsel>, <Koppelart>)
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>, <Satzwechsel>, <LS>, <Satzwechsel>)

Märk

Förkortat skrivsätt

Vid anvisningarna COUPOF, COUPOFS, COUPRES och COUPDEL är ett förkortat skrivsätt utan angivande av styrspindeln möjligt.

Betydelse

COUPDEF:	Definiera/ändra koppling användarspecifikt
COUPON:	Koppla till koppling. Utgående från det aktuella varvtalet synkroniserar sig följdspindeln till styrspindeln
COUPONC:	Överta koppling vid tillkoppling med tidigare programmering av M3 S . . . eller M4 S . . . Ett differensvarvtal för följdspindeln övertas omedelbart.
COUPOF:	Koppla från koppling. - med omedelbart blockbyte: COUPOF (<S2>, <S1>) - Blockbyte först efter att ha kört förbi fränkopplingspositionen(erna) <POSFS> resp. <POSLS>: COUPOF (<S2>, <S1>, <POSFS>) COUPOF (<S2>, <S1>, <POSFS>, <POSLS>)
COUPOFS:	Koppla från en koppling med stopp för följdspindeln. Blockbyte så fort som möjligt med omedelbart blockbyte: COUPOFS (<S2>, <S1>) Blockbyte först efter att ha kört förbi fränkopplingspositionen: COUPOFS (<S2>, <S1>, <POSFS>)
COUPRES:	Återställa kopplingsparametrar till projekterade MD och SD
COUPDEL:	Radera användardefinierad koppling
WAITC:	Vänta synkroniseringsvillkor (NOC lyfts till IPO vid blockbyte)
<FS>:	Beteckning för följdspindel
Tillvalsparametrar:	
<LS>:	Beteckning för styrspindel Uppgift med spindelnummer: t.ex. S2, S1
<ZFS>, <NLS>:	Utväxlingsförhållande mellan FS och LS. <ZFS> / <NLS> = täljare / nämnare Förinställning: <ZFS> / <NLS> = 1.0 ; Uppgift av nämnaren tillval

<Satzwechsel>:	Blockbytesbeteende	
	Blockbytet görs:	
	"NOC"	genast
	"FINE"	med uppnåendet av "Synkronkörning fin"
	"COARSE"	med uppnåendet av "Synkronkörning grov"
	"IPOSTOP"	med uppnåendet av IPOSTOP, dvs. efter börvärdessidig synkronkörning (förinställning)
	Blockbytesbeteendet är modalt verksamt.	
<Koppelart>:	Kopplingstyp: Koppling mellan FS och LS	
	"DV"	Börvärdeskoppling (förinställning)
	"AV"	Ärvärdeskoppling
	"VV"	Hastighetskoppling
	Kopplingstypen är modalt verksamt.	
<POSFS>:	Vinkelförskjutning mellan styr- och följdspindel	
	Värdeområde:	0°... 359,999°
<POSFS>, <POSLS>:	Frånkopplingspositioner för följd- och styrspindel	
	"Blockbytet frigges efter att ha kört förbi POS _{FS} , POS _{LS} "	
	Värdeområde:	0°... 359,999°

Exempel

Arbeta med styr- och följdspindel

Programkod	Kommentar
N05 M3 S3000 M2=4 S2=500	Styrspindel = masterspindel = spindel 1 Följdspindel = spindel 2 Styrspindel roterar med 3000 varv/min, Följdspindel roterar med 500 varv/min.
N10 COUPDEF(S2,S1,1,1,"NOC","Dv")	Definition av kopplingen (kan också projekteras).
...	
N70 SPCON	Ta styrspindel i lägesreglering (börvärdesk.).
N75 SPCON(2)	Ta följdspindel i lägesreglering.
N80 COUPON(S2,S1,45)	Tillkoppling av flygande till offsetposition = 45 grader.
...	
N200 FA[S2]=100	Positioneringshastighet = 100 grd/min
N205 SPOS[2]=IC(-90)	Köra 90 grader överlagrat i negativ riktning.
N210 WAITC(S2,"Fine")	Vänta på synkronkörning "fin".
N212 G1 X... Y... F...	Bearbetning
...	
N215 SPOS[2]=IC(180)	Köra 180 grader överlagrat i positiv riktning.
N220 G4 S50	Fördröjningstid = 50 varv för masterspindeln
N225 FA[S2]=0	Projekterad hast. (MD) aktiveras.
N230 SPOS[2]=IC(-7200)	20 varv. Kör med projekterad hastighet i negativ riktning.

Programkod	Kommentar
...	
N350 COUPOF(S2,S1)	Frånkoppling flygande, S=S2=3000
N355 SPOSA[2]=0	Stoppa FS vid noll grader.
N360 G0 X0 Y0	
N365 WAITS(2)	Vänta på spindel 2.
N370 M5	Stoppa FS.
N375 M30	

Programmering av ett differensvarvtal

Programkod	Kommentar
	Styrspindel = masterspindel = spindel 1
	Följdspindel = spindel 2
N01 M3 S500	Styrspindel roterar med 500 varv/min.
N02 M2=3 S2=300	Följdspindel roterar med 300 varv/min.
...	
N10 G4 F1	Fördröjningstid för masterspindel.
N15 COUPDEF (S2,S1,-1)	Kopplingsfaktor med utväxlingsförhållande -1:1
N20 COUPON (S2,S1)	Aktivera koppling. Varvtalet för följdspindelns resulterar ur varvtalet för styrspindelns och kopplingsfaktorn.
...	
N26 M2=3 S2=100	Programmering av ett differensvarvtal.

Exempel vid övertagande av en rörelse till differensvarvtal

1. Koppla till koppling vid tidigare programmering av följdspindelns med COUPON

Programkod	Kommentar
	Styrspindel = masterspindel = spindel 1
	Följdspindel = spindel 2
N05 M3 S100 M2=3 S2=200	Styrspindel roterar med 100 varv/min, följdspindel med 200 varv/min.
N10 G4 F5	Fördröjningstid = 5 sekunder för masterspindelns
N15 COUPDEF(S2,S1,1)	Utväxlingsförhållande FS till LS är 1,0 (förin- ställning).
N20 COUPON(S2,S1)	Tillkoppling av flygande till styrspindel.
N10 G4 F5	Följdspindel roterar med 100 varv/min.

2. Koppla till koppling vid tidigare programmering av följdspindelns med COUPONC

Programkod	Kommentar
	Styrspindel = masterspindel = spindel 1
	Följdspindel = spindel 2
N05 M3 S100 M2=3 S2=200	Styrspindel roterar med 100 varv/min, följdspindel med 200 varv/min.
N10 G4 F5	Fördröjningstid = 5 sekunder för masterspindelns

Programkod	Kommentar
N15 COUPDEF(S2,S1,1)	Utväxlingsförhållande FS till LS är 1,0 (förinställning).
N20 COUPONC(S2,S1)	Tillkoppling flygande till styrspindel och övertagande av tidigare varvtal till S2.
N10 G4 F5	S2 roterar med 100varv/min + 200varv/min = 300varv/min

3. Tillkoppling av koppling vid stående följdspindel med COUPON

Programkod	Kommentar
	Styrspindel = masterspindel = spindel 1 Följdspindel = spindel 2
N05 SPOS=10 SPOS[2]=20	Följdspindel S2 i positioneringsdrift.
N15 COUPDEF(S2,S1,1)	Utväxlingsförhållande FS till LS är 1,0 (förinställning).
N20 COUPON(S2,S1)	Tillkoppling av flygande till styrspindel.
N10 G4 F1	Kopplingen stängs, S2 blir stående på 20 grader.

4. Tillkoppling av koppling vid stående följdspindel med COUPONC

Märk

Positionerings- eller axeldrift

Befinner sig följdspindeln före tillkopplingen i positionerings- eller axeldrift, då förhåller sig följdspindeln vid COUPON (<FS>, <LS>) och COUPONC (<FS>, <LS>) lika.

Märk

Styrspindel och axeldrift

Befinner sig styrspindeln före definitionen av kopplingen i axeldrift, verkar också efter tillkopplingen av kopplingen hastighetsgränsvärdet från maskindatum:

MD32000 \$MA_MAX_AX_VELO (maximal axelhastighet)

För att undvika detta beteende måste axeln före definitionen av kopplingen kopplas till spindelriften (M3 S... eller M4 S...).

Ytterligare informationer

Projekterad koppling

Vid den projekterade kopplingen fastläggs LS och FS via maskindatum. De projekterade spindlarna kan inte förändras i detaljprogrammet. Parametreringen av kopplingen kan göras med COUPDEF i detaljprogrammet (förutsättning: inget skrivskydd fastlagt).

Användardefinierad koppling

Med COUPDEF kan en koppling definieras på nytt eller förändras i detaljprogrammet. Är redan en koppling aktiv måste denna före definitionen av en ny koppling först raderas med COUPDEL.

En koppling definieras fullständigt genom:

COUPDEF (<FS>, <LS>, <ÜFS>, <ÜLS>, Satzwechselperhalten, Koppelart)

Följdspindel (FS) och styrspindel (LS)

Med axelnamnen för FS (följd-) och LS (styrspindel) bestäms kopplingen entydigt. Axelnamnen måste programmeras med varje anvisning COUPDEF. De andra kopplingsparametrarna är modalt verksamma och måste programmeras endast när de blir ändrade.

Exempel:

COUPDEF (S2, S1)

Utväxlingsförhållande

Utväxlingsförhållandet anges som varvtalsförhållande mellan FS och LS:

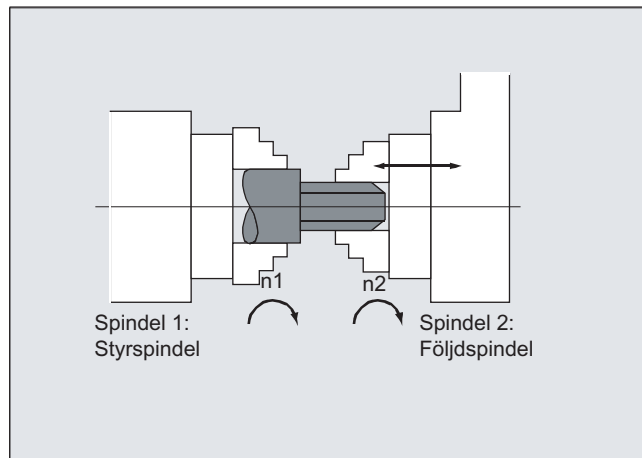
Följdspindel / styrspindel = täljare / nämnare

Täljaren måste programmeras. Nämnaren måste inte programmeras. För nämnaren sätts sedan defaultvärdet 1.0.

Exempel:

Följdspindel S2 och styrspindel S1, utväxlingsförhållande = 1 / 1

COUPDEF (S2, S1, 1.0)



Märk

Utväxlingsförhållandet kan också förändras vid tillkopplad koppling och roterande spindlar.

Blockbytesbeteende NOC, FINE, COARSE, IPOSTOP

Vid programmeringen av blockbytesbeteendet är följande förkortade skrivsätt möjligt:

- "NO": genast (förinställning)
- "FI": med uppnåendet av "Synkronkörning fin"
- "CO": med uppnåendet av "Synkronkörning grov"
- "IP": med uppnåendet av IPOSTOP, dvs. efter börvärdessidig synkronkörning

Kopplingstyp

Märk

Kopplingstypen får förändras endast vid fränkopplad koppling.

Koppla till synkron drift COUPON, <POSFS>

- Tillkoppling av kopplingen med valfri vinkelreferens mellan LS och FS:
 - COUPON (S2, S1)
 - COUPON (S2)
 - Tillkoppling av kopplingen med vinkelförskjutning <POSFS>
<POSFS> hänför sig till 0°-positionen för styrspindeln i positiv rotationsriktning
Värdeområde <POSFS>: 0°... 359,999°
 - COUPON (S2, S1, 30)
-

Märk

Vinkelförskjutningen kan ändras även vid aktiv koppling.

Positionering av följdspindeln

Även vid tillkopplad synkronspindelkoppling låter sig FS positioneras oberoende av LS i området $\pm 180^\circ$.

- Spindelpositionering av FS med SPOS
Exempel: SPOS[2]=IC(-90)
Ytterligare informationer till SPOS finns i:
Litteratur:
Programmeringshandbok Grunder

Differensvarvtal

Ett differensvarvtal uppstår i varvtalsstyrdriften och aktiv synkronspindelkoppling genom förteckenbehäftad överlagring av ett FS-varvtal på grund av LS-rörelse och ett FS-varvtal på grund av spindelprogrammering:

- Synkronspindelkoppling med COUPONC
- $S<FS>=<Drehzahl> [M<FS>=<Drehrichtung>]$

Märk**Randvillkor**

- Med rotationsriktningen M3 / M4 måste också varvtalet S . . . programmeras på nytt.
- Överlagringen av ett spindelvarvtal (M<Drehrichtung> S<FS>) genom LS-rörelsen vid synkronspindelkoppling COUPONC blir endast verksam när överlagringen är frigiven.
- Dynamiken för styrspindeln måste inskränkas så mycket att vid en överlagring följdspindelns dynamikgränsvärden inte överskrids.

Ytterligare informationer till differensvarvtal se:

Litteratur:

Funktionshandbok Tilläggsfunktioner; Synkronspindel (S3)

Hastighet, acceleration: FA, ACC, OVRA, VELOLIMA

Axial hastighet och acceleration för en följdspindel kan programmeras med:

- $FA[SPI(S<n>)]$ resp. $FA[S<n>]$ (axial hastighet)
- $ACC[SPI(S<n>)]$ resp. $ACC[S<n>]$ (axial acceleration)
- $OVRA[SPI(S<n>)]$ resp. $OVRA[S<n>]$ (axial övermanning)
- $VELOLIMA[SPI(S<n>)]$ resp. $VELOLIMA[S<n>]$ (axial hastighetsökning resp. reducering) Geschwindigkeitsüberhöhung bzw. -reduktion)

Med $<n> = 1, 2, 3, \dots$ (spindelnummer för följdspindlarna)

Litteratur:

Programmeringshandbok Grunder

Märk

En reduktion eller överhöjning av den maximala axiala jerken är inte verksam vid spindlar.

Ytterligare informationer till den axiala dynamiken finns i:

Litteratur:

Funktionshandbok Tilläggsfunktioner; Roterande axlar (R2)

Programmerbart blockbytesbeteende WAITC

Med WAITC kan blockbytesbeteendet, t.ex. efter ändring av kopplingsparametrar eller positioneringsförlopp, föreskrivas med olika synkronkörningsvillkor (grov, fin, IPOSTOP). Är inga synkronkörningsvillkor angivna, gäller det vid definitionen COUPDEF angivna blockbytesbeteendet.

Exempel

- Vänta på uppnåendet av synkronkörningsvillkoret `FINE` vid följdspindel `S2` och `COARSE` vid följdspindel `S4`: `WAITC (S2, "FINE", S4, "COARSE")`
- Vänta på uppnåendet av synkronkörningsvillkoret enligt `COUPDEF`: `WAITC ( )`

Koppla från koppling `COUPOF`

Med `COUPOF` kan fränkopplingsbeteendet för kopplingen föreskrivas:

- Fränkoppling av kopplingen med omedelbart blockbyte:
 - `COUPOF (S2, S1)` (med angivande av styrspindeln)
 - `COUPOF (S2)` (utan angivande av styrspindeln)
- Fränkoppling av kopplingen efter förbikörning av fränkopplingspositioner. Blockbytet görs efter förbikörningen av fränkopplingspositionerna.
 - `COUPOF (S2, S1, 150)` (fränkopplingsposition FS: 150°)
 - `COUPOF (S2, S1, 150, 30)` (fränkopplingsposition FS: 150°, LS: 30°)

Koppla från koppling med stopp av följdspindeln `COUPOFS`

Med `COUPOFS` kan fränkopplingsbeteendet för kopplingen föreskrivas med stopp av följdspindeln:

- Fränkoppling av kopplingen med stopp av följdspindeln och omedelbart blockbyte:
 - `COUPOFS (S2, S1)` (med angivande av styrspindeln)
 - `COUPOFS (S2)` (utan angivande av styrspindeln)
- Fränkoppling av kopplingen efter förbikörning av fränkopplingspositioner med stopp av följdspindeln. Blockbytet görs efter förbikörningen av fränkopplingspositionerna.
 - `COUPOFS (S2, S1, 150)` (fränkopplingsposition FS: 150°)

Radera kopplingar `COUPDEL`

Med `COUPDEL` raderas kopplingen:

- `COUPDEL (S2, S1)` (med angivande av styrspindeln)
- `COUPDEL (S2)` (utan angivande av styrspindeln)

Återställa kopplingsparametrar `COUPRES`

Med `COUPRES` aktiveras i maskin- och settingdata parametrerade värden för kopplingen:

- `COUPRES (S2, S1)` (med angivande av styrspindeln)
- `COUPRES (S2)` (utan angivande av styrspindeln)

Systemvariabler

- Aktuell kopplingstillstånd för följdspindeln
Det aktuella kopplingstillståndet för en följdspindel kan läsas bitcoderat via:
`<Wert> = $AA_COUP_ACT [<FS>]`

Bit	<Wert>	Betydelse
-	0	Ingen koppling aktiv
2	4	Synkronspindelkoppling aktiv

Observera:

- Alla andra värden hänför sig till axeldrift
- Är spindeln en följdspindel med flera kopplingar returneras som värde för kopplingstillståndet alla kopplingar som summatillstånd.

- Aktuell vinkelförskjutning

Den aktuella vinkelförskjutningen för följdspindeln till styrspindeln kan läsas via:

- \$AA_COUP_OFFS [<FS>] (börvärdessidig vinkelförskjutning)
- \$VA_COUP_OFFS [<FS>] (ärvärdessidig vinkelförskjutning)

Användningsexempel

Kompensering av vinkelförskjutningsdifferens i NC-programmet efter upphävande av spårföljningsdriften:

Vinkelförskjutningsdifferens = programmerad vinkelförskjutning - systemvariabel

Litteratur

Utförliga informationer till systemvariabler finns i:

Listhandbok Systemvariabler

13.6 Generisk koppling (CP...)

Den "Generiska kopplingen" är en allmän kopplingsfunktion, i vilken alla kopplingsegenskaper för de bestående kopplingstyperna (medsläpning, styrvärdeskoppling, elektronisk växel och synkronspindel) är sammanfattade.

Funktionen möjliggör en flexibel typ av programmering:

- Användaren kan för sin applikation fritt välja nödvändiga kopplingsegenskaper (byggglådeprincip).
- Varje kopplingsegenskap kan programmeras separat.
- Kopplingsegenskaperna för en definierad koppling (t. ex. kopplingsfaktor) kan ändras.
- En senare användning av ytterligare kopplingsegenskaper är möjlig.
- Koordinatreferenssystemet för följdaxeln (baskoordinatsystem eller maskinkoordinatsystem) är programmerbart.
- Vissa kopplingsegenskaper kan också programmeras i synkronaktioner.

Litteratur: Funktionshandbok Synkronaktioner

Märk

De hittillsvarande kopplingsanropen för medsläpningen (TRAIL*), styrvärdeskoppling (LEAD*), elektronisk växel (EG*) och synkronspindel (COUP*) stöds fortfarande via anpassningscykler.

Översikt över alla nyckelord och kopplingsegenskaper

Den följande tabellen visar en översikt av alla nyckelord för den generiska kopplingen och de därmed programmerbara kopplingsegenskaperna:

Nyckelord	Kopplingsegenskap / betydelse	Syntax
CPDEF	Lägga till en kopplingsmodul	CPDEF= (<FAx>)
CPDEL	Radering av en kopplingsmodul	CPDEL= (<FAx>)
CPLA	Definition av en styraxel	CPLA [<FAx>] = (<LAx>)
CPLDEF	Definition av en styraxel och tillägg av en kopplingsmodul (även möjligt med CPDEF + CPLA)	CPLDEF [<FAx>] = (<LAx>) eller CPDEF= (<FAx>) CPLA [<FAx>] = (<LAx>)
CPLDEL	Radering av en styraxel till en kopplingsmodul (även möjligt med CPDEL + CPLA)	CPLDEL [<FAx>] = (<LAx>) eller CPDEL= (<FAx>) CPLA [<FAx>] = (<LAx>)
CPON	Tillkoppling av en kopplingsmodul	CPON= (<FAx>)
CPOF	Frånkoppling av en kopplingsmodul	CPOF= (<FAx>)
CPLON	Tillkoppling av en styraxel till en kopplingsmodul	CPLON [<FAx>] = <LAx>

Nyckelord	Kopplingsegenskap / betydelse	Syntax
CPLOF	Frånkoppling av en styraxel till en kopplingsmodul	CPLOF [<FAx>] = <LAx>
CPLNUM	Täljare för kopplingsfaktorn	CPLNUM [FAx, LAx] = <Wert>
CPLDEN	Nämnare för kopplingsfaktorn	CPLDEN [FAx, LAx] = <Wert>
CPLCTID	Nummer för kurvtabeln	CPLCTID [FAx, LAx] = <Wert>
CPLSETVAL	Kopplingsreferens	CPLSETVAL [FAx, LAx] = "<Kopplungsbezug>"
		"<Kopplungsbezug>":
		"CMDPOS" Börvärdeskoppling
		"CMDVEL" Hastighetskoppling
		"ACTPOS" Ärvärdeskoppling
CPFRS	Koordinatreferenssystem	CPFRS [FAx] = "<Koordinatenbezug>"
		"<Koordinatenbezug>":
		"BCS" Baskoordinatsystem
		"MCS" Maskinkoordinatsystem
CPBC	Blockbyteskriterium	CPBC [FAx] = "<Satzwechselkriterium>"
		"<Satzwechselkriterium>":
		"NOC" Blockbyte sker oberoende av kopplings-tillstånd.
		"IPOSTOP" Blockbyte görs vid synkronkörning på börvärdessidan.
		"COARSE" Blockbyte görs vid synkronkörning "grov" på ärvärdessidan.
		"FINE" Blockbyte görs vid synkronkörning "fin" på ärvärdessidan.
CPFPOS + CPON	Synkronposition för följdaxeln vid tillkopplingen	CPON=FAx CPFPOS [FAx] = <Wert>
CPLPOS + CPON	Synkronposition för styraxeln vid tillkopplingen	CPLPOS [FAx, LAx] = <Wert>

Nyckelord	Kopplingsegenskap / betydelse	Syntax
CPFMSON	Synkroniseringsmodus	CPFMSON[FAx]="<Synchronisationsmodus>"
		"<Synchronisation smodus>":
		"CFAST" Kopplingen stängs tidsoptimerat.
		"CCOARSE" Kopplingen kopplas först till när den enligt kopplingslagen krävda följdaxelpositionen befinner sig i området för den aktuella följdaxelpositionen.
		"NTGT" Nästa tandlucka uppsöks tidsoptimerat.
		"NTGP" Nästa tandlucka uppsöks vägoptimerat.
		"NRGT" Nästa segment enligt förhållandet mellan gång-tal och tand-tal uppsöks tidsoptimerat.
		"NRGP" Nästa segment enligt förhållandet mellan gång-tal och tand-tal uppsöks vägoptimerat.
		"ACN" Endast vid roterande axlar! Den roterande axeln förflyttar sig i negativ axelriktning till synkronpositionen. Synkroniseringen sker genast.
		"ACP" Endast vid roterande axlar! Den roterande axeln förflyttar sig i positiv axelriktning till synkronpositionen. Synkroniseringen sker genast.
		"DCT" Endast vid roterande axlar! Den roterande axeln förflyttar sig tidsoptimerat till den programmerade synkronpositionen. Synkroniseringen sker genast.
		"DCP" Endast vid roterande axlar! Den roterande axeln förflyttar sig vägoptimerat till den programmerade synkronpositionen. Synkroniseringen sker genast.

Nyckelord	Kopplingsegenskap / betydelse	Syntax		
CPFMON	Beteende hos följdaxeln vid tillkoppling	CPFMON[FAx]="<Einschaltverhalten>"		
		"<Einschaltverhalten>":	"STOP"	Endast vid spindlar! En aktiv rörelse av följdspindeln stoppas före tillkopplingen.
			"CONT"	Endast vid spindlar och huvudkörningsaxlar! Den aktuella rörelsen för följdaxeln/-spindeln övertas i kopplingen som start-rörelse.
			"ADD"	Endast vid spindlar! Rörelseandelarna i kopplingen verkar förutom den aktuella överlagrade rörelsen, dvs. den aktuella rörelsen för följdaxeln/-spindeln bibehålls som överlagrad rörelse.
CPFMOF	Beteende hos följdaxeln vid fullständig fränkoppling	CPFMOF[FAx]="<Ausschaltverhalten>"		
		"<Ausschaltverhalten>":	"STOP"	Stopp för följdaxel/-spindel. En aktiv överlagrad rörelse bromsas till stillestånd. Därefter öppnas kopplingen
			"CONT"	Endast vid spindlar och huvudkörningsaxlar! Följdspindeln fortsätter att gå med det(den) vid tidpunkten för fränkopplingen aktuella varvtalet/hastigheten.
CPFPOS + CPOF	Fränkopplingsposition för följdaxeln vid fränkoppling	CPOF=(FAx) CPFPOS[FAx]=<Wert>		

Nyckelord	Kopplingsegenskap / betydelse	Syntax		
CPMRESET	Kopplingsbeteende vid RESET	CPMRESET [FAx] = "<Reset-Verhalten>"		
		"<Reset-Verhalten>":	"NONE"	Det aktuella tillståndet för kopplingen bibehålls.
			"ON"	Är den motsvarande kopplingsmodulen anlagd så koppas kopplingen till. Alla definierade styraxel-relationer aktiveras. Detta sker också när redan alla eller delar av dessa styraxel-relationer är aktiva, dvs. även vid en fullständigt aktiverad koppling sker en ny-synkronisering.
			"OF"	En aktiv överlagrad rörelse bromsas till stillestånd. Därefter kopplas kopplingen från. Anlades den motsvarande kopplingsmodulen utan explicit definition (CPDEF) så raderas kopplingsmodulen. I andra fall förblir den fortfarande anlagd dvs. den kan fortfarande användas.
			"OFC"	Endast möjligt vid spindlar! Följdspindeln fortsätter att gå med det(den) vid tidpunkten för fränkopplingen aktuella varvtalet/hastigheten. Kopplingen kopplas från. Anlades den motsvarande kopplingsmodulen utan explicit definition (CPDEF) så raderas kopplingsmodulen. I andra fall förblir den fortfarande anlagd dvs. den kan fortfarande användas.
			"DEL"	En aktiv överlagrad rörelse bromsas till stillestånd. Därefter blir kopplingen inaktiverad och sedan raderad.
			"DELC"	Endast möjligt vid spindlar! Följdspindeln fortsätter att gå med det(den) vid tidpunkten för fränkopplingen aktuella varvtalet/hastigheten. Kopplingen blir inaktiverad och sedan raderad.

Nyckelord	Kopplingsegenskap / betydelse	Syntax
CPMSTART	Kopplingsbeteende vid detalj-programstart	CPMSTART [FAx] = "<Start-Verhalten>"
		"<Start-Verhalten>":
		"NONE"
		"ON"
		Det aktuella tillståndet för kopplingen bibehålls.
		Koppling tillkopplad. Alla definierade styraxel-relationer aktiveras. Detta sker också när redan alla eller delar av dessa styraxel-relationer är aktiva, dvs. även vid en fullständigt aktiverad koppling sker en ny-synkronisering.
		"OF"
		Kopplingen kopplas från. Anlades den motsvarande kopplingsmodulen utan explicit definition (CPDEF) så raderas kopplingsmodulen. I andra fall förblir den fortfarande anlagd dvs. den kan fortfarande användas.
		"DEL"
		Kopplingen blir inaktiverad och sedan raderad.
CPMPRT	Kopplingsbeteende vid detalj-programstart under blocksökning via programtest	CPMPRT [FAx] = "<Start-Verhalten>"
		"<Start-Verhalten>":
		se CPMSTART
CPLINTR	Förflyttningsvärde för en styraxels ingångsvärde	CPLINTR [FAx, LAx] = <Wert>
CPLINSC	Skalfaktor för en styraxels ingångsvärde	CPLINSC [FAx, LAx] = <Wert>
CPLOUTTR	Förflyttningsvärde för en kopplings utgångsvärde	CPLOUTTR [FAx, LAx] = <Wert>
CPLOUTSC	Skalfaktor för en kopplings utgångsvärde	CPLOUTSC [FAx, LAx] = <Wert>
CPSYNCOV	Tröskelvärde för positionssynkronisering "Grov"	CPSYNCOV [FAx] = <Wert>
CPSYNFIP	Tröskelvärde för positionssynkronisering "Fin"	CPSYNFIP [FAx] = <Wert>
CPSYNCOV2	Andra tröskelvärde för positionssynkronisering "Grov"	CPSYNCOV2 [FAx] = <Wert>
CPSYNFIP2	Andra tröskelvärde för positionssynkronisering "Fin"	CPSYNFIP2 [FAx] = <Wert>
CPSYNCOV	Tröskelvärde för hastighets-synkronisering "Grov"	CPSYNCOV [FAx] = <Wert>
CPSYNFIV	Tröskelvärde för hastighets-synkronisering "Fin"	CPSYNFIV [FAx] = <Wert>

Nyckelord	Kopplingsegenskap / betydelse	Syntax
CPMBRAKE	Beteende hos följdaxeln vid vissa stoppsignaler och -kommandon	CPMBRAKE[FAx]=<Bitcodierter Wert>
CPMVDI	Beteende hos följdaxeln vid vissa NC/PLC-gränssnittssignaler	CPMVDI[FAx]=<Bitcodierter Wert>
CPMALARM	Undertryckning av speciella kopplingsrelaterade larmmeddelanden	CPMALARM[FAx]=<Bitcodierter Wert>
CPSETTYPE	Kopplingstyp	CPSETTYPE[FAx]="<Kopplungstyp>"
		"<Kopplungstyp>":
		"CP" Fri programmerbarhet
		"TRAIL" Kopplingstyp "medsläpning"
		"LEAD" Kopplingstyp "styrvärdeskoppling"
		"EG" Kopplingstyp "elektronisk växel"
		"COUP" Kopplingstyp "synkronspindel"

FAx: Följdaxel/-spindel

LAx: Styraxel/-spindel

Märk

Kopplingsegenskaper som inte programmeras explicit (i detaljprogram eller i synkronaktioner) blir verksamma med sina standardinställningar.

Beroende av inställningen för nyckelordet CPSETTYPE kan i stället för standardinställningarna (CPSETTYPE="CP") också förinställda kopplingsegenskaper bli verksamma.

Litteratur

För utförliga informationer till Generisk koppling se:

- Funktionshandbok Specialfunktioner; M3: Axelkopplingar, Kapitel: "Generisk koppling"

13.7 Master-/Slave-koppling (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS)

"Master/Slave-kopplingen" möjliggör:

- inkopplingen av slave-axlarna på master-axeln när de deltagande axlarna står stilla.
- kopplingen och separeringen av **roterande**, varvtalsstyrda spindlar.
- den dynamiska projekteringen.

Märk

Positioneringsdrift

För axlar och spindlar i positioneringsdrift stängs och separeras kopplingen endast vid stillestånd.

Syntax

```
MASLON(<Slave_1>,<Slave_2>,...)
MASLOF(<Slave_1>,<Slave_2>,...)
MASLOFS(<Slave_1>,<Slave_2>,...)
```

Dynamisk projektering:

```
MASLDEF(<Slave_1>,<Slave_2>,...,<Master>)
MASLDEL(<Slave_1>,<Slave_2>,...)
```

Betydelse

MASLON:	Koppla till en temporär Master/Slave-koppling	
	<Slave_x>,...:	Slave-axlar 1 ... n
MASLOF:	Separera en temporär Master/Slave-koppling	
	<Slave_1>,...:	Slave-axlar 1 ... n
MASLOFS:	Separera Master/Slave-koppling och automatiskt bromsa Slave-spindlar (se anvisning "Kopplingsbeteende för spindlar i varvtalsstyrdrift"!)	
	<Slave_1>,...:	Slave-axlar 1 ... n
MASLDEF:	Lägga till/ändra Master/Slave-förband utifrån detaljprogrammet	
	<Slave_1>,...:	Slave-axlar 1 ... n
	<Master>:	Master-axel
MASLDEL:	Separera master/slave-koppling och radera definitionen av förbandet	
	<Slave_1>,...:	Slave-axlar 1 ... n
	Observera: De i maskindata projekterade master-/slave-definitionerna bibehålls.	

Märk**Kopplingsbeteende för spindlar i varvtalsstyrdrift**

För spindlar i varvtalsstyrdrift fastläggs kopplingsbeteendet av MASLON, MASLOF, MASLOFS och MASLDEL explicit via följande maskindatum:

MD37263 \$MA_MS_SPIND_COUPLING_MODE

I standardinställningen med MD37263 = 0 görs inkopplingen och separeringen av slave-axlarna uteslutande när de deltagande axlarna står stilla. MASLOFS motsvarar MASLOF.

Vid MD37263 = 1 blir kopplingsanvisningen omedelbar och därmed också utförd i rörelsen. Kopplingen stängs genast vid MASLON och vid MASLOFS eller MASLOF genast separerad. De vid denna tidpunkt roterande slave-spindlarna behåller vid MASLOF sina varvtal till förnyad varvtalsprogrammering. Vid MASLOFS bromsas de däremot automatiskt.

Märk

Vid MASLOF/MASLOFS utgår det implicita fördekoderingsstoppet. Orsakat genom det saknade fördekoderingsstoppet levererar \$P-systemvariablerna inga aktualiserade värden för slave-axlarna till tidpunkten för förnyad programmering.

Märk

För slave-axlarna kan ärvärdet genom PRESETON synkroniseras till samma värde som master-axeln. För detta måste den varaktiga master-/slave-kopplingen kortfristig kopplas från, för att sätta ärvärdet för den ej refererade slave-axeln på värdet för master-axeln med POWER ON. Därefter upprättas åter den varaktiga kopplingen.

Den varaktiga master-/slave-kopplingen aktiveras med följande MD-inställning:

MD37262 \$MA_MS_COUPLING_ALWAYS_ACTIVE = 1

Den ingen inverkan på språkkommandona för den temporära kopplingen.

Exempel**Exempel 1: Ärvärdesinställning vid en slave-axel i en master/slave-koppling**

Vid en permanent master/slave-koppling sätts med PRESETON ärvärdet för slave-axeln på värdet för master-axeln.

Programkod	Kommentar
\$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=0	; Koppla från permanent koppling för slave-axeln
NEWCONF	; Aktivera maskindataändring
STOPRE	
MASLOF(Y1)	; Koppla från temporär koppling
PRESETON(AX2,\$VA_IM(M_AX))	; Ärvärde för slave-axeln = ärvärde för master-axeln
\$MA_MS_COUPLING_ALWAYS_ACTIVE[AX2]=1	; Koppla till permanent koppling för slave-axeln

13.7 Master-/Slave-koppling (MASLDEF, MASLDEL, MASLON, MASLOF, MASLOFS)

Programkod	Kommentar
NEWCONF	; Aktivera maskindataändring

Exempel 2: Dynamisk projektering av en master/slave-koppling

För att kopplingen efter axelcontainer-vridningen ska kunna stängas med en annan spindel måste dessförinnan den gamla kopplingen separeras, projekteringen raderas och den nya kopplingen projekteras.

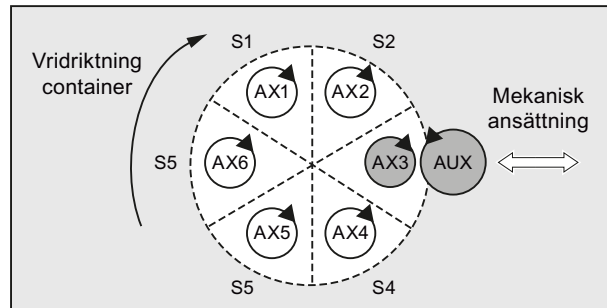


Bild 13-1 Före axelcontainer-vridningen.

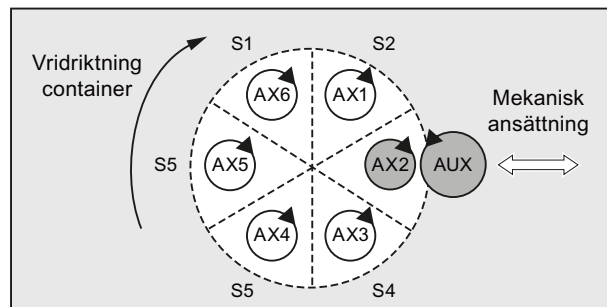


Bild 13-2 Efter axelcontainer-vridningen runt en slot:

Programkod	Kommentar
MASLDEF (AUX, S3)	; AUX: Slave, S3: Master = AX3
MASLON (AUX)	; Koppling till
M3=3 S3=4000	; Vrida master
MASLDEL (AUX)	; Skilja koppling och radera
AXCTSWE (CT1)	; Frige axelcontainer-vridning
MASLDEF (AUX, S3)	; AUX: Slave, S3: Master = AX2

Litteratur

- Funktionshandbok Specialfunktioner; Kapitel "TE3: Varvtal-/vridmomentkoppling, master-slave"
- Funktionshandbok Tilläggfunktioner; Kapitel "B3: Decentrala system - endast 840D sl" > "NCU-link" > "Axelcontainer"

Synkronaktioner

14.1 Definition av en synkronaktion

En synkronaktion definieras i ett block i ett detaljprogram. Inom detta block får inga ytterligare kommandon programmeras, som inte är beståndsdel i synkronaktionen.

En synkronaktion består av följande komponenter:

Giltighet, ident-nr (tillval)	Villkorsdel (tillval)			Aktionsdel		
	Frekvens	G-kommando (tillval)	Villkor	Nyckelord	G-kommando (tillval)	Aktioner
--- ¹⁾ ID=<Nr> IDS=<Nr>	--- ¹⁾ WHENEVER FROM WHEN EVERY	G...	Logiskt uttryck	DO	G...	Aktion 1 ... Aktion n

¹⁾ inte programmerad

Syntax

```
DO <Aktion 1> ... <Aktion n>
<Häufigkeit> [<G-Funktion>] <Bedingung> DO <Aktion 1> ... <Aktion n>
ID=<Nr> <Häufigkeit> [<G-Funktion>] <Bedingung> DO <Aktion 1> ...
<Aktion n>
IDS=<Nr> <Häufigkeit> [<G-Funktion>] <Bedingung> DO <Aktion 1> ...
<Aktion n>
```

Litteratur

En utförlig beskrivning av funktionalitet hos synkronaktioner finns i:

Funktionshandbok Synkronaktioner

Pendling

15.1 Asynkron pendling (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB)

En pendelaxel går mellan de två vändpunkterna 1 och 2 med föreskriven matning fram och tillbaka tills pendelrörelsen kopplas från.

Andra axlar kan valfritt interpoleras under pendelrörelsen. Via en banrörelse eller med en positioneringsaxel kan en kontinuerlig ansättning uppnås. Därvid består dock **inget samband** mellan pendel- och ansättningsrörelsen.

Egenskaper hos den asynkrona pendlingen

- Den asynkrona pendlingen är axelspecifikt verksam utöver blockgränser.
- Via detaljprogrammet är en blocksynkron tillkoppling av pendelrörelsen garanterad.
- En gemensam interpolering av flera axlar och en överlagring av pendelsträckor är inte möjligt.

Programmering

Via de följande kommandona är en för genomarbetningen av NC-programmet lämplig tillkoppling och påverkan av den asynkrona pendlingen möjlig från detaljprogrammet.

De programmerade värdena förs in blocksynkront i de motsvarande settingdata under huvudkörningen och förblir verksamma till nästa ändring.

Syntax

```
OSP1 [<Achse>]=<Wert>  OSP2 [<Achse>]=<Wert>
OST1 [<Achse>]=<Wert>  OST2 [<Achse>]=<Wert>
FA [<Achse>]=<Wert>
OSCTRL [<Achse>]=(<Setzoption>,<Rücksetzoption>)
OSNSC [<Achse>]=<Wert>
OSE [<Achse>]=<Wert>
OSB [<Achse>]=<Wert>
OS [<Achse>]=1
OS [<Achse>]=0
```

Betydelse

<Achse>:	Namn på pendelaxeln		
OS:	Koppla till/från pendlingen		
	Värde:	1	Koppla till pendlingen
		0	Koppla från pendlingen
OSP1:	Fastlägga position för vändpunkt 1		

15.1 Asynkron pendling (OS, OSP1, OSP2, OST1, OST2, OSCTRL, ONSNC, OSE, OSB)

OSP2:	Fastlägga position för vändpunkt 2 Observera: Om ett inkrementellt förfarande är aktivt, så beräknas positionen inkrementellt till den sista i NC-programmet programmerade motsvarande vändpositionen.		
OST1:	Fastlägga hållningstid i vändpunkt 1 i [s]		
OST2:	Fastlägga hållningstid i vändpunkt 2 i [s]		
	<Wert>:	-2	Interpolering fortsätts utan väntan på precisionsstopp
		-1	Vänta på grovt precisionsstopp
		0	Vänta på fint precisionsstopp
		>0	Vänta på fint precisionsstopp och sedan vänta tills den angivna hållningstiden är över Observera: Enheten för hållningstiden är identisk med den via G4 programmerade hållningstiden.
FA:	Fastlägga matningshastighet Som matningshastighet gäller den definierade matningshastigheten för positioneringsaxeln. Är igen matningshastighet definierad gäller det i maskindatum lagrade värdet.		
OSCTRL:	Ange inställnings- och återställningsoptioner Optionsvärdena 0 - 3 kodifierar beteendet i vändpunkterna vid fränkopplingen. En av varianterna 0 - 3 kan väljas. De övriga inställningarna är efter behov kombinerbara med den valda varianten. Flera optioner fogas till varandra med plustecken (+).		
	<Wert>:	0	Vid fränkoppling av pendelrörelsen stoppa i nästa vändpunkt (förinställning) Observera Endast möjligt genom återställning av värdena 1 och 2.
		1	Vid fränkoppling av pendelrörelsen stoppa i vändpunkt 1
		2	Vid fränkoppling av pendelrörelsen stoppa i vändpunkt 2
		3	Vid fränkoppling av pendelrörelsen ej uppsöka någon vändpunkt om inga renbränningsslag är programmerade
		4	Efter renbränningen uppsöka slutposition
		8	Avbryts pendelrörelsen av restvägsradering, ska sedan renbränningsslag genomarbetas och ev. slutpositionen uppsökas.
		16	Avbryts pendelrörelsen av restvägsradering ska som vid fränkoppling den motsvarande vändpositionen uppsökas.
		32	Ändrad matning är aktiv först från och med nästa vändpunkt
		64	FA lika med 0, FA = 0: Vägöverlagring är aktiv FA skild från 0, FA <> 0: Hastighetsöverlagring är aktiv
		128	Vid roterande axel DC (kortaste väg)
		256	Renbränningsslag utförs som dubbelslag. (Standard) 1=renbränningsslag utförs som enkelslag.
		512	Först uppsöka startposition
ONSNC:	Fastlägga antal renbränningsslag		

15.1 Asynkron pendling (OS, OSP1, OSP2, OST1, OST2, OSCTRL, OSNSC, OSE, OSB)

OSE:	Fastlägga slutposition (i WKS) som ska uppsökas efter fränkoppling av pendlingen Observera: Vid programmering av "OSE" blir för "OSCTRL" implicit option 4 verksam.
OSB:	Fastlägga startposition (i WKS) som ska uppsökas efter tillkoppling av pendlingen Startpositionen uppsöks före vändpunkt 1. Överensstämmer startpositionen med vändposition 1 så uppsöks härnäst vändposition 2. När startpositionen uppnås verkar ingen hållningstid, inte heller när startpositionen överensstämmer med vändposition 1, i stället väntas på fint precisionsstopp. Ett inställt villkor för precisionsstopp respekteras. Observera: För att startpositionen ska uppsökas måste Bit 9 vara inställt i settingdatum SD43770 \$SA_OSCILL_CTRL_MASK.

Exempel

Exempel 1: Pendelaxel ska pendla mellan två vändpunkter

Pendelaxeln ska pendla mellan position 10 och 100. Vändpunkt 1 ska uppsökas med fint precisionsstopp, vändpunkt 2 med grovt precisionsstopp. Matningen för pendelaxeln ska uppgå till 250. I slutet av bearbetningen ska 3 renbränningsslag göras och pendelaxeln ska styra till slutposition 200. Matningen för ansättningsaxeln ska uppgå till 1, slutet på ansättningen i X-riktning ska vara uppnådd vid position 15.

Programkod	Kommentar
WAITP(X,Y,Z)	; Utgångsläge
G0 X100 Y100 Z100	; Omkoppling till positioneraxeldrift.
WAITP(X,Z)	
OSP1[Z]=10 OSP2[Z]=100	; Vändpunkt 1, vändpunkt 2.
OSE[Z]=200	; Ändposition
OST1[Z]=0 OST2[Z]=-1	; Hållningstid vid U1: Fint precisionsstopp ; Hållningstid vid U2: Grovt precisionsstopp
FA[Z]=250 FA[X]=1	; Matning pendelaxel, matning ansättningsaxel.
OSCTRL[Z]=(4,0)	; Sättningsoptioner.
OSNSC[Z]=3	; 3 renbränningsslag.
OS[Z]=1	; Starta pendling.
WHEN \$A_IN[3]==TRUE DO DELDTG(X)	; Restvägsradering.
POS[X]=15	; Utgångsläge för X-axeln
POS[X]=50	; Slutläge för X-axeln.
OS[Z]=0	; Stoppa pendling.
M30	

Märk

Kommandoföljden "OSP1[Z]=..." till "OSNCS[Z]=..." kan också programmeras i ett block.

Exempel 2: Pendling med online-ändring av vändpositionen

De för den asynkrona pendlingen erforderliga settingdata kan ställas in i detaljprogrammet.

Skrivs settingdata in direkt i detaljprogrammet så blir ändringen verksam redan vid förkörningstidpunkten. Synkront beteende kan uppnås via ett fördekoderingsstopp (STOPRE).

Programkod	Kommentar
\$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	; När ärvärdet för pendelaxeln har överskridit vändpunkten,
ID=2 WHENEVER \$AA_IM[Z] < \$\$AA_OSCILL_REVERSE_POS2[Z] DO \$AA_OVR[X]=0	stoppas ansättningsaxeln.
OS[Z]=1 FA[X]=1000 POS[X]=40	; Koppla till pendling.
OS[Z]=0	; Koppla från pendling.
M30	

Ytterligare informationer**Pendelaxel**

Fö pendelaxeln gäller:

- Varje axel kan användas som pendelaxel.
- Flera pendelaxlar kan vara aktiva samtidigt (maximalt: Antal positioneringsaxlar).
- För pendelaxeln är alltid - oberoende av i programmet aktuellt giltigt G-kommando - linjärinterpolering G1 aktiv.

Pendelaxeln kan:

- vara engångsaxel för den dynamiska transformationen
- vara styraxel vid Gantry- och medsläpningsaxlar
- förflyttas:
 - utan jerkbegränsning "BRISK" eller
 - med jerkbegränsning "SOFT" eller
 - med knäckt accelerationskurva (som positioneringsaxlar)

Pendelvändpunkter

Vid fastläggande av pendelpositionerna ska det tas hänsyn till de aktuella förflyttningarna :

- Absolut uppgift
"OSP1[Z]=<Wert>"
Position vändpunkt = summan av förflyttningarna + programmerat värde
- Relativ uppgift
"OSP1[Z]=IC(<Wert>)"
Position vändpunkt = vändpunkt 1 + programmerat värde

Exempel:

Programkod
N10 OSP1[Z]=100 OSP2[Z]=110
...
N40 OSP1[Z]=IC(3)

WAITP

Ska det pendlas med en geometriaxel, så måste denna frigivas för pendling med "WAITP".

Efter avslutad pendling förs pendelaxeln åter in som positioneringsaxel med "WAITP" och kan åter användas normalt.

Pendling med rörelsesynkronaktioner och hållningstider

Efter det de inställda hållningstiderna gått till ända, äger vid pendlingen det interna blockbytet rum (synligt på de nya restvägarna för axlarna). Vid blockbyte kontrolleras fränkopplingsfunktionen. Därvid blir efter den inställda styrinställningen för rörelseförloppet (OSCTRL) fränkopplingsfunktionen fastlagd. *Detta tidsbeteende kan påverkas av matningsövermanningen.*

Ibland utförs därefter ännu ett pendelslag, innan renbränningsslagen startas eller slutpositionen uppsöks. *Därvid uppstår intrycket att fränkopplingsbeteendet förändrar sig. Detta är dock inte fallet.*

15.2 Via synkronaktioner styrd pendling (OSCILL)

Vid denna typ av pendling är en ansättningsrörelse tillåten endast vid vändpunkterna resp. inom definierade vändningsområden.

Allt efter krav kan pendelrörelsen under ansättningen

- fortsättas eller
- stoppas tills ansättningen är fullständigt genomförd.

Syntax

1. Fastlägga parametrar för pendlingen
2. Definiera rörelsesynkronaktioner
3. Tillordna axlar, fastlägga ansättning

Betydelse

OSP1[<Pendelachse>]=	Position för vändpunkt 1
OSP2[<Pendelachse>]=	Position för vändpunkt 2
OST1[<Pendelachse>]=	Hållningstid i vändpunkt 1 i sekunder
OST2[<Pendelachse>]=	Hållningstid i vändpunkt 2 i sekunder
FA[<Pendelachse>]=	Matning för pendelaxeln
OSCTRL[<Pendelachse>]=	Inställnings- resp. återställningsposition
OSNSC[<Pendelachse>]=	Antal renbränningsslag
OSE[<Pendelachse>]=	Slutposition
WAITP(<Pendelachse>)	Frige axel för pendling

Axeltillordning, ansättning

OSCILL[<Pendelachse>]=(<Zustellachse 1>,<Zustellachse 2>,<Zustellachse 3>)

POSP[<Zustellachse>]=(<Endpos>,<Teillänge>,<Modus>)

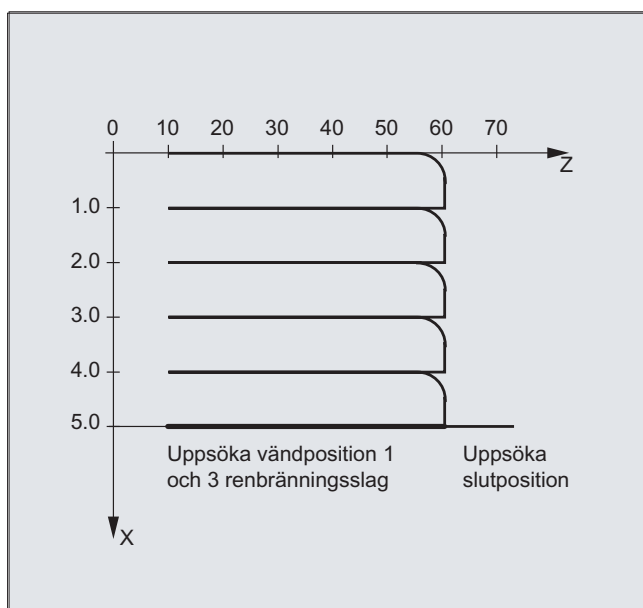
OSCILL:	Tillordna ansättningsax(el)(lar) för pendelaxeln
POSP:	Fastlägga total och delansättningar (se Kapitel Fil- och programförvaltning)
Endpos:	Slutposition för ansättningsaxeln efter det att alla delansättningar har körts.
Teillänge:	Storlek på delansättningen i vändpunkten/vändområdet
Modus:	Uppdelning av den totala ansättningen i delansättningar = två lika stora reststeg (förinställning); = alla delansättningar lika stora

Rörelsesynkronaktioner

WHEN... .. DO	när..., då...
WHENEVER ... DO	alltid när..., då...

Exempel

I vändpunkt 1 ska ingen ansättning göras. Vid vändpunkt 2 ska ansättningen göras redan på avståndet ii2 före vändpunkt 2 och pendelaxeln inte vänta i vändpunkten på avslutningen av delansättningen. Axel Z är pendelaxel och axel X ansättningsaxel.



1. Parametrar för pendlingen

Programkod	Kommentar
DEF INT ii2	; Definiera variabel för vändområde 2
OSP1[Z]=10 OSP2[Z]=60	; Definiera vändpunkt 1 och 2
OST1[Z]=0 OST2[Z]=0	; Vändpunkt 1: Fint precisionsstopp Vändpunkt 2: Fint precisionsstopp
FA[Z]=150 FA[X]=0,5	; Matning pendelaxel Z, matning ansättningsaxel X
OSCTRL[Z]=(2+8+16,1)	; Koppla från pendelrörelse i vändpunkt 2; efter RWL renbränna och uppsöka slutposition; efter RWL uppsöka motsvarande vändposition
OSNC[Z]=3	; Renbränningslag
OSE[Z]=70	; Slutposition = 70
ii2=2	; Ställa in vändområde
WAITP(Z)	; Tillåter pendling för Z-axeln

2. Rörelsesynkronaktion

Programkod	Kommentar
WHENEVER \$AA_IM[Z]<\$SA_OSCILL_REVERSE_POS2[Z] DO -> \$AA_OVR[X]=0 \$AC_MARKER[0]=0	; Alltid när den aktuella positionen för pendelaxeln Z i MKS är mindre än början på vändområde 2, då sätts den axiella övermanningen för ansättningsaxeln X på 0% och flaggan med index 0 på värdet 0.

Programkod	Kommentar
WHENEVER \$AA_IM[Z]>=\$SA_OSCILL_REVERSE_POS2[Z] DO \$AA_OVR[Z]=0	; Alltid när den aktuella positionen för pendelaxeln Z i MKS är större än eller lika med vändposition 2, då sätts den axiala övermanningen för pendelaxeln Z på 0%.
WHENEVER \$AA_DTEPW[X]==0 DO \$AC_MARKER[0]=1	; Alltid när restvägen för delansättningen är lika med 0, då sätts flaggan med index 0 på värdet 1.
WHENEVER \$AC_MARKER[0]==1 DO \$AA_OVR[X]=0 \$AA_OVR[Z]=100	; Alltid när flaggan med index 0 är lika med 1, då sätts den axiala övermanningen för ansättningsaxeln X på 0%. Därmed förhindras en för tidig ansättning (pendelaxel Z har ännu inte åter lämnat vändområde 2, men ansättningsaxel X är beredd för en förnyad ansättning). Sätter den axiala övermanningen för pendelaxeln Z från 0% (aktion i den 2:a synkronaktionen) till förflyttning åter på 100%.

-> måste programmeras i ett block

3. Starta pendling

Programkod	Kommentar
OSCILL[Z]=(X) POSP[X]=(5,1,1)	; Starta axlarna Pendelaxeln Z får axel X som ansättningsaxel ;tillordnad. Axeln X ska till slutpositionen 5 köra i steg på 1.
M30	; Programslut

Ytterligare information

- Fastlägga pendelparametrar**
Före rörelseblocket, som innehåller tillordningen av ansättnings- och pendelaxel samt fastläggandet av ansättningen, ska parametrarna för pendlingen fastläggas (se "Asynkron pendling").
- Fastlägga rörelsesynkronaktioner**
Via synkronvillkor följer:
Undertrycka ansättning, tills pendelaxeln befinner sig inom ett vändområde (ii1, ii2) eller i en vändpunkt (U1, U2).
Stoppa pendelrörelse under ansättningen i vändpunkten.
Starta pendelrörelse åter efter avslutad delansättning.
Start av nästa delansättning fastläggs.
- Tillordna pendel- och ansättningsaxel** samt fastlägga **total och delansättning**.

Fastlägga pendelparametrar

Tillordning av pendel- och ansättningsaxel: OSCILL

OSCILL[<Pendelachse>]=(<Zustellachse1>,<Zustellachse2>,<Zustellachse3>)

Med kommandot "OSCILL" görs axeltillordningarna och starten av pendelrörelsen.

Maximalt kan en pendelaxel tillordnas 3 ansättningsaxlar.

Märk

Före starten av pendlingen måste synkronvillkoren för axlarnas beteende vara fastlagt.

Fastlägga ansättningar: POSP

POSP[<Zustellachse>]=(<Endpos>,<Teillänge>,<Modus>)

Med kommandot "POSP" meddelas styrningen:

- Total ansättning (via slutpositionen)
- Storleken på respektive delansättning vid vändpunkten resp. i vändområdet
- Delansättningsbeteendet när slutpositionen uppnås (via moden)

Mode = 0	För de båda sista delansättningarna görs en uppdelning av den kvarvarande vägen fram till målpunkten på 2 lika stora reststeg (förinställning).
Mode = 1	Alla delansättningar är lika stora. De beräknas ur den totala ansättningen.

Fastlägga rörelsesynkronaktioner

De i det följande utförda rörelsesynkronaktionerna används helt allmänt till pendling.

Du finner exempellösningar för lösningen av de enskilda kraven, som tjänar som komponent för upprättande av användarspecifika pendelrörelser.

Märk

I enskilda fall kan synkronvillkoren också programmeras annorlunda.

Nyckelord

WHEN ... DO ...	när..., då...
WHENEVER ... DO	alltid när..., då...

Funktioner

Med de i det följande detaljerat beskrivna språkverktygen kan följande funktioner realiserats:

1. Ansättning i vändpunkt.
2. Ansättning i vändområde.
3. Ansättning i båda vändpunkterna.

4. Stoppande av pendelrörelsen i vändpunkten.
5. Åter starta pendelrörelsen.
6. Inte starta delansättningen för tidigt.

För alla här som exempel framställda synkronaktioner gäller följande antaganden:

- Vändpunkt 1 < vändpunkt 2
- Z = pendelaxel
- X = ansättningsaxel

Märk

För närmare förklaringar se Kapitel Rörelsesynkronaktioner.

Tillordna pendel- och ansättningsaxel samt fastlägga total och delansättning

Ansättning i vändområde

Ansättningsrörelsen ska börja inom ett vändområde innan vändpunkten har uppnåtts.

Dessa synkronaktioner förhindrar ansättningsrörelsen tills pendelaxeln befinner sig i ett vändområde.

Under givna antaganden(se ovan) resulterar följande anvisningar:

Vändområde 1:

```
WHENEVER $AA_IM[Z]>$SA_OSCILL_RESERVE_POS1[Z]+ii1 DO $AA_OVR[X]=0
```

Alltid när den aktuella positionen för pendelaxeln i MKS är större än början på vändområde 1, då sätts den axiella övermanningen för ansättningsaxeln på 0%.

Vändområde 2:

```
WHENEVER $AA_IM[Z]<$SA_OSCILL_RESERVE_POS2[Z]+ii2 DO $AA_OVR[X]=0
```

Alltid när den aktuella positionen för pendelaxeln i KS är mindre än början på vändområde 2, då sätts den axiella övermanningen för ansättningsaxeln på 0%.

Ansättning i vändpunkt

Så länge som pendelaxeln inte har uppnått vändpunkten äger ingen rörelse rum för ansättningsaxeln.

Under givna antaganden(se ovan) resulterar följande anvisningar:

Vändområde 1:

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Alltid när den aktuella positionen för pendelaxeln Z i MKS är större eller mindre än positionen för vändpunkt 1, då sätts den axiella övermanningen för ansättningsaxel X på 0% och den axiella övermanningen för pendelaxeln Z på 100%.

Vändområde 2:**För vändpunkt 2:**

```
WHENEVER $AA_IM[Z]<>$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Alltid när den aktuella positionen för pendelaxeln Zu i MKS är större eller mindre än positionen för vändpunkt 2, då sätts den axiala övermanningen för ansättningsaxel X på 0% och den axiala övermanningen för pendelaxeln Z på 100%.

Stoppande av pendelrörelsen i vändpunkten

Pendelaxeln stoppas i vändpunkten, samtidigt börjar ansättningsrörelsen. Pendelrörelsen fortsätter när ansättningsrörelsen är fullständigt utförd.

Samtidigt kan denna synkronaktion användas till att starta ansättningsrörelsen om denna stoppades av en föregående synkronaktion som ännu är verksam.

Under givna antaganden(se ovan) resulterar följande anvisningar:

Vändområde 1:

```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS1[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Alltid när den aktuella positionen för pendelaxeln i MKS är lika med vändposition 1, då sätts den axiala övermanningen för pendelaxeln på 0% och den axiala övermanningen för ansättningsaxeln på 100%.

Vändområde 2:

```
WHENEVER $SA_IM[Z]==$SA_OSCILL_RESERVE_POS2[Z] DO $AA_OVR[X]=0
$AA_OVR[Z]=100
```

Alltid när den aktuella positionen för pendelaxeln Zu i MKS är lika med vändposition 2, då sätts den axiala övermanningen för pendelaxeln X på 0% och den axiala övermanningen för ansättningsaxeln på 100%.

Online-utvärdering av vändpunkten

Står på den högra sidan om jämförelsen en med \$\$ kännetecknad huvudkörningsvariabel, så utvärderas de båda variablerna kontinuerligt i IPO-takt och jämförs med varandra.

Märk

Mer informationer härtill se kapitel "Rörelsesynkronaktioner".

Åter starta pendelrörelsen

Denna synkronaktion används för att fortsätta pendelaxelns rörelse när delansättningsrörelsen är avslutad.

Under givna antaganden(se ovan) resulterar följande anvisningar:

```
WHENEVER $AA_DTEPW[X]==0 DO $AA_OVR[Z]=100
```

Alltid när restvägen för delansättningen för ansättningsaxel X i WKS är lika med noll, då sätts den axiala övermanningen för pendelaxeln på 100%.

Nästa delansättning

Efter utförd ansättning måste en för tidig start av nästa delansättning förhindras.

Därtill används en kanalspecifik flagga (`$AC_MARKER[Index]`), som sätts i slutet av delansättningen (delrestväg $\equiv 0$) och raderas när vändområdet lämnas. Då förhindras nästa ansättningsrörelse med en synkronaktion.

Under givna antaganden (se ovan) resulterar t.ex. för vändpunkt 1 följande anvisningar:

1. Sätta flagga:

```
WHENEVER $AA_DTEPW[X]==0 DO $AC_MARKER[1]=1
```

Alltid när restvägen för delansättningen för ansättningsaxel X i WKS är lika med noll, då sätts flaggan med index 1 på 1.

2. Radera märke

```
WHENEVER $AA_IM[Z]<> $SA_OSCILL_RESERVE_POS1[Z] DO $AC_MARKER[1]=0
```

Alltid när den aktuella positionen för pendelaxel Z i MKS är större eller mindre än positionen för vändpunkt 1, då sätts flagga 1 på 0.

3. Förhindra ansättning

```
WHENEVER $AC_MARKER[1]==1 DO $AA_OVR[X]=0
```

Alltid när flaggan är lika med 1, då sätts den axiala övermanningen för ansättningsaxeln X på 0%.

Stansning och nibbling

16.1 Aktivering/inaktivering

16.1.1 Koppla till/från stansning och nibbling (SPOF, SON, PON, SONS, PONS, PDELAYON, PDELAYOF, PUNCHACC)

Aktivera/inaktivera stansning resp. nibbling

Med `PON` och `SON` aktiveras stans- resp. nibbelfunktionen. `SPOF` avslutar alla stans- och nibbelspecifika funktioner. De modalt verksamma kommandona `PON` och `SON` utesluter varandra, dvs. `PON` inaktiverar `SON` och omvänt.

Stansning/nibbling med inledning

Funktionerna `SONS` och `PONS` kopplas också till stans- resp. nibbelfunktionerna.

I motsats till den vid `SON/PON` verksamma slagstyrningen på interpoleringsnivå sker vid dessa funktioner den signaltekniska styrningen av slagutlösningen på servonivå. Härigenom kan arbetas med högre slagfrekvenser och därmed högre stansprestanda.

Under signalutvärderingen i inledningen är alla funktioner reglade, som leder till positionsändring för nibbel- eller stansaxlarna (t.ex. manuell förflyttning med handratt, ändringar av frames via PLC, mätfunktioner).

Stansning med fördröjning

`PDELAYON` förorsakar en fördröjd utmatning av stansslaget. Det modalt verksamma kommandot har förberedande funktion och står därmed som regel före `PON`. Efter `PDELAYOF` stansas normalt vidare.

Märk

Fördröjningstiden ställs in i settingdatum `SD42400 $SC_PUNCH_DWELLTIME`.

Vägberoende acceleration

Med `PUNCHACC` kan en accelerationskurva fastläggas, som definierar olika accelerationer beroende på hålavstånd.

Andra stans-interface

Maskiner som omväxlande ska använda ett andra stans-interface (andra stansenhet eller ett jämförbart medium), kan kopplas om till ett andra par av styrningens snabba digitala in- och utgångar (I/O-par). För båda stans-interfaces är den fulla stans-/nibbel-funktionaliteten

användbar. Omkopplingen mellan den första och andra stans-interfacen görs via kommandona SPIF1 och SPIF2.

Märk

Förutsättning: Via maskindata måste ett andra I/O-par vara definierat för stansfunktionaliteten (→ se uppgifter från maskintillverkaren!).

Syntax

```
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
```

Betydelse

PON:	Aktivera stansning
SON:	Aktivera nibbling
PONS:	Aktivera stansning med inledning
SONS:	Aktivera nibbling med inledning
SPOF:	Inaktivera stansning/nibbling
PDELAYON:	Aktivera stansning med fördröjning
PDELAYOF:	Inaktivera stansning med fördröjning
PUNCHACC:	Aktivera vägberoende acceleration
	Parametrar:
	<Smin> Minsta hålavstånd
	<Amin> Begynnelseacceleration <Amin> kan vara större än <Amax>.
	<Smax> Största hålavstånd
	<Amax> Slutacceleration <Amax> kan vara större än <Amin>.
SPIF1:	Aktivera första stans-interface Slagstyrningen sker via det första paret av snabba I/O.
SPIF2:	Aktivera andra stans-interface Slagstyrningen sker via det andra paret av snabba I/O.
	Observera: Efter RESET eller styrningsstart är alltid den första stans-interfacen aktiv. Används endast en stans-interface så måste denna inte programmeras.

Exempel

Exempel 1: Aktivera nibbling

Programkod	Kommentar
...	
N70 X50 SPOF	; Positionera utan stansutlösning.
N80 X100 SON	; Aktivera nibbling, utlösning av ett slag före rörelsen (X=50) och i slutet av den programmerade rörelsen (X=100).
...	

Exempel 2: Stansning med fördröjning

Programkod	Kommentar
...	
N170 PDELAYON X100 SPOF	; Positionera utan stansutlösning, aktivering av den fördröjda stansutlösningen.
N180 X800 PON	; Aktivera stansning. Efter det slutpositionen uppnått matas stansslaget ut fördröjt.
N190 PDELAYOF X700	; Inaktivera stansning med fördröjning, normal stansutlösning i slutet av den programmerade rörelsen.
...	

Exempel 3: Stansning med två stans-interfaces

Programkod	Kommentar
...	
N170 SPIF1 X100 PON	; I slutet av blocket sker en slagutlösning på den första snabba utgången. Signalen "Slag aktivt" övervakas på den första ingången.
N180 X800 SPIF2	; Den andra slagutlösningen sker på den andra snabba utgången. Signalen "Slag aktivt" övervakas på den andra ingången.
N190 SPIF1 X700	; Slagstyrningen för alla ytterligare slag sker med den första interfacen.
...	

Ytterligare informationer

Stansning och nibbling med inledning (PONS/SONS)

Stansning och nibbling med inledning är inte möjligt i flera kanaler samtidigt. PONS resp. SONS kan alltid aktiveras endast i en kanal.

Vägberoende acceleration (PUNCHACC)

Exempel:

PUNCHACC (2, 50, 10, 100)

Hållavstånd under 2mm:

Det körs med en acceleration på 50% av maximalaccelerationen.

Hålavstånd på 2mm till 10mm:

Accelerationen stegras proportionellt med avståndet till 100%.

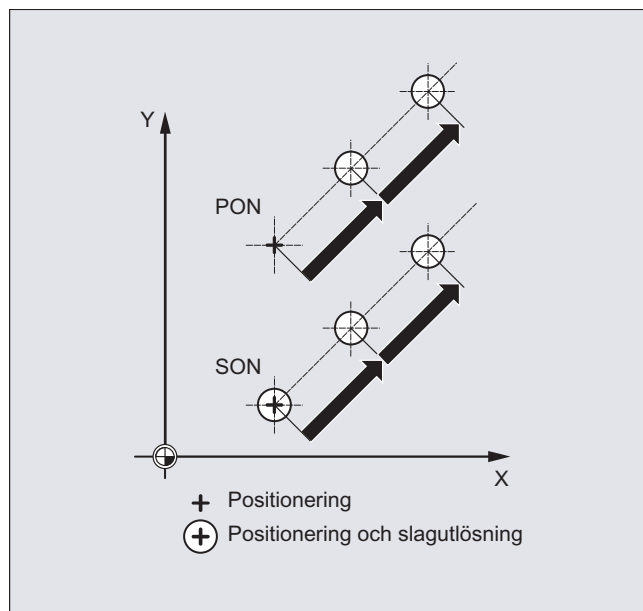
Hålavstånd större än 10mm:

Förflyttning med en acceleration på 100%.

Utlösning av första slaget

Utlösningen av det första slaget efter aktivering av funktionen sker vid nibbling och stansning tidsmässigt olika:

- PON/PONS:
 - Alla slag - också det i det första blocket efter aktivering - sker i blockslutet.
- SON/SONS:
 - Det första slaget efter aktivering av nibblingen sker redan i blockbörjan.
 - Alla ytterligare slag utlöses alltid i blockslutet.



Stansning och nibbling på stället

En slagutlösning sker endast när blocket innehåller en förflyttningssinformation för stans- eller nibbelaxlarna (axlar i det aktiva planet).

För att trots detta utlösa ett slag på samma ställe, programmeras en av stans-/nibbelaxlarna med förflyttning 0.

Arbeta med vridbara verktyg

Märk

För att ansätta vridbara verktyg tangentiellt till den programmerade banan, använder du tangentialstyrningen.

Användning av M-kommandon

Med hjälp av makroteknik är det fortfarande möjligt, att använda speciella M-funktioner i stället för språkkommandon (kompatibilitet). Därvid gäller följande motsvarigheter till äldre system:

M20, M23	△	SPOF
M22	△	SON
M25	△	PON
M26	△	PDELAYON

Exempel för makrofil:

Programkod	Kommentar
DEFINE M25 AS PON	; Stansning till
DEFINE M125 AS PONS	; Stansning med inledning till
DEFINE M22 AS SON	; Nibbling till
DEFINE M122 AS SONS	; Nibbling med inledning till
DEFINE M26 AS PDELAYON	; Stansning med fördröjning till
DEFINE M20 AS SPOF	; Stansning, nibbling från
DEFINE M23 AS SPOF	; Stansning, nibbling från

Programmeringsexempel:

Programkod	Kommentar
...	
N100 X100 M20	; Positionera utan stansutlösning.
N110 X120 M22	; Aktivera nibbling, före och efter rörelse slagutlösning.
N120 X150 Y150 M25	; Aktivera stansning, slagutlösning i slutet av rörelsen.
...	

16.2 Automatisk väguppdelning

Indelning i delsträckor

Vid aktiverad stansning resp. nibbling förorsakar både SPP och även SPN en uppdelning av den för banaxlarna programmerade totala förflyttningssträckan i ett antal lika långa delsträckor (ekvidistant väguppdelning). Internt motsvarar varje delsträcka ett block.

Antal slag

Vid stansningen sker det första slaget i slutpunkten av den första delsträckan, vid nibbling däremot vid startpunkten för den första delsträckan. Över den totala förflyttningssträckan resulterar därmed följande tal:

Stansning: Antal slag = antal delsträckor

Nibbling: Antal slag = antal delsträckor + 1

Hjälpfunktioner

Hjälpfunktioner utförs i det första av de skapade blocken.

Syntax

SPP=

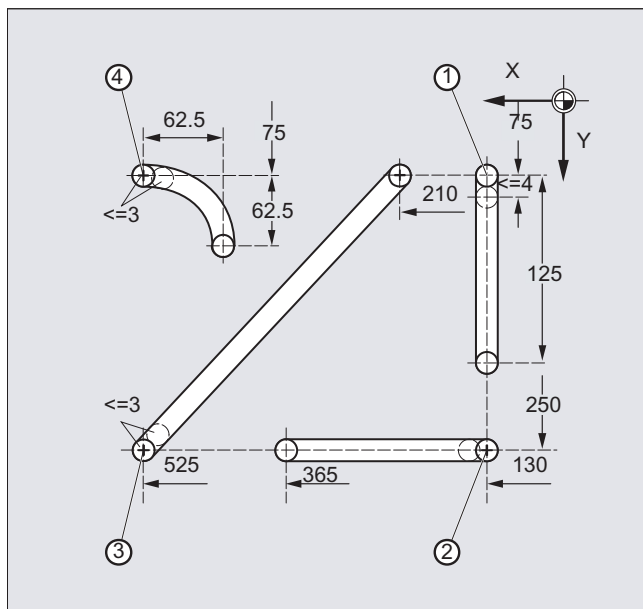
SPN=

Betydelse

SPP:	Storlek på delsträckan (maximalt slagavstånd); modalt verksam
SPN:	Antal delsträckor per block; blockvis verksam

Exempel 1

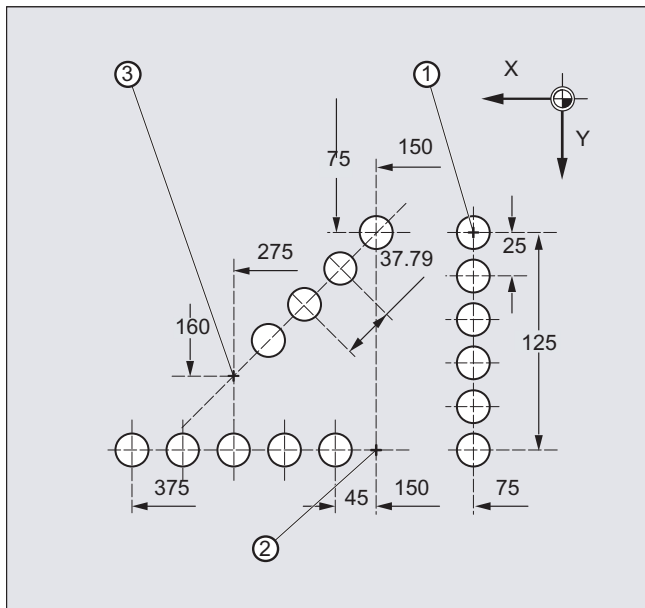
De programmerade nibbelsträckorna ska delas upp automatiskt i lika stora delsträckor.



Programkod	Kommentar
N100 G90 X130 Y75 F60 SPOF	; Positionera på startpunkt 1
N110 G91 Y125 SPP=4 SON	; Nibbling till; maximal delsträck- längd för automatisk väguppdelning: 4 mm
N120 G90 Y250 SPOF	; Nibbling från; positionera på startpunkt 2
N130 X365 SON	; Nibbling till; maximal delsträck- längd för automatisk väguppdelning: 4 mm
N140 X525 SPOF	; Nibbling från; positionera på startpunkt 3
N150 X210 Y75 SPP=3 SON	; Nibbling till; maximal delsträck- längd för automatisk väguppdelning: 3 mm
N160 X525 SPOF	; Nibbling från; positionera på startpunkt 4
N170 G02 X-62.5 Y62.5 I J62.5 SPP=3 SON	; Nibbling till; maximal delsträck- längd för automatisk väguppdelning: 3 mm
N180 G00 G90 Y300 SPOF	; Nibbling från

Exempel 2

För de enskilda hålraderna ska en automatisk väguppdelning göras. För uppdelningen anges alltid den maximala delsträcklängden (SPP-värde).



Programkod	Kommentar
N100 G90 X75 Y75 F60 PON	; Positionera på startpunkt 1; stansning till stansa enkelhål
N110 G91 Y125 SPP=25	; Maximal delsträcklängd för automatisk väguppdelning: 25 mm
N120 G90 X150 SPOF	; Stansning från; positionera på startpunkt 2
N130 X375 SPP=45 PON	; Stansning till; maximal delsträcklängd för automatisk väguppdelning: 45 mm
N140 X275 Y160 SPOF	; Stansning från; positionera på startpunkt 3
N150 X150 Y75 SPP=40 PON	; Stansning till; i stället för den programmerade del- sträcklängden på 40 mm blir den ;beräknade delsträck- längden på 37,79 mm använd.
N160 G00 Y300 SPOF	; Stansning från; positionera

16.2.1 Väguppdelning vid banaxlar

Längd på delsträcka SPP

Med **SPP** anger du det maximala slagavståndet och därmed den maximala längden på delsträckor, i vilka den totala förflyttningssträckan ska delas upp. Frånkopplingen av kommandot sker med **SPOF** eller **SPP=0**.

Exempel:

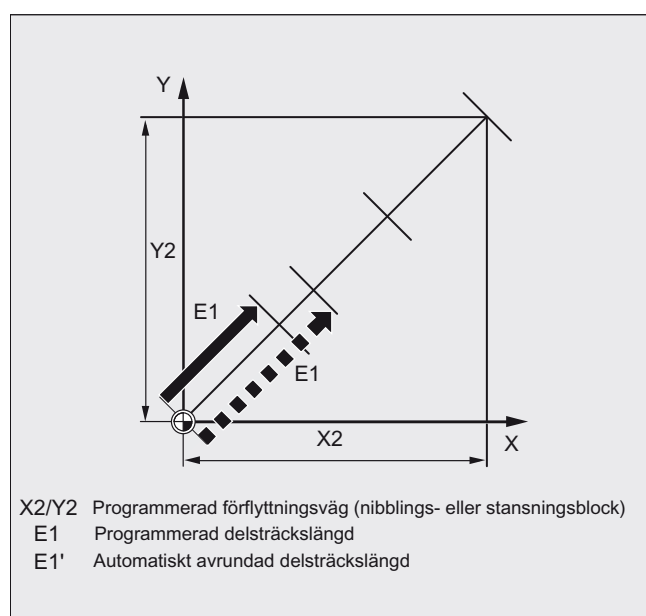
```
N10 SON X0 Y0
```

```
N20 SPP=2 X10
```

Den totala förflyttningssträckan på 10 mm delas upp i 5 delsträckor på var och en 2 mm ($SPP=2$).

Märk

Väguppdelningen med SPP görs alltid ekvidistant: alla delsträckor är lika långa. Det betyder, den programmerade delsträckstorleken (värde på SPP) är giltig endast när kvoten av den totala förflyttningssträckan och SPP -värdet är ett helt tal. Är detta inte fallet så reduceras internt storleken på delsträckan så att en heltalig kvot resulterar.



Exempel:

```
N10 G1 G91 SON X10 Y10
```

```
N20 SPP=3.5 X15 Y15
```

Vid den totala förflyttningssträckan på 15 mm och en delsträckslängd på 3,5 mm resulterar en ej heltalig kvot (4.28). Därmed sker en reduktion av SPP -värdet till närmsta möjliga heltaliga kvot. I detta fall resulterar en delsträckslängd på 3 mm.

Antal delsträckor SPN

Med SPN definierar du antalet delsträckor, som ska skapas ur den totala förflyttningssträckan. Längden på delsträckorna beräknas automatiskt. Eftersom SPN är blockvis verksam, måste dessförinnan stansning eller nibbling aktiveras med PON eller SON .

16.2.2 Väguppdelning vid singelaxlar

Finns förutom banaxlarna också singelaxlar definierade som stans-nibbel-axlar, så kan också de vara underordnade den automatiska väguppdelningen.

Beteende hos singelaxeln vid SPP

Den programmerad längden på delsträckan (SPP) hänför sig principiellt till banaxlarna. Därför ignoreras i ett block, i vilket förutom singelaxelrörelse och SPP-värdet ingen banaxel är programmerad, SPP-värdet.

Finns både singel- och även banaxel programmerade i blocket, så rättar sig singelaxelns beteende efter inställningen i motsvarande maskindatum.

1. Standardinställning

Vägen för singelaxeln fördelas likformigt på de av SPP skapade intermediära blocken.

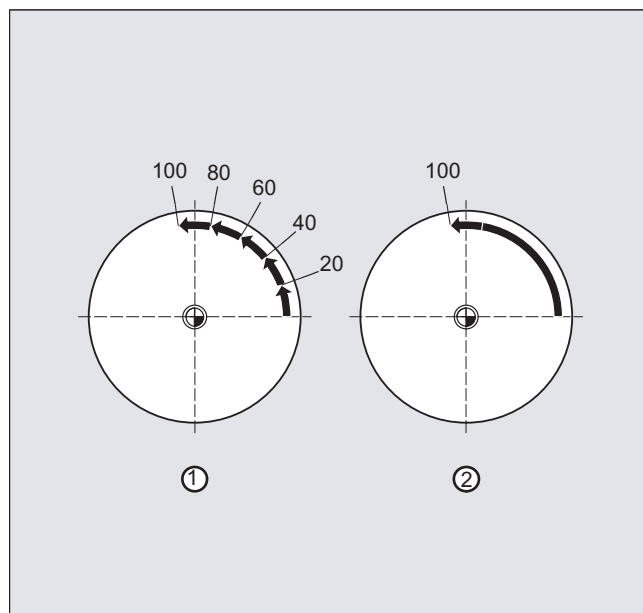
Exempel:

```
N10 G1 SON X10 A0
```

```
N20 SPP=3 X25 A100
```

Med slagsträckan på 3 mm skapas vid den totala förflyttningssträckan för X-axeln (banaxel) på 15 mm 5 block.

A-axeln vrider sig därmed 20° i varje block.



1. Singelaxel utan väguppdelning

Singelaxeln kör sin totala väg i det första av de skapade blocken.

2. Olika väguppdelning

Singelaxelns beteende är beroende av interpoleringen av banaxlarna:

- Cirkelinterpolering: väguppdelning
- Linjärinterpolering: ingen väguppdelning

Beteende vid SPN

Det programmerade antalet delsträckor gäller också när en banaxel inte är programmerad samtidigt.

Förutsättning: Singelaxeln är definierad som stans-nibbel-axel.

Slipning

17.1 Koppla till/från slipspecifik verktygsövervakning (TMON, TMOF)

Med de fördefinierade procedurerna TMON...) och TMOF...) kopplas den slipspecifika verktygsövervakningen (geometri- och varvtalsövervakning) till resp. från.

Förutsättning

De verktygsspecifika parametrarna \$TC_TPG1 bis \$TC_TPG9 måste vara inställda.

Syntax

```
TMON (<Tnr>)
...
TMOF (<Tnr>)
```

Betydelse

TMON (. . .) :	Koppla till slipspecifik verktygsövervakning Kommandot måste programmeras i den kanal i vilken den slipspecifika verktygsövervakningen ska kopplas till.
TMOF (. . .) :	Koppla från slipspecifik verktygsövervakning Kommandot måste programmeras i den kanal i vilken den slipspecifika verktygsövervakningen ska kopplas från.
<Tnr>:	T-nummer Observera: Endast erforderligt när övervakningen i stället för det aktiva verktyget som är under användning, ska kopplas till resp. från för en ej aktiv slipskiva.
TMOF (0) :	Koppla från övervakning för alla verktyg

Ytterligare funktioner

18.1 Axelfunktioner (AXNAME, AX, SPI, AXTOSPI, ISAXIS, AXSTRING, MODAXVAL)

"AXNAME" används t.ex. vid upprättandet av allmängiltiga cykler, när namnen på axlarna inte är kända.

"AX" används för den indirekta programmeringen av geometri- och synkronaxlar. Axelbeteckningen lagras därvid i en variabel av typ AXIS eller levereras av ett kommando som "AXNAME" eller "SPI".

"SPI" används när axelfunktionerna för en spindel t.ex. synkronspindel programmeras.

"AXTOSPI" används för att omvandla en axelbeteckning till ett spindelindex (inverterad funktion till SPI).

"AXSTRING" används för att omvandla en axelbeteckning (datatyp AXIS) till en string (inverterad funktion till "AXNAME").

"ISAXIS" används i allmängiltiga cykler för att säkerställa att en viss geometriaxel finns och därmed ett efterföljande anrop av \$P_AXNX inte avbryts med fel.

"MODAXVAL" används för att vid roterande modulo-axlar fastställa modulo-positionen.

Syntax

```
AXNAME ("String")
AX[AXNAME ("String")]
SPI (n)

AXTOSPI (A) eller AXTOSPI (B) eller AXTOSPI (C)
AXSTRING (SPI (n))
ISAXIS (<Geometrieachsnummer>)
<Modulo-Position>=MODAXVAL (<Achse>, <Achsposition>)
```

Betydelse

AXNAME:	Konverterar en ingångsstring i axelbeteckning; ingångsstringen måste innehålla giltigt axelnamn.
AX:	Variabel axelbeteckning
SPI:	Konverterar spindelnummer i axelbeteckning; överföringsparametern måste innehålla ett giltigt spindelnummer.
n:	Spindelnummer
AXTOSPI:	Omvandlar en axelbeteckning i ett spindelindex av typ Integer. "AXTOSPI" motsvarar den inverterade funktionen till "SPI".
X, Y, Z:	Axelbeteckning av typ AXIS som variabel eller konstant
AXSTRING:	String med tillordnat spindelnummer matas ut.

ISAXIS:	Kontrollerar om den angivna geometriaxeln finns.
MODAXVAL:	Fastställer vid roterande modulo-axlar modulo-positionen; denna motsvarar modulo-resten relaterad till det parameterade modulo-området (uppgår i standardinställningen till 0 till 360 grader; via MD30340 MODULO_RANGE_START och MD30330 \$MA_MODULO_RANGE kan början och storlek på modulo-området förändras).

Märk**SPI-tillägg**

Axelfunktionen SPI(n) kan också användas för läsning och skrivning av framekomponenter. Därmed kan frames t.ex. skrivas med syntaxen `$P_PFRAME[SPI(1),TR]=2.22`.

Genom den extra programmeringen av axelpositioner via adressen

`AX[SPI(1)]=<Achsposition>` kan en axel förflyttas. Förutsättning för detta är att spindeln befinner sig i positionerings- eller axeldrift.

Exempel**Exempel 1: AXNAME, AX, ISAXIS**

Programkod	Kommentar
<code>OVRA[AXNAME("Planachse")]=10</code>	; Övermannig för planaxel
<code>AX[AXNAME("Planachse")]=50.2</code>	; Slutposition för planaxel
<code>OVRA[SPI(1)]=70</code>	; Övermannig för spindel 1
<code>AX[SPI(1)]=180</code>	; Slutposition för spindel 1
<code>IF ISAXIS(1)==FALSE GOTOF WEITER</code>	; Finns abskissa?
<code>AX[\$P_AXN1]=100</code>	; Flytta abskissa
<code>WEITER:</code>	

Exempel 2: AXSTRING

Vid programmeringen med `AXSTRING[SPI(n)]` matas inte längre axelindex för den axel som är tillordnad spindeln ut som spindelnummer, utan stringen "Sn" matas ut.

Programkod	Kommentar
<code>AXSTRING[SPI(2)]</code>	; String "S2" matas ut.

Exempel 3: MODAXVAL

Modulo-positionen för der roterande modulo-axeln ska fastställas.

Utgångsvärde för beräkningen är axelpositionen 372.55.

Det parameterade modulo-området uppgår till 0 till 360 grader:

`MD30340 MODULO_RANGE_START = 0`

`MD30330 $MA_MODULO_RANGE = 360`

Programkod	Kommentar
<code>R10=MODAXVAL(A,372.55)</code>	; Beräknad modulo-position R10 = 12.55.

Exempel 4: MODAXVAL

När den programmerade axelbeteckningen inte hänför sig till en roterande modulo-axel, då returneras värdet (<Achsposition>) som ska omvandlas oförändrat.

Programkod	Kommentar
R11=MODAXVAL(X,372.55)	; X är linjäraxel; R11 = 372.55.

18.2 Omkopplingsbara geometriaxlar (GEOAX)

Med funktionen "Omkopplingsbara geometriaxlar" kan de via maskindata parametrerade geometriaxlarna ersättas av andra kanalaxlar.

Syntax

```
GEOAX (<n>, <Kanalachse>, <n>, <Kanalachse>, <n>, <Kanalachse>)
GEOAX ()
```

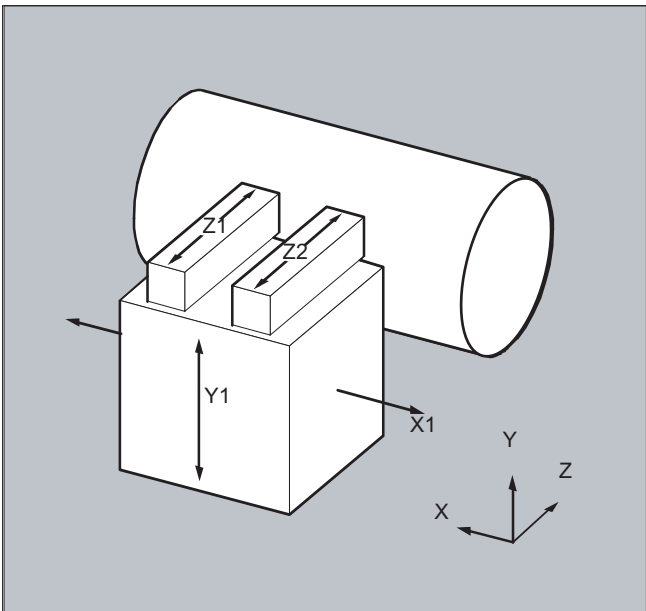
Betydelse

GEOAX (. . .)	Funktion för omkoppling av geometriaxlar. Observera: GEOAX () utan parameteruppgift aktiverar åter den i maskindata parametrerade grundkonfigurationen för geometriaxlarna.
<n>	Nummer för den geometriaxel, som ersätts av den angivna kanalaxeln. Värdeområde: 0, 1, 2, 3 Observera: 0: Den angivna kanalaxeln avlägsnas utan ersättning från geometriaxelförbandet 1: 1:a geometriaxeln $\triangleq$ koordinataxel X (abskissa) i WKS 2: 2:a geometriaxeln $\triangleq$ koordinataxel Y (ordinata) i WKS 3: 3:e geometriaxeln $\triangleq$ koordinataxel Z (applikata) i WKS
<Kanalachse>	Namnet på den kanalaxel som ska tas upp i geometriaxelförbandet.

Exempel

Exempel 1: Koppla två axlar omväxlande som geometriaxel

En verktygsslid kan förflyttas via kanalaxlarna X1, Y1, Z1, Z2:



Geometriaxlarna är så projekterade att efter tillkopplingen först Z1 är verksam som 3:e geometriaxel under geometriaxelnamnet "Z" och tillsammans med X1 och Y1 bildar geometriaxelförbandet.

I detaljprogrammet ska nu axlarna Z1 och Z2 omväxlande användas som geometriaxel Z:

Programkod	Kommentar
...	
N100 GEOAX(3,Z2)	; Som 3:e geometriaxel (Z) fungerar kanalaxel Z2.
N110 G1 ...	
N120 GEOAX(3,Z1)	; Som 3:e geometriaxel (Z) fungerar kanalaxel Z1.
...	

Exempel 2: Omkoppling av geometriaxlarna vid 6 kanalaxlar

En maskin har 6 kanalaxlar med namnen XX, YY, ZZ, U, V, W.

Grundinställningen av geometriaxelkonfigurationen via maskindata är:

Kanalaxel XX = 1:a geometriaxel (X-axel)

Kanalaxel YY = 2:a geometriaxeln (Y-axel)

Kanalaxel ZZ = 3:e geometriaxeln (Z-axel)

Programkod	Kommentar
N10 GEOAX()	; Grundkonfiguration för geometriaxlarna är verksam.
N20 G0 X0 Y0 Z0 U0 V0 W0	; Alla axlar i snabbtransport till position 0.
N30 GEOAX(1,U,2,V,3,W)	; Kanalaxel U blir första (X), V andra (Y) ; och W tredje geometriaxel (Z).
N40 GEOAX(1,XX,3,ZZ)	; Kanalaxel XX blir första (X), ZZ tredje ; geometriaxel (Z). Kanalaxel V förblir andra ; geometriaxel (Y).

18.2 Omkopplingsbara geometriaxlar (GEOAX)

Programkod	Kommentar
N50 G17 G2 X20 I10 F1000	; Sluten cirkel i X/Y-planet. Flyttning av ; kanalaxlarna XX och V.
N60 GEOAX(2,W)	; Kanalaxel W blir andra geometriaxel (Y).
N80 G17 G2 X20 I10 F1000	; Sluten cirkel i X/Y-planet. Flyttning av ; kanalaxlarna XX och W.
N90 GEOAX()	; Återställning till grundtillstånd.
N100 GEOAX(1,U,2,V,3,W)	; Kanalaxel U blir första (X), V andra ; (Y) och W tredje geometriaxel (Z).
N110 G1 X10 Y10 Z10 XX=25	Kanalaxlarna U, V, W går var och en till ; position 10. XX som extraaxel går till position 25.
N120 GEOAX(0,V)	; V tas ur geometriaxelförbandet. ; U och W är fortfarande första (X) och tredje ; geometriaxel (Z). ; Den andra geometriaxeln (Y) förblir obelagd.
N130 GEOAX(1,U,2,V,3,W)	Kanalaxel U förblir första (X), V blir andra (Y) och W förblir tredje geometriaxel (Z).
N140 GEOAX(3,V)	V blir tredje geometriaxel (Z), varvid W ; skrivs över och därmed tas ur geometriaxelförbandet ; . ; Den andra geometriaxeln (Y) ; förblir obelagd.

Maskindata

Axelkonfiguration

Tillordning av geometri-, extra- och maskinaxlar till kanalaxlar:

- MD10000 \$MN_AXCONF_MACHAX_NAME_TAB
- MD20050 \$MC_AXCONF_GEOAX_ASSIGN_TAB
- MD20060 \$MC_AXCONF_GEOAX_NAME_TAB
- MD20070 \$MC_AXCONF_MACHAX_USED
- MD20080 \$MC_AXCONF_CHANAX_NAME_TAB
- MD35000 \$MA_SPIND_ASSIGN_TO_MACHAX

Reset-beteende

Reset-beteende för ändrade geometriaxeltillordningar:

- MD20110 \$MC_RESET_MODE_MASK, Bit 12
- MD20118 \$MC_GEOAX_CHANGE_RESET

NC-start-beteende

- MD20112 \$MC_START_MODE_MASK, Bit 12

Meddelande till PLC-användarprogram

Parametermöjlighet för M-kommandot som vid en geometriaxelomkoppling matas ut vid NC/PLC-gränssnittet:

- MD22532 \$MC_GEOAX_CHANGE_M_CODE

Randvillkor**Ingen geometriaxelomkoppling**

- Är en av de följande funktionerna är aktiv, är ingen geometriaxelomkoppling möjlig:
 - Transformation
 - Spline-interpolering
 - Verktygsradiekompensering
 - Verktygsfinkompensering
- Geometriaxeln och en ytterligare kanalaxel har samma namn.
- En av axlarna som deltar i geometriaxelomkopplingen deltar i en aktion, som varar utanför blockgränserna. T. ex. blockövergripande positioneringsaxel eller följdaxel i en axelkoppling.

Roterande axlar

Roterande axlar kan inte göras till geometriaxlar.

Axeltillstånd efter ersättningen

En genom omkopplingen i geometriaxelförbandet ersatt axel, kan efter omkopplingsförloppet programmeras via sitt kanalaxelnamn som extraaxel.

Frames, skyddsområden, arbetsfältbegränsningar

Med omkopplingen av geometriaxlarna raderas alla frames, skyddsområden och arbetsfältbegränsningar.

Polarkoordinater

Ett byte av geometriaxlar med GEOAX sätter analogt med ett planbyte med G17-G19 de modala polarkoordinaterna på värdet 0.

DRF, NPV

En eventuell handratt-förflyttning (DRF) eller en extern nollpunktsförflyttning (NPV) förblir verksam efter omkopplingen.

Grundkonfiguration för geometriaxlarna

Kommandot GEOAX () anropar grundkonfigurationen för geometriaxelförbandet.

Efter POWER ON och vid omkoppling till driftsläget "Referenspunktkörning" återkopplas automatiskt till grundkonfigurationen.

Verktyslängdkompensering

En aktiv verktyslängdkompensering är verksam också efter omkopplingsförloppet. Den gäller dock för de nyupptagna resp. positionsbytta geometriaxlarna som ännu inte utkörd. Vid första rörelsekommandot för dessa geometriaxlar består den resulterande förflyttningsvägen till följd därav av summan av verktyslängdkompensering och programmerad förflyttningsväg.

Geometriaxlar som vid en omkoppling bibehåller sin position i axelförbandet, behåller också sin status beträffande verktyslängdkompenseringen.

Geometriaxelkonfiguration vid aktiv transformation

- Den för en aktiv transformation via transformations-maskindata parametrerad geometriaxelkonfiguration kan inte förändras via funktionen "Omkopplingsbara geometriaxlar".
- Olika geometriaxelkonfiguration för en transformation; olika datablock måste parametreras i transformations-maskindata.
- En via **GEOAX** förändrad geometriaxelkonfiguration raderas genom aktivering av en transformation.
- Beträffande geometriaxlarna har den transformations-specifika geometriaxelparametreringen av aktiva transformationer företräde före omkopplingen av geometriaxlar gällande parametreringen.
Exempel: En transformation är aktiv Enligt maskindata ska transformationen bibehållas vid kanal-reset. Samtidigt ska vid en kanal-reset dock grundkonfigurationen för geometriaxlarna upprättas. Den geometriaxelkonfiguration bibehålls, som fastlades för transformationen.
- Med fränkopplingen av en transformation blir åter den parametrerade grundinställningen för geometriaxelkonfigurationen verksam.

Driftsläge JOG, maskinfunktion REF

Vid omkopplingen till driftsläge JOG, maskinfunktion REF (köra referenspunkt), blir den i maskindata parametrerade geometriaxelkonfigurationen verksam.

18.3 Axelcontainer (AXCTSWE, AXCTSWED, AXCTSWEC)

Via kommandona "AXCTSWE" resp. "AXCTSWED" friges vridningen av den angivna axelcontainern.

Via kommandot "AXCTSWEC" återkallas en redan tilldelad frigivning för axelcontainer-vridning.

Syntax

AXCTSWE (<ID>)
 AXCTSWED (<ID>)
 AXCTSWEC (<ID>)

Betydelse

AXCTSWE:	Frigivning för vridning av axelcontainern Programbearbetningen stoppas inte av "AXCTSWE". Vridningen sker så snart som alla i axelcontainern deltagande kanalerna har utdelat frigivningen.	
AXCTSWED:	Frigivning för vridning av axelcontainern utan att ta hänsyn till andra i axelcontainern deltagande kanaler Observera - Kommandovariant till förenkling av idrifttagandet av detaljprogrammet resp. synkronaktion. - Beteendet beträffande de andra i axelcontainern deltagande kanalerna kan föreskrivas via: MD12760 \$MN_AXCT_FUNCTION_MASK, Bit 0	
AXCTSWEC:	Återkallande av frigivningen för vridning av axelcontainern Observera Frigivningen för vridning av axelcontainern kan endast återkallas, när vridningen ännu inte har påbörjats: \$AN_AXCTSWA[<Achsccontainer>] == 0 För systemvariabel se "Axelcontainer (AXCTSWE, AXCTSWED, AXCTSWEC) (Sida 633)"	
<ID>:	Beteckning för axelcontainern eller en containeraxel:	
	CT<Nummer>:	Defaultbeteckning för en axelcontainer: MD12750 \$MN_AXCT_NAME_TAB Exempel: "CT1"
	<Container>:	Användarspecifik beteckning för en axelcontainer: MD12750 \$MN_AXCT_NAME_TAB Exempel: "CONTAINER_1"
	<Achse>:	Beteckning för en i kanalen känd containeraxel

Märk**Steglängd**

Steglängden för en axelcontainer-vridning ställs in via settingdatum:

SD41700 \$SN_AXCT_SWWIDTH

Ytterligare informationer**Diagnos**

Den aktuella statusen för en axelcontainer kan läsas via följande systemvariabel:

Systemvariabel	Typ	Beskrivning
\$AC_AXCTSWA[<Name>]	BOOL	Kanal-specifik status för axelcontainern
\$AN_AXCTSWA[<Achscontainer>]	BOOL	NCU-specifik status för axelcontainern
\$AN_AXCTSWE[<Achscontainer>]	INT	Slot-specifik status för axelcontainer-vridningen Systemvariabeln levererar bit-vis status för slotten i axelcontainern. Varje bit motsvarar en slot.
\$AN_AXCTAS[<Achscontainer>]	INT	Antal platser (slots) med vilka axelcontainer aktuellt kopplades vidare.

Achscontainer-vridning med implicit GET/GETD

Via efterföljande maskindatum kan ställas in att med kommandot "AXCTSWE" alla kanalens containeraxlar kan hämtas till kanalen med hjälp av implicita "GET / GETD". Ett axelbyte är åter möjligt först efter utförd containervridning.

MD10722 \$MN_AXCHANGE_MASK, Bit 1 = 1

Märk

Axelcontainervridningen med implicita "GET / GETD" genomförs **inte** för en axel i tillståndet "Huvudkörningsaxel" (t. ex. PLC-axel), eftersom axeln för axelcontainervridningen skulle vara tvungen att lämna tillståndet.

18.4 Vänta på giltig axelposition (WAITENC)

Med språkkommandot "WAITENC" kan väntas i NC-programmet tills för de med MD34800 \$MA_WAIT_ENC_VALID = 1 projekterade axlarna synkroniserade resp. restaurerade axelpositioner står till förfogande.

I väntetillståndet kan ett stopp ske t.ex. genom start av en ASUP eller genom driftslägesbyte efter JOG. Med fortsättningen av programmet återtas ev. väntetillståndet igen.

Märk

Väntetillståndet visas i användargränssnittet av hållningstillståndet "Vänta på mät-system".

Syntax

"WAITENC" kan programmeras i programdelen till ett valfritt NC-program.

Programmeringen måste göras i ett eget block:

```
...
WAITENC
...
```

Exempel

"WAITENC" används t.ex. i det händelsestyrda användarprogrammet .../_N_CMA_DIR/_N_PROG_EVENT_SPF som det följande användningsexemplet visar.

Användningsexempel: Verktygsåtergång efter POWER OFF med orienteringstransformation

En bearbetning med verktygsorientering avbröts genom strömavbrott.

Vid anslutande start anropas det händelsestyrda användarprogrammet .../_N_CMA_DIR/_N_PROG_EVENT_SPF.

I det händelsestyrda användarprogrammet väntas med "WAITENC" på synkroniserade resp. restaurerade axelpositioner för att därefter kunna beräkna en frame, som riktar WKS i verktygsriktningen.

Programkod	Kommentar
...	
IF \$P_PROG_EVENT == 4	; Start.
IF \$P_TRAFO <> 0	; Transformation tillvaldes.
WAITENC	; Vänta på giltiga axelpositioner för orienteringsaxlarna.
TOROTZ	; Vrida Z-axeln i WKS i verktygsaxelns riktning.
ENDIF	
M17	
ENDIF	
...	

Därefter kan verktyget i driftsläget JOG friköras genom en återgångsrörelse i verktygsaxelns riktning.

18.5 Programmerbar parameterblockomkoppling (SCPARA)

Med kommandot "SCPARA" kan omkopplingen till ett bestämt parameterblock för en axel krävas.

Märk

Ingen parameterblockomkoppling under gängbearbetning

Vid gängskärning G33 och gängtappning G331 / G332 väljs parameterblocket av styrningen och kan inte ändras.

Spärrad parameterblockomkoppling

En parameterblockomkoppling kan också krävas via NC/PLC-gränssnittet. För att undvika omkopplingskonflikter, kan parameterblockomkopplingen för NC (SCPARA) spärras via NC/PLC-gränssnittet:

DB31, ... DBX9.3 (parameterblockuppgift spärrad av NC)

Märk

Krävs en parameterblockomkoppling av "SCPARA" under det att parameterblockomkopplingen är spärrad via NC/PLC-gränssnittet, avvisas omkopplingen utan felmeddelande.

Syntax

SCPARA[<Achse>] = <Wert>

Betydelse

SCPARA:	Kommando: Koppla om parameterblock	
<Achse>:	Axelbeteckning (kanalaxel)	
	Typ:	AXIS
<Wert>:	Parameterblocknummer: 1, 2, 3, ... max. parameterblocknummer	

Exempel

Programkod	Kommentar
...	
N110 SCPARA[X] = 3	; Val: Axel X, 3:e parameterblocket
...	

Ytterligare informationer

Frigivning av parameterblockomkoppling

Parameterblockomkopplingen för axeln måste frigivas explicit:

MD35590 \$MA_PARAMSET_CHANGE_ENABLE[<Achse>]

Läsa parameterblocknummer

Numret för det valda parameterblockets (bör-parameterblock) kan läsas via systemvariabeln \$AA_SCPAR.

Litteratur

Utförliga informationer till parameterblocke finns i:

Funktionshandbok Grundfunktioner; Kapitel "Hastigheter, bör-/ärvärdessystem, reglering (G2)" > "Reglering" > "Parameterblock för lägesregleraren"

18.6 Kontrollera förefintlig NC-språkomfattning (STRINGIS)

Med funktionen "STRINGIS(...)" kan kontrolleras om den angivna stringen står till förfogande som element i NC-programmeringsspråket i den aktuella språkomfattningen.

Definition

```
INT STRINGIS (STRING <Name>)
```

Syntax

```
STRINGIS (<Name>)
```

Betydelse

STRINGIS:	Funktion med returvärde
<Namn>:	Namn på det element i NC-programmeringsspråket som ska kontrolleras
Returvärde:	Formatet för returvärdet är yxx (decimalt).

Element i NC-programmeringsspråket

Följande element i NC-programmeringsspråket kan kontrolleras:

- G-kommandon i alla existerande G-grupper t.ex. "G0", "INVCW", "POLY", "ROT", "KONT", "SOFT", "CUT2D", "CDON", "RMBBL", "SPATH"
- DIN- eller NC-adresser som t.ex. "ADIS", "RNDM", "SPN", "SR", "MEAS"
- Funktioner t.ex. "TANG(...)" eller "GETMDACT"
- Procedurer t.ex. "SBLOF".
- Nyckelord t.ex. "ACN", "DEFINE" eller "SETMS"
- Systemdata t.ex. maskindata \$M... , settingdata \$S... eller optionsdata \$O...
- Systemvariabel \$A... , \$V... , \$P...
- Räkneparametrar R...
- Cykelnamn för aktiverade cykler
- GUD- och LUD-variabler
- Makro-namn
- Label-namn

Retur-värde

Returvärdet är relevant endast i de första 3 decimalplatserna. Formatet för returvärdet är yxx, med y = basinformation och xx = detaljinformation.

Returvärde	Betydelse
000	String 'name' är inte känt i det föreliggande systemet ¹⁾
100	String 'name' är ett element i NC-programmeringsspråket, men aktuellt inte programmerbar (option/funktion är inaktiv)

18.6 Kontrollera förefintlig NC-språkomfattning (STRINGIS)

Returvärde	Betydelse
2xx	String 'name' är ett programmerbart element i NC-programmeringsspråket (option/funktion är aktiv). Detaljinformationen xx innehåller ytterligare informationer över typen av element:
	xx Betydelse
	01 DIN-adress eller NC-adress ²⁾
	02 G-kommando (t.ex. G04, INVCW)
	03 Funktion med returvärde
	04 Funktion utan returvärde
	05 Nyckelord (t.ex. DEFINE)
	06 Maskin- (\$M...), setting- (\$S...) eller optionsdatum (\$O...)
	07 Systemparameter, t.ex. systemvariabel (\$...) eller räkneparameter (R...)
	08 Cykel (Cykeln måste vara laddad i NC och vara aktiv i cykelprogrammen ³⁾)
	09 GUD-variabel (GUD-variabeln måste vara definierad i GUD-definitionsfiler och GUD-variablerna vara aktiverade)
	10 Makronamn (Makron måste vara definierad i makro-definitionsfilerna och makros vara aktiverade) ⁴⁾
	11 LUD-variabel för det aktuella detaljprogrammet
	12 ISO G-kommando (ISO språkmode måste vara aktivt)
400	String 'name' är en NC-adress, som inte identifierades som xx == 01 eller xx == 10 och inte är G eller R ²⁾
y00	Ingen specifik tillordning möjlig

1) Känd som styrningsberoende är eventuellt endast en undermängd i Siemens NC-språkkommandon, t.ex. SINUMERIK 802D sl. På dessa styrningar returneras för strings, som principiellt är NC-språkkommandon värdet 0. Detta beteende kan ändras via MD10711 \$MN_NC_LANGUAGE_CONFIGURATION. Vid MD10711 = 1 returneras då alltid värdet 100 för Siemens NC-språkkommandon.

2) NC-adresser är följande bokstäver: A, B, C, E, I, J, K, Q, U, V, W, X, Y, Z. Dessa NC-adresser kan också programmeras med adresstillägg. Adresstillägget kan anges vid kontrollen med STRINGIS. Exempel: 201 == STRINGIS("A1"). Bokstäverna: D, F, H, L, M, N, O, P, S, T är NC-adresser eller hjälpfunktioner som används användardefinierat. För dem returneras alltid värdet 400. Exempel: 400 == STRINGIS("D"). Dessa NC-adresser kan vid kontrollern med STRINGIS inte anges med adresstillägg. Exempel: 000 == STRINGIS("M02"), aber 400 == STRINGIS("M").

3) Namn på cykelparametrar kan inte kontrolleras med STRINGIS.

4) Som makro definierad adress t.ex. G, H, M, L identifieras som makro

Exempel

I de följande exemplen antas att de som string angivna NC-språkelementen, såvida inget speciellt är noterat, är principiellt programmerbara i styrningen.

1. String "T" är definierad som hjälpfunktion:

```
400 == STRINGIS ("T")
000 == STRINGIS ("T3")
```

2. String "X" är definierad som axel:

```
201 == STRINGIS ("X")
201 == STRINGIS ("X1")
```

3. String "A2" är definierad som NC-adress med tillägg:

```
201 == STRINGIS ("A")
201 == STRINGIS ("A2")
```

4. String "INVCW" är definierad som benämnt G-kommando:
202 == STRINGIS ("INVCW")
5. String "\$MC_GCODE_RESET_VALUES" är definierad som maskindatum:
206 == STRINGIS (" \$MC_GCODE_RESET_VALUES ")
6. String "GETMDACT" är en NC-språkfunktion:
203 == STRINGIS ("GETMDACT ")
7. String "DEFINE" är ett nyckelord:
205 == STRINGIS ("DEFINE")
8. String "\$TC_DP3" är en systemparameter (verktygslängdkomponent):
207 == STRINGIS (" \$TC_DP3 ")
9. String "\$TC_TP4" är en systemparameter (verktygsstorlek):
207 == STRINGIS (" \$TC_TP4 ")
10. String "\$TC_MPP4" är en systemparameter (magasinplatstillstånd):
 - Verktygsmagasin-förvaltning är aktiv: 207 == STRINGIS (" \$TC_MPP4 ") ;
 - Verktygsmagasin-förvaltning är inte aktiv: 000 == STRINGIS (" \$TC_MPP4 ")
 Se även under avsnitt: Verktygsmagasin-förvaltning.
11. String "MACHINERY_NAME" är definierad som GUD-variabel:
209 == STRINGIS ("MACHINERY_NAME")
12. String "LONGMACRO" är definierad som makro:
210 == STRINGIS ("LONGMACRO")
13. String "MYVAR" är definierad som LUD-variabel:
211 == STRINGIS ("MYVAR")
14. String "XYZ" är inget i NC känt kommando, GUD-variabel, makro- eller cykel-namn:
000 == STRINGIS ("XYZ")

Verktygsmagasin-förvaltning

Är funktionen verktygsmagasin-förvaltning inte aktiv, levererar STRINGIS för systemparametrarna i verktygsmagasin-förvaltningen, oberoende av maskindatum

- MD10711 \$MN_NC_LANGUAGE_CONFIGURATION

alltid värdet 000.

ISO Modus

Är funktionen "ISO Modus" aktiv:

- MD18800 \$MN_MM_EXTERN_LANGUAGE (aktivering externa NC-språk)
- MD10880 \$MN_MM_EXTERN_CNC_SYSTEM (styrningssystem som ska adapteras)

kontrollerar STRINGIS den angivna stringen först som SINUMERIK G-kommando. Är stringen inget SINUMERIK G-kommando kontrolleras den sedan som ISO G-kommando.

Programmerade omkopplingar (G290 (SINUMERIK Mode), G291 (ISO Mode)) har ingen inverkan på STRINGIS.

Exempel

De för funktionen STRINGIS(...) relevanta maskindata har följande värden:

- MD10711 \$MN_NC_LANGUAGE_CONFIGURATION = 2 (Endast de NC-språkkommandon ses som kända vars optioner är inställda)
- MD19410 \$ON_TRAFO_TYPE_MASK = 'H0' (Option: Transformationer)
- MD10700 \$MN_PREPROCESSING_LEVEL='H43' (förbearbetning för cykler aktiv)

Det följande exempelprogrammet genomarbetas utan felmeddelande:

Programkod	Kommentar
N1 R1=STRINGIS("TRACYL")	R1 == 0, da TRACYL på grund av den saknade transformations-optionen som "ej känd" identifierad
N2 IF STRINGIS("TRACYL") == 204	
N3 TRACYL(1,2,3)	; N3 hoppas över
N4 ELSE	
N5 G00	; och i stället utförs N5
N6 ENDIF	
N7 M30	

18.7 Anropa fönster från detaljprogrammet interaktivt (MMC)

Via det fördefinierade underprogrammet MMC(...) kan utifrån ett NC-program, användarspecifika dialoger visas på användargränssnittet till t. ex. SINUMERIK Operate.

Projekteringen av dialogerna kan göras för följande dialogtyper:

- Run MyScreens
- Easy XML
- Användar-XML

Litteratur

- Programmeringshandbok Run MyScreens
- Programmeringshandbok Easy XML

Syntax

```
MMC ("<ADDRESS>, <COMMAND>, <FILE>, <DIALOG>", "<QUIT>")
```

Betydelse

MMC (. . .) :	Underprogrambeteckning Parametrarna blir platskodade och angivna åtskilda med komma inom två strings, kommando-stringen och kvitterings-stringen.	
Parameter inom kommando-stringen:		
<ADDRESS>:	Manöverområde, i vilket de projekterade användardialogerna utförs	
	Funktion	Manöverområden
	Användardialog "Run MyScreens"	CYCLES
	Användardialog "Easy XML"	CYCLES
	Användar-XML	XML
	Popup "Run MyScreens"	POPUPDLG
	Popup "Easy XML"	POPUPDLG
<COMMAND>:	Kommando att utföra	
	Funktion	Kommandon
	Användardialog "Run MyScreens"	PICTURE_ON, PICTURE_OFF
	Användardialog "Easy XML"	PICTURE_ON, PICTURE_OFF
	Användar-XML	XML_ON, XML_OFF
	Popup "Run MyScreens"	PICTURE_ON, PICTURE_OFF
	Popup "Easy XML"	PICTURE_ON, PICTURE_OFF

18.7 Anropa fönster från detaljprogrammet interaktivt (MMC)

<FILE>:	Namn på den fil, i vilken den dialog som ska visas, är programmerad	
	Funktion	Filer
	Användardialog "Run MyScreens"	<name>.com
	Användardialog "Easy XML"	<name>.xml
	Användar-XML	<name>.xml
	Popup "Run MyScreens"	<name>.com
	Popup "Easy XML"	<name>.xml
	Popup "Easy XML" med projektering direkt i NC-programmet (se exempel 2)	xmldial_emb.xml
<DIALOG>:	Namn på den dialog som ska visas	
	Funktion	Dialognamn
	Alla funktioner utom Popup "Easy XML" med projektering direkt i NC-programmet	I filen <FILE> projekterat namn för dialogen
	Popup "Easy XML" med projektering direkt i NC-programmet (se exempel 3)	main
Parameter inom kvitterings-stringen:		
<QUIT>:	Kvitteringstyp	
	N:	Ingen kvittering. Programbearbetningen fortsätts efter sändning av kommandot. Det kommer inget svar, när kommandot inte kan utföras med framgång. Observera Kvitteringstyp "N" ska användas, när en visningstid (fördörjningstid) programmeras i NC-programmet (se nedan exempel 2)
	A:	Asynkron kvittering Programgenomarbetningen fortsätts efter annullering av kommandot. Returvärdet lagras i en inom ramen av dialogprojekteringen definierad användarspecifik kvitteringsvariabel (GUD-variabel) och kan läsas i NC-programmet.

Exempel

Exempel 1

Visning av en dialog och reaktion på användarmanövrering i ett NC-program.

Programkod	Kommentar
; Kvitteringsvariabeln QUIT anslades redan som global användarvariabel (GUD)	
; av typ STRING vid projekteringen av dialogen:	
; DEF NCK STRING[20] QUIT	
QUIT = "XXX"	; Initiera kvitteringsvariabel
G4 F5	
MMC("CYCLES, PICTU-	; Visa dialog
RE_ON, test.com, test1", "A")	; - Manöverområde: CYCLES
	; - Bildstatus: PICTURE_ON (visa)
	; - Dialogbild-fil: test.com
	; - Dialogbild: test1

Programkod	Kommentar
INPUT:	; Vänta på användarmanövrering
STOPRE	; Fördekoderingsstopp
IF MATCH (QUIT,"RUN") >= 0 GOTOF WORK	; Softkey "RUN"
IF MATCH (QUIT,"CHK") >= 0 GOTOF CHECK	; Softkey "CHK"
GOTOB INPUT	; => Vänta
WORK:	; Softkey "RUN" nedtryckt
MSG("Fortsätt med bearbetningen -> NC-Start")	; Mata ut meddelande
MMC("CYCLES,PICTURE_OFF","N")	; Stänga dialog
M0	; Vänta på NC-Start
GOTOF END	; => Programslut
CHECK:	; Softkey "CHK" nedtryckt
MSG("Uppsöka position -> NC-Start")	; Mata ut meddelande
MMC("CYCLES,PICTURE_OFF","N")	; Stänga dialog
M0	; Vänta på NC-Start
GOTOF END	; => Programslut
END:	
...	

Exempel 2

Visningstiden för en dialog fastläggs i NC-programmet t.ex. via en fördröjningstid.

Programkod	Kommentar
F1000 G94	
...	
MMC("POPUPDLG,PICTURE_ON,xmldial_emb.xml,main","N")	; Visa dialog
X200	
Z40	
MMC("POPUPDLG,PICTURE_OFF","N")	; Stänga dialog

Exempel 3

Inbäddning av en PopUp-Script i ett NC-program och dess användning.

Programkod

```

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">

```


Programkod

```

;      <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>
;          <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>

```

Programkod

```
; </main_dialog>
; ===== Programdel =====
...
G94 F100
MMC("POPUPDLG,PICTURE_ON,xmldial_emb.xml,main","N")
G4 F4
X200
MMC("POPUPDLG,PICTURE_OFF","N")
G4 F2
X0
...
```

Randvillkor

- Definitionsfilerna *.com i dialogerna måste läggas i mappen "proj".
- Easy XML definitionsfilerna *.xml i dialogerna måste läggas i mappen "appl".
Läggs definitionsfilerna i en annan katalog, måste sökuppgiften göras indirekt, utgående från katalogen "appl".
- Användardefinierade dialoger kan inte visas samtidigt från flera kanaler.

18.8 Programkörningstid / arbetsstycksräknare

För att stödja operatören av verktygsmaskinen ställs informationer över programkörningstid och antal arbetsstycken i ordning.

Dessa informationer kan bearbetas som systemvariabler i NC- och/eller PLC-program. Samtidigt står dessa informationer till förfogande för indikering på användargränssnittet.

18.8.1 Programkörningstid

Funktionen "Programkörningstid" ställer NC-intern timer för övervakning av teknologiska processer till förfogande, som kan läsas via NC- och kanal-specifika systemvariabler i detaljprogrammet och i synkronaktioner.

Trigger för mätning av körningstid (\$AC_PROG_NET_TIME_TRIGGER) är den enda skrivbara variabeln i funktionen och tjänar till selektiv mätning av programavsnitt. Dvs. genom skrivning av triggern i NC-programmet kan tidsmätningen kopplas till och åter från.

Systemvariabel	Betydelse	Aktivitet
NC-specifik		
\$AN_SETUP_TIME	Tid sedan den sista styrningsstarten med standardvärdet ("kallstart") i minuter Återställs vid varje styrningsstart med standardvärdet automatiskt på "0".	• alltid aktiv
\$AN_POWERON_TIME	Tid sedan den sista normalstarten av styrningen ("varmstart") i minuter Återställs vid varje normalstart av styrningen automatiskt på "0".	
Kanal-specifik		
\$AC_OPERATING_TIME	Total körtid för NC-programmen i driftsläget Automatik i sekunder Värdet återställs vid varje styrningstart automatiskt på "0".	• aktivering via MD27860 • endast driftsläget AUTOMATIK
\$AC_CYCLE_TIME	Körtid för det valda NC-programmet i sekunder Värdet återställs med starten av ett nytt NC-program automatiskt på "0".	
\$AC_CUTTING_TIME	Bearbetningstid i sekunder Körtiden för banaxlarna (minst en är aktiv) utan aktiv snabbgång i alla NC-program mellan NC-start och programslut/NC-Reset mäts. Mätningen avbryts dessutom vid aktiv fördröjningstid. Värdet återställs vid varje styrningsstart med standardvärdet automatiskt på "0".	

Systemvariabel	Betydelse	Aktivitet	
\$AC_ACT_PROG_NET_TIME	Aktuell netto-körningstid för det aktuella NC-programmet i sekunder Återställs med starten av ett nytt NC-program automatiskt på "0".	- alltid aktiv - endast driftsläget AUTOMATIK	
\$AC_OLD_PROG_NET_TIME	Netto-körningstid för det just korrekt med M30 avslutade programmet i sekunder		
\$AC_OLD_PROG_NET_TIME_COUNT	Ändringar på \$AC_OLD_PROG_NET_TIME Efter POWER ON står \$AC_OLD_PROG_NET_TIME_COUNT på "0". \$AC_OLD_PROG_NET_TIME_COUNT höjs alltid när styrningen har skrivit \$AC_OLD_PROG_NET_TIME på nytt.		
\$AC_PROG_NET_TIME_TRIGGER	Trigger för mätning av körningstid:		endast driftsläget AUTOMATIK
	0	Neutralt tillstånd Triggern är inte aktiv.	
	1	Avsluta Avslutar mätningen och kopierar värdet från \$AC_ACT_PROG_NET_TIME in \$AC_OLD_PROG_NET_TIME. \$AC_ACT_PROG_NET_TIME sätts på "0" och går därefter vidare.	
	2	Start Startar mätningen och sätter därvid \$AC_ACT_PROG_NET_TIME på "0". \$AC_OLD_PROG_NET_TIME förändras inte	
	3	Stopp Stoppar mätningen. Förändrar inte \$AC_OLD_PROG_NET_TIME och håller \$AC_ACT_PROG_NET_TIME konstant tills fortsättningen.	
	4	Fortsätta Fortsättning av mätningen dvs. en tidigare stoppad mätning återupptas. \$AC_ACT_PROG_NET_TIME går vidare. \$AC_OLD_PROG_NET_TIME förändras inte.	
Med POWER ON återställs alla systemvariabler på "0"!			

Märk**Maskintillverkare**

Tillkopplingen av den aktiverbara timern görs via maskindatum MD27860 \$MC_PROCESSTIMER_MODE.

Beteendet för den aktiva tidsmätningen vid vissa funktioner (t.ex. GOTOS, Override = 0%, aktiv provkörningsmatning, programtest, ASUP, PROG_EVENT, ...) konfigureras via maskindata MD27850 \$MC_PROG_NET_TIMER_MODE och MD27860 \$MC_PROCESSTIMER_MODE.

Litteratur:

Funktionshandbok Grundfunktioner; BAG, kanal, programdrift, Reset-beteende (K1), Kapitel: Programkörningstid

Märk**Resttid för ett arbetsstycke**

När lika arbetsstycken produceras efter varandra, kan den förblivande resttiden för ett arbetsstycke fastställas med följande timervärden:

- bearbetningstid för det sist producerade arbetsstycket (se \$AC_OLD_PROG_NET_TIME)
- Aktuell bearbetningstid (se \$AC_ACT_PROG_NET_TIME)

Resttiden visas förutom den aktuella bearbetningstiden på användargränssnittet.

Märk**Användning av STOPRE**

Systemvariablerna \$AC_OLD_PROG_NET_TIME och \$AC_OLD_PROG_NET_TIME_COUNT skapar inga implicita fördekoderingsstopp. Vid användning i detaljprogrammet är det okritiskt om värdet för systemvariablerna stammar från den föregående programkörningen. Men när triggern för mätning av körningstiden (\$AC_PROG_NET_TIME_TRIGGER) skrivs högfrekvent och \$AC_OLD_PROG_NET_TIME sig därigenom ofta ändrar, då borde en explicit `STOPRE` användas i detaljprogrammet.

Randvillkor

- **Blocksökning**
Vid blocksökning registreras inga programkörningstider.
- **REPOS**
Tiden för ett REPOS-förlopp räknas till den aktuella bearbetningstiden (\$AC_ACT_PROG_NET_TIME).

Exempel**Exempel 1: Mäta tid för "mySubProgrammA"**

Programkod
...

Programkod

```

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

```

Efter det programmet har genomarbetat raden N80 står i \$AC_OLD_PROG_NET_TIME nettokörningstiden för "mySubProgrammA".

Värdet för \$AC_OLD_PROG_NET_TIME:

- bibehålls utöver M30.
- aktualiseras efter varje körning av slinga.

Exempel 2: Mäta tid för "mySubProgrammA" och "mySubProgrammC"**Programkod**

```

...
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

```

18.8.2 Arbetsstycksräknare

Funktionen "Arbetsstycksräknare" ställer diverse räknare till förfogande, som speciellt kan användas för styrningsintern räkning av arbetsstycken.

Räknarna existerar som kanalspecifika systemvariabler med skriv- och läsåtkomst i värdeområdet från 0 till 999 999 999.

Systemvariabel	Betydelse
\$AC_REQUIRED_PARTS	Antal arbetsstycke som ska tillverkas (bör-arbetsstycken) I denna räknare kan det antal arbetsstycken definieras, som när det uppnåtts leder till att är-arbetsstyckstalet \$AC_ACTUAL_PARTS återställs på "0".
\$AC_TOTAL_PARTS	Antalet totalt tillverkade arbetsstycken (är-arbetsstyckstal totalt) Denna räknare anger antalet för alla från starttidpunkten tillverkade arbetsstycken. Värdet återställs endast vid en styrningsstart med standardvärden automatiskt på "0".

Systemvariabel	Betydelse
\$AC_ACTUAL_PARTS	Antalet tillverkade arbetsstycken (är-arbetsstyckstal) I denna räknare registreras antalet för alla från starttidpunkten tillverkade arbetsstycken. När bör-arbetsstyckstalet (\$AC_REQUIRED_PARTS) uppnås återställs räknaren automatiskt på "0" (förutsatt \$AC_REQUIRED_PARTS > 0).
\$AC_SPECIAL_PARTS	Antalet av användaren räknade arbetsstycken Denna räknare tillåter användaren en arbetsstycksräkning enligt egen definition. Ett utmatande av larm kan definieras när bör-arbetsstyckstalet (\$AC_REQUIRED_PARTS) har uppnåtts. En nollställning av räknaren måste användaren göra själv.

Märk

Alla arbetsstycksräknare ställs vid en styrningsstart med standardvärden på "0" och kan läsas och skrivas oberoende av deras aktivering.

Märk

Via kanalspecifika maskindata kan inflytande utövas på räknaraktiveringen, tidpunkten för nollställningen och räknealgoritmen.

Märk**Arbetsstycksräkning med användardefinierat M-kommando**

Via maskindata kan ställas in, att räkneimpulserna för de olika arbetsstycksräknarna utlöses via användardefinierade M-kommandon i stället för via programslut M2/M30.

18.9 Process DataShare - utgåva på en extern apparat/fil (EXTOPEN, WRITE, EXTCLOSE)

Skrivandet av data från ett detaljprogram på en extern apparat/fil sker i tre steg:

1. Öppna extern apparat/fil
Med kommandot EXTOPEN öppnas den externa apparaten/filen för kanalen för skrivning.
2. Skriva data
Utmatningsdatum kan förberedas med stringfunktionerna till NC-språket, t.ex. SPRINT.
Själva skrivandet sker via WRITE-kommandot.
3. Stänga extern apparat/fil
Med kommandot EXTCLOSE eller genom att programslut uppnåtts (M30) samt vid kanal-reset frigges åter den i kanalen belagda externa apparaten/filen.

Syntax

```
DEF INT <Result>
DEF STRING[<n>] <Output>
...
EXTOPEN(<Result>,<ExtDev>,<SyncMode>,<AccessMode>,<WriteMode>)
...
<Output>="Ausgabe Daten"
WRITE(<Result>,<ExtDev>,<Output>)
...
EXTCLOSE(<Result>,<ExtDev>)
```


Betydelse

EXTOPEN:	Fördefinierad procedur för att öppna en extern apparat/fil	
<Result>:	Parameter 1: Resultatvariabel	
	Med hjälp av händelsevariabelvärdet kan i programmet utvärderas hur operationen lyckas och i enlighet därmed fortsätts.	
	Typ:	INT
	Värden:	0 Inget fel
		1 Extern apparat kan inte öppnas
		2 Extern apparat är inte projekterad
		3 Extern apparat projekterad med ogiltig sökväg
		4 Inga åtkomsträtter för extern apparat
		5 Användningsmodus: Extern apparat redan "exklusivt" belagd
		6 Användningsmodus: Extern apparat redan "shared" belagd
		7 Fillängd större än LOCAL_DRIVE_MAX_FILESIZE
		8 Maximalt antal externa apparater överskridet
		9 Option för LOCAL_DRIVE inte inställd
		11 V.24-gränssnitt redan belagt av Easy-Message-funktion (endast 828D)
		12 Skrivmodus: Uppgift motsägande till extdev.ini
		16 Ogiltig extern sökväg programmerad
		22 Extern apparat inte mounted

<ExtDev>:	Parameter 2: Symbolisk beteckning för den externa apparat/fil som ska öppnas	
	Typ:	STRING
	Den symboliska beteckningen består av:	
	1. det logiska apparatnamnet	
	2. ev. följt av en sökväg för fil (tillagd med "/").	
	Följande logiska apparatnamn är definierade:	
	"LOCAL_DRIVE":	Lokalt CompactFlash Card (fördefinierat)
	"CYC_DRIVE":	Reserverad enhetsuppgift för användning i SIEMENS-cykler (fördefinierad)
	"/dev/ext/1", ... "/dev/ext/9":	Disponibla nätverksenheter Observera: Projektering i filen extdev.ini nödvändig!
	"/dev/cyc/1", "/dev/cyc/2":	Reserverade enhetsuppgifter för användning i SIEMENS-cykler Observera: Projektering i filen extdev.ini nödvändig!
	"/dev/v24":	V.24-gränssnitt Observera: Projektering i filen extdev.ini nödvändig!
	Filsökningsväg:	
	- Till "LOCAL_DRIVE" och "CYC_DRIVE" måste en filsökningsväg anges t.ex.: "LOCAL_DRIVE/my_dir/my_file.txt" - De logiska apparatnamnen "/dev/ext/1...9" och "/dev/cyc/1...2" kan per projektering: - redan hänvisa till en fil, då får endast de logiska apparatnamnen anges t.ex.: "/dev/ext/4" - eller till en pärm, då måste en filsökningsväg anges t.ex.: "/dev/ext/5/my_dir/my_file.txt" - Till "/dev/v24" får ingen filsökningsväg vara tillagd.	
	Observera:	
	För de logiska apparatnamnen "/dev/ext/1...9", "/dev/v24" och "/dev/cyc/1...2" ignoreras om stora/små bokstäver skrivs, vid uppgift av sökväg för en fil är signifikant om stora/små bokstäver skrivs. För "LOCAL_DRIVE" och "CYC_DRIVE" är endast stora bokstäver tillåtna.	

18.9 Process DataShare - utgåva på en extern apparat/fil (EXTOPEN, WRITE, EXTCLOSE)

<SyncMode>:	Parameter 3: Bearbetningsmodus för WRITE-kommandona till denna apparat/fil				
	Typ:	STRING			
	Värden:	<table><tr><td>"SYN":</td><td>Synkron skrivning Utförandet av programmet stoppas till skrivningsförloppet har avslutats. Om den synkrona skrivningen avslutats med framgång kan endast kontrolleras genom utvärdering av felvariabeln till WRITE-kommandot.</td></tr><tr><td>"ASYN":</td><td>Asynkron skrivning Utförandet av programmet avbryts inte av WRITE-kommandot. Observera: Händelsevariabeln till WRITE-kommandot säger inte mycket i detta modus och har alltid värdet 0 (inget fel). Det finns i detta modus ingen säkerhet att WRITE-kommandot var framgångsrikt.</td></tr></table>	"SYN":	Synkron skrivning Utförandet av programmet stoppas till skrivningsförloppet har avslutats. Om den synkrona skrivningen avslutats med framgång kan endast kontrolleras genom utvärdering av felvariabeln till WRITE-kommandot.	"ASYN":
"SYN":	Synkron skrivning Utförandet av programmet stoppas till skrivningsförloppet har avslutats. Om den synkrona skrivningen avslutats med framgång kan endast kontrolleras genom utvärdering av felvariabeln till WRITE-kommandot.				
"ASYN":	Asynkron skrivning Utförandet av programmet avbryts inte av WRITE-kommandot. Observera: Händelsevariabeln till WRITE-kommandot säger inte mycket i detta modus och har alltid värdet 0 (inget fel). Det finns i detta modus ingen säkerhet att WRITE-kommandot var framgångsrikt.				
<AccessMode>:	Parameter 4: Användningsmodus för denna apparat/fil				
	Typ:	STRING			
	Värden:	<table><tr><td>"SHARED":</td><td>Apparat/fil begärs i "Shared"-modus. Andra kanaler kan också använda apparaten dvs. likaså öppna i detta modus.</td></tr><tr><td>"EXCL":</td><td>Apparat/fil används exklusivt i kanalen, ingen annan kanal kan också använda apparaten.</td></tr></table>	"SHARED":	Apparat/fil begärs i "Shared"-modus. Andra kanaler kan också använda apparaten dvs. likaså öppna i detta modus.	"EXCL":
"SHARED":	Apparat/fil begärs i "Shared"-modus. Andra kanaler kan också använda apparaten dvs. likaså öppna i detta modus.				
"EXCL":	Apparat/fil används exklusivt i kanalen, ingen annan kanal kan också använda apparaten.				
<WriteMode>:	Parameter 5: Skrivmodus för WRITE-kommandona till denna apparat/fil (option)				
	Typ:	STRING			
	Värden:	<table><tr><td>"APP":</td><td>Tillägg Filen bibehålls med sitt innehåll, skrivanrop läggs till vid slutet.</td></tr><tr><td>"OVR":</td><td>Överskrivning Innehållet i filen raderas och upprättas på nytt genom efterföljande skrivanrop.</td></tr></table>	"APP":	Tillägg Filen bibehålls med sitt innehåll, skrivanrop läggs till vid slutet.	"OVR":
"APP":	Tillägg Filen bibehålls med sitt innehåll, skrivanrop läggs till vid slutet.				
"OVR":	Överskrivning Innehållet i filen raderas och upprättas på nytt genom efterföljande skrivanrop.				
Observera: Med denna parameter kan det i filen extdev.ini projekterade skrivmoduset inte skrivas över. I händelse av konflikt kvitteras EXTOPEN-anropet med fel.					

WRITE:	Fördefinierad procedur för att skriva utmatningsdata
--------	--

EXTCLOSE:	Fördefinierad procedur för att stänga en öppnad extern apparat/fil		
<Result>:	Parameter 1: Resultatvariabel		
	Typ:	INT	
	Värden:	0	Inget fel
		16	Ogiltig extern sökväg programmerad
		21	Fel vid stängning av den externa apparaten
<ExtDev>:	Parameter 2: Symbolisk beteckning för den externa apparat/fil som ska stängas, beskrivning se under EXTOPEN! Observera: Beteckningen måste vara identisk till den i EXTOPEN-anropet angivna beteckningen!		

Exempel

Programkod

```

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("Fehler bei EXTOPEN:" << RESULT)
N80      ELSE
N90          EXTDEVICE=TRUE
N100     ENDIF
...
N200     PHASE=4
N210     IF EXTDEVICE
N220         OUTPUT=SPRINT("Ende Phase: %D",PHASE)
N230         WRITE(RESULT,"LOCAL_DRIVE/my_file.txt",OUTPUT)
N240     ENDIF
...

```

Se även

Stringoperationer (Sida 83)

Skriva fil (WRITE) (Sida 143)

18.10 Larm (SETAL)

I ett NC-program kan larm ställas in. Dessa visas i användargränssnitt i ett speciellt fält. Met ett larm är alltid en reaktion hos styrningen förbunden i enlighet med larmkategori.

Litteratur:

Mer detaljerade informationer över larmreaktionerna se handboken för Idrifttagande.

Syntax

```
SETAL (<Alarmnummer>[, <Zeichenkette>])
```

Betydelse

SETAL:	Nyckelord för programmeringen av ett larm. SETAL måste programmeras i ett eget NC-block.	
<Alarmnummer>:	Variabel av typ INT. Innehåller larmnumret. Det giltiga området för larmnumren ligger mellan 60000 och 69999, varav 60000 till 64999 är reserverade för SIEMENS-cykler och 65000 till 69999 står till förfogande för användaren.	
<Zeichenkette>:	Vid programmeringen av användarcykellarmen kan dessutom en teckenföljd med upp till 4 parametrar angivas. I dessa parametrar kan variabla användartexter definieras. Men även följande parametrar står till förfogande:	
	Parametrar	Betydelse
	%1	Kanalnummer
	%2	Blocknummer, label
	%3	Textindex för cykellarm
	%4	Extra larmparametrar

Märk

Larmtexter måste projekteras i användargränssnittet.

Märk

Ska en utmatning av larm göras i det på användargränssnittet aktiva språket, behöver användaren informationer över det aktuellt på HMI inställda språket. Denna information kan avfrågas i detaljprogrammet och i synkronaktionerna via systemvariabeln \$AN_LANGUAGE_ON_HMI (se "Aktuellt språk i HMI (Sida 908)").

Exempel

Programkod	Kommentar
...	
N100 SETAL(65000)	; Sätta larm nr 65000
...	

18.11 Utvidgat stoppande och återdragande (ESR)

Funktionen "Utvidgat stoppande och återdragande", i fortsättningen betecknad med ESR erbjuder möjligheten att reagera processberoende flexibelt i felsituationer:

- **Utvidgat stoppande**
I den mån som den specifika felsituationen tillåter stoppas alla för det utvidgade stoppandet frigivna axlarna ordnat.
- **Återdragning**
Det verktyg som befinner sig i ingrepp dras så snabbt som möjligt tillbaka från arbetsstycket.
- **Generatordrift (SINAMICS-servofunktion "Vdc-reglering")**
Vid underskridandet av ett parameterbart värde för mellankretsspänningen t. ex. vid avbrott i nätspänningen, genereras den för återdragningen behövliga elektriska energin genom återgångsmatning av bromsenergin från en för detta avsedd servo (generatordrift).

Utlösningsskällor

Allmänna källor (NC-extern/global eller BAG-/kanalspecifik)

- Digitala ingångar (t. ex. på NCU-komponent) resp. den styrningsinterna, återläsbara avbilden av digitala utgångar (\$A_IN, \$A_OUT)
- Kanaltillstånd (\$AC_STAT)
- VDI-signaler (\$A_DBB)
- Samlingsmeddelanden för ett antal larm (\$AC_ALARM_STAT)

Axiala källor

- Nödåtergångströskel för följdaxeln (synkrongång för den elektroniska kopplingen, \$VC_EG_SYNCDIFF[<Folgeachse>])
- Servo: Mellankret-varningströskel (hotande underspänning), \$AA_ESR_STAT[<Achse>]
- Servo: Generator-minimalvarvtal-tröskel (ingen rotationsenergie finns längre som kan återgångsmatas), \$AA_ESR_STAT[<Achse>].

Sammanbindningslogik för de statisk synkronaktionerna: Källor-/reaktions-sammanbindning

De flexibla sammanbindningsmöjligheterna av de statiska synkronaktionerna används för att på grund av källor utlösa bestämda reaktioner relativt snabbt.

Sammanbindningen av alla relevanta källor med hjälp av statiska synkronaktioner ligger i användarens händer. Han kan också utvärdera käll-systemvariablerna som helhet eller med hjälp av bit-masker också selektivt och knyta sina önskade reaktioner härtill. De statiska synkronaktionerna är verksamma i alla driftslägen.

Litteratur:

Funktionshandbok Synkronaktioner

Aktivering

Funktionsfrigivning

Funktionerna generatordrift, stoppande, återdragning frigges genom att sätta den tillhörande styrsignalen \$AA_ESR_ENABLE. Denna styrsignal kan förändras av synkronaktioner.

Funktionsutlösning

ESR utlöses gemensamt för alla frigivna axlar genom att sätta systemvariabeln \$AC_ESR_TRIGGER.

Generatordriften blir "automatiskt" aktiv i servon vid identifikation av hotande mellankrets-underspänning.

Drivningsautarktt stoppande och/eller återdragande blir aktiva vid identifikation av ett kommunikationsbortfall (mellan NC och servo) samt vid identifikation av en mellankrets-underspänning i servon (konfiguration och frigivning förutsatta).

Drivningsautarktt stoppande och/eller återdragande kan dessutom också utlösas från NC-sidan genom att sätta motsvarande styrsignal \$AN_ESR_TRIGGER (broadcast-kommando till alla drivningar).

Litteratur

Utförliga informationer till ESR se:

Funktionshandbok Specialfunktioner; Utvidgat stoppande och återdragande (R3)

18.11.1 NC-styrd ESR

18.11.1.1 NC-styrd återdragning (POLF, POLFA, POLFMASK, POLFMLIN)

För NC-styrd återdragning är vissa utgångsvillkor nödvändiga (se "NC-styrd återdragning (POLF, POLFA, POLFMASK, POLFMLIN) (Sida 659)"). Är dessa villkor uppfyllda då aktiveras för den(de) i kanalen konfigurerade återgångsaxeln(larna) snabblyftning (LIFTFAST) genom att sätta systemvariabeln \$AC_ESR_TRIGGER (resp. \$AA_ESR_TRIGGER för singelaxlar).

Syntax

```
POLF(<Achse>)=<Position>
POLFA(<Achse>,<Typ>,<Position>)
POLFMASK(<Achse_1>,<Achse_2>,...)
POLFMLIN(<Achse_1>,<Achse_2>,...)
```

För POLFA är följande förkortade former tillåtna:

```
POLFA(<Achse>,<Typ>) ; Förkortad form för singelaxel-återdragning
POLFA(Achse,0/1/2) ; Snabb inaktivering eller aktivering
POLFA(Achse,0,$AA_POLFA[Achse]) ; Förorsakar ett fördekoderingsstopp
POLFA(Achse,0) ; Förorsakar inget fördekoderingsstopp
```

Betydelse

POLF:	Adressför angivande av målpositionen för återgångsaxeln POLF är modalt verksam.	
	<Achse>:	Namn för geometri- eller kanal-/maskinaxel som drar tillbaka
	<Position>:	Återgångsposition
		Typ: REAL För geometriaxel gäller WKS, annars MKS. Vid samma beteckningar för geometri- och kanal-/maskinaxel dras tillbaka i WKS..
POLFA:	Fördefinierat underprogramanrop för angivande av återgångsposition för singelaxlar	
	<Achse>:	Kanalaxelbeteckning
	<Typ>:	Positionsangivningsmode
		Typ: INT
		Värde: 0: Markera positionsvärde som ogiltigt
		1: Positionsvärde är absolut
		2: Positionsvärde är inkrementellt (distans)
	Observera: Är en axel ingen singelaxel eller saknas typen resp. typ=0, då meddelas ett motsvarande larm.	
POLFMASK:	<Position>:	Återgångsposition (s. o.)
	Observera: Positionsvärdet övertas också med typ=0. Men då är detta värde markerat som ogiltigt och måste programmeras på nytt för återdragningen.	
	Fördefinierat underprogramanrop för val av de axlar som efter utlösningen av snabblyftningen ska dras tillbaka oberoende av varandra .	
	<Achse_1>, ...:	Namnen på de axlar som vid snabblyftning ska gå till sina med POLF definierade positioner. Alla angivna axlar måste finnas i samma korrdinatsystem.
	POLFMASK() utan angivande av en axel inaktiverar snabblyftningen för alla axlar som drogs tillbaka oberoende av varandra.	
POLFMLIN:	Fördefinierat underprogramanrop för val av de axlar som efter utlösningen av snabblyftningen ska dras tillbaka i linjärt sammanhang .	
	<Achse_1>, ...:	s. o.
	POLFMLIN() utan angivande av en axel inaktiverar snabblyftningen för alla axlar som drogs tillbaka i linjärt sammanhang.	

Märk

Innan via POLFMASK eller POLFMLIN snabblyftningen till en fast position kan frigivas, måste för de valda axlarna en position ha programmerats med POLF.

Märk

Friges axlar efter varandra med POLFMASK, POLFMLIN eller POLFMLIN, POLFMASK gäller för respektive axel alltid det sista fastläggande.

Märk

De med `POLF` programmerade positionerna och aktiveringen genom `POLFMASK` eller `POLFMLIN` raderas vid detaljprogramstart. Det vill säga användaren måste i varje detaljprogram på nytt programmera värdena för `POLF` och de selekterade axlarna i `POLFMASK` resp. `POLFMLIN`.

Märk

Om vid användning av den förkortade formen `POLFA` endast typen ändras, då måste användaren säkerställa att antingen återgångspositionen eller återgångsvägen innehåller ett förnuftigt värde. Speciellt ska återgångsposition och återgångsväg ställas in på nytt efter `POWER ON`.

Exempel

Återdragning av en singelaxel:

Programkod	Kommentar
MD37500 \$MA_ESR_REACTION[AX1]=21	; NC-styrd återdragning.
...	
\$AA_ESR_ENABLE[AX1]=1	
POLFA(AX1,1,20.0)	; AX1 tilldelas den axiala återgångspositionen 20.0 (absolut).
\$AA_ESR_TRIGGER[AX1]=1	; Från och med här börjar återgången.

Ytterligare informationer**Förutsättningar för NC-styrd återdragning**

- I kanalen är en återgångsaxel för den NC-styrda återdragningen projekterad:
MD37500 \$MA_ESR_REACTION = 21
- ESR måste vara frigiven för denna axel:
\$AA_ESR_ENABLE = 1
- Fördröjningstider är definierade:
MD21380 \$MC_ESR_DELAY_TIME1
MD21381 \$MC_ESR_DELAY_TIME2
- I detaljprogrammet är de axelspecifika återgångspositionerna programmerade med `POLF`.
- Axlarna är valda med `POLFMASK`/`POLFMLIN` för den NC-styrda återdragningen.
- För återgångsrörelsen måste frigivningssignalerna vara satta och förbli satta.

Frige och starta NC-styrd återdragning

När systemvariabeln `$AC_ESR_TRIGGER` = 1 är satt och när i denna kanal en återgångsaxel är konfigurerad (dvs. MD37500 \$MA_ESR_REACTION = 21) och för denna axel `$AA_ESR_ENABLE` = 1 är satt, då aktiveras i denna kanal snabblyftning (`LIFTFAST`).

18.11 Utvidgat stoppande och återdragande (ESR)

Den med `POLF` (resp. `LFPOS`) konfigurerade lyftningsrörelsen för den(de) med `POLFMASK` eller `POLFMLIN` selekterade axeln(axlarna) ersätter den för denna axel (dessa axlar) i detaljprogrammet fastlagda banrörelsen.

För återgången står maximalt summan av tiderna MD21380 `$MC_ESR_DELAY_TIME1` och MD21381 `$MC_ESR_DELAY_TIME2` till förfogande. När denna tidsfrist gått ut inleds också för återgångsaxeln snabbbromsning med anslutande medföljning.

Märk

Den utvidgade återgångsrörelsen (dvs. genom `$AC_ESR_TRIGGER` utlöst `LIFTFAST/` `LFPOS`) kan **inte avbrytas** och kan endast avslutas i förtid med nödstopp.

Märk

Genom `$AC_ESR_TRIGGER` utlöst återdragning är reglad mot flerfaldig återgång.

Singelaxel-återdragning

Vid singelaxel-återdragning måste med `POLFA` återgångspositionen för singelaxeln vara programmerad och de följande villkoren måste respekteras:

- `$AA_ESR_ENABLE = 1`
- `<Achse>` måste vid triggertidpunkten (`$AA_ESR_TRIGGER = 1`) vara en singelaxel.
- `<Typ>` måste vara antingen 1 eller 2.

Återgångsriktning vid snabblyftning

Vid tidpunkten för aktiveringen av snabblyftningen tas det hänsyn till den giltiga framen.

Märk

Frames med vridning påverkar via `POLF` också den riktning i vilken lyftningen sker.

Axelbyte

Återgångsaxlar ska alltid tillordnas exakt en NC-kanal och får inte bytas mellan kanalerna. Vid försök att byta en återgångsaxel till en annan kanal meddelas ett larm. Först efter det denna axel åter inaktiverats med `$AA_ESR_ENABLE[AX] = 0` kan axeln bytas till en annan kanal. Efter det axelbyte gjorts kan axlar åter sättas i gång med `$AA_ESR_ENABLE[AX] = 1`.

Neutrala axlar

Neutrala axlar kan inte utföra någon NC-styrd ESR.

18.11.1.2 NC-styrt stoppande

För den(de) i kanalen konfigurerade axlen(axlarna) som ska stoppas, aktiveras det NC-styrda stoppandet genom att sätta systemvariablerna `$AC_ESR_TRIGGER` (resp. `$AA_ESR_TRIGGER` för singelaxlar).

Förutsättningar

- I kanalen är en axel som ska stoppas projekterad för det NC-styrda stoppandet:
MD37500 \$MA_ESR_REACTION = 22
- ESR måste vara frigiven för denna axel:
\$AA_ESR_ENABLE = 1
- Fördröjningstider är definierade:
MD21380 \$MC_ESR_DELAY_TIME1 (fördröjningstid ESR-axlar)
MD21381 \$MC_ESR_DELAY_TIME2 (ESR-tid för interpoleringsbromsning)

Förlopp

För varaktigheten av tidsfristen i MD21380 interpolerar axeln ostörd vidare enligt programmeringen. När tidsfristen i MD21380 har gått till ända, inleds interpoleringsstyrd bromsning (rampstopp). För den interpoleringsstyrda bromsningen står då maximalt tidsfristen i MD21381 till förfogande. När denna tidsfrist har gått till ända inleds snabbbromsning med anslutande medföljning.

Exempel

Stoppande av en singelaxel:

Programkod	Kommentar
MD37500 \$MC_ESR_REACTION[AX1]=22	; NC-styrt stoppande.
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	; Från och med här börjar stoppan-
	det.

18.11.2 Drivningsautarkt ESR

18.11.2.1 Projektera drivningsautarkt stoppande (ESRS)

Med funktionen `ESRS ( . . . )` projekteras drivparametrarna för "Stoppandet" av den drivningsautarka ESR-funktionen.

Syntax

```
ESRS(<Achse_1>,<Stillsetzzeit_1>[,...,<Achse_n>,<Stillsetzzeit_n>])
```

Betydelse

ESRS (...):	Funktion för att skriva drivningsparametrarna för ESR-funktionen "Stoppande" Funktionen: • måste stå ensam i blocket. • utlöser fördekoderingsstopp. • kan inte användas i synkronaktioner.	
<Achse_1>, ..., <Achse_n>:	Axel, för vilken det drivningsautarka stoppandet ska projekteras I drivningen skrivs för denna axel drivparametrarna p0888 (konfiguration): p0888 = 1	
	Typ:	AXIS
	Värdeområde:	Kanalaxelbeteckning
<Stillsetzzeit_1>, ..., <Stillsetzzeit_n>:	Tidsrymd över vilken drivningen fortsätter konstant med det aktuella varvtalsbörvärdet efter uppträdandet av ett fel I drivningen skrivs för den angivna axeln drivparametrarna p0892 (tidsnivå): p0892 = <Stillsetzzeit>	
	Enhet:	s
	Typ:	REAL
	Värdeområde:	0.00 - 20.00
I ett funktionsanrop kan maximalt 5 axlar programmeras; n = 5		

18.11.2.2 Projektera drivningsautarkt återdragande (ESRR)

Med funktionen `ESRR(...)` projekteras drivparametrarna för "Återdragandet" av den drivningsautarka ESR-funktionen.

Syntax

```
ESRR(<Achse_1>,<Rückzugsweg_1>,<Rückzugsgeschwindigkeit_1>[,...,<Achse_n>,<Rückzugsweg_n>,<Rückzugsgeschwindigkeit_n>])
```

Betydelse

ESRR (...):	Funktion för att skriva drivparametrarna för ESR-funktionen "Återdragande" Funktionen: • måste stå ensam i blocket. • utlöser fördekoderingsstopp. • kan inte användas i synkronaktioner.	
<Achse_1>, ..., <Achse_n>:	Axel, för vilken det drivningsautarka återdragandet ska projekteras I drivningen skrivs för denna axel drivparametrarna p0888 (konfiguration): p0888 = 2	
	Typ:	AXIS
	Värdeområde:	Kanalaxelbeteckning
<Rückzugsweg_1>, ..., <Rückzugsweg_n>:	Återdragningsvägen räknas för drivningen om i ett återdragningsvarvtal. Värdet skrivs för den angivna axeln i drivningsparametern p0893 (varvtal): p0893 = (<Rückzugsweg_n> omräknat i återdragningsvarvtal)	
	Enhet:	mm/min, inch/min, grad/min (beroende på axelns enhet)
	Typ:	REAL
	Värdeområde:	MIN - MAX
<Rückzugsgeschwindigkeit_1>, ..., <Rückzugsgeschwindigkeit_n>:	Återdragningshastigheten räknas för drivningen om i en tidsrymd. Värdet skrivs för den angivna axeln i drivningsparametern p0892 (tidsnivå) [s]: p0892 = <Rückzugsweg_n> / <Rückzugsgeschwindigkeit_n>	
	Enhet:	mm/min, inch/min, grad/min (beroende på axelns enhet)
	Typ:	REAL
	Värdeområde:	0.00 - MAX
I ett funktionsanrop kan maximalt 5 axlar programmeras; n = 5		

18.12 Definiera råämne (WORKPIECE)

Styrningen måste känna till form och storlek för ett råämne, för att kunna visa det i den grafiska simuleringen. Användaren har därför möjlighet, att definiera råämnen via användargränssnittet eller direkt i NC-programmet. Råämnesdefinitionerna förblir bibehållna utöver (programslut-/kanal-/BAG-)reset. Med nästa start av styrningen raderas de automatiskt.

Syntax

```
WORKPIECE("<WP>", "<RefP>", "<ZeroOffset>", "<Type>", <Par5>,
<Par6>, ..., <Par12>)
```

Betydelse

WORKPIECE (...):		Fördefinierad procedur för definition av ett råämne	
		Fördekoderingsstopp:	ja
		Ensam i blocket:	ja
Parameter:			
1	"<WP>":	Namn på arbetsstycket (option)	
		Datatyp:	STRING
		En uppgift är nödvändig, endast när flera arbetsstycken kan finnas i en kanal. Utan uppgift antas automatiskt "WORKP<n>" varvid <n> är numret för den deklarerande kanalen.	
2	"<RefP>":	Uppspänning (option; endast på fräsmaskiner)	
		Datatyp:	STRING
		Värdeområde:	"Bord" Uppspänning på det fasta bordet
			"A" Uppspänning på roterande axel A
			"B" Uppspänning på roterande axel B
			"C" Uppspänning på roterande axel C
		Förutsättning: Bordet resp. den roterande axeln måste vara frikopplat via motsvarande maskindatum för råämnesuppspanningen (se Idrifttagningshandbok SINUMERIK Operate).	
3	"<ZeroOffset>":	Inställbar nollpunktsförflyttning för placering av råämnet (kan inte programmeras) Valet av en inställbar nollpunktsförflyttning för placering av råämnet erbjuds endast vid råämnesinmatning via användargränssnittet. Vid den direkta råämnesdefinitionen i detaljprogrammet hänför sig råämnet däremot alltid till den aktuellt giltiga nollpunktsförflyttningen.	
4	"<Type>":	Råämnesform	
		Datatyp:	STRING
		Värdeområde:	"CYLINDER": Cylinder
			"PIPE": Rör
			"RECTANGLE": Koncentriskt rätblock
			"BOX": Rätblock
		"N_CORNER": N-hörning	

5 ... 12	<Par5> ... <Par12>:	Parameter för beskrivning av råämnesformen	
		Datatyp:	REAL
		Antalet nödvändiga parametrar och deras betydelse är beroende av respektive råämnesform och av värdet för bitparametern. Se: • Tabell "Parameter för beskrivning av råämnesformen" • Tabell "Bitparameter"	
WORKPIECE () :		Ett WORKPIECE-anrop utan parameter raderar alla råämnesdefinitioner.	
WORKPIECE (<WP>) :		Ett WORKPIECE-anrop med arbetsstycksnamn raderar endast denna råämnesdefinition.	

Tabell 18-1 Parameter för beskrivning av råämnesformen

Råämnesform	Parameter							
	<Par5>	<Par6>	<Par7>	<Par8>	<Par9>	<Par10>	<Par11>	<Par12>
Cylinder	Bitparameter Realvärde, som interpreteras som bitcoderat heltal Bits definierar betydelsen av följande parametrar (se Tabell "Bitparameter").	Referenspunkt Z_0	Längd Z_1	Bearbetningsmått Z_B	Ytterdiameter d_0	-	Vridning runt den roterande axeln	-
Rör		Referenspunkt Z_0	Längd Z_1	Bearbetningsmått Z_B	Ytterdiameter d_0	Vägg-tjocklek (ink) / innerdiameter d_1 (abs)	Vridning runt den roterande axeln	-
Koncentriskt rättblock		Referenspunkt Z_0	Längd Z_1	Bearbetningsmått Z_B	Bredd W	Längd L	Vridning runt den roterande axeln	-
Rättblock		Referenspunkt Z_0	Längd Z_1	Bearbetningsmått Z_B	X_0	Y_0	X_1	Y_1
N-hörning		Referenspunkt Z_0	Längd Z_1	Bearbetningsmått Z_B	Antal hörn	Nyckelvidd	Vridning runt den roterande axeln	-

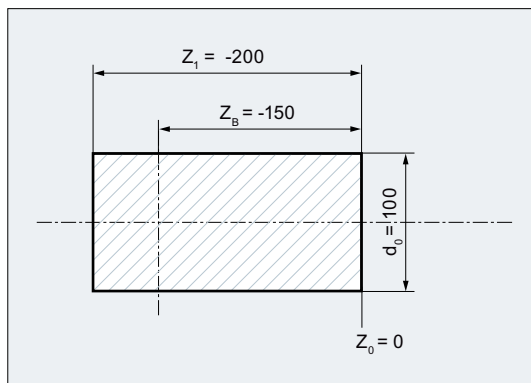
Tabell 18-2 Bitparameter

Bit	Betydelse	
4 (0x0010)	Rättblock: X_1	
	= 0	ink
	= 1	abs
5 (0x0020)	Rättblock: Y_1	
	= 0	ink
	= 1	abs

Bit	Betydelse	
6 (0x0040)	Längd Z_1 (slutmått)	
	= 0	ink
	= 1	abs
Bit 7 (0x0080)	Bearbetningsmått Z_B	
	= 0	ink
	= 1	abs
Bit 8 (0x0100)	Rör: Väggtjocklek / innerdiameter	
	= 0	ink
	= 1	abs
9 (0x0200)	N-hörning	
	= 0	Nyckelvidd
	= 1	Kantlängd
12 (0x1000)	Uppspänning för svarvmaskiner	
	= 0	Huvudspindel
	= 1	Motspindel
13 (0x2000)	Motspindel	
	= 0	med spegling
	= 1	utan spegling

Exempel

Exempel 1: Cylinderformigt råämne på en svarvmaskin



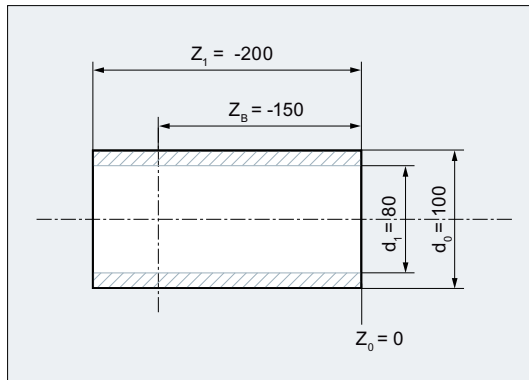
Programkod

Kommentar

...

Programkod	Kommentar
WORKPIECE(,,, "CYLINDER",0,0,-200,-150,100)	; Råämnedefinition:
...	; Råämneshänvisning:
	; Råämneshänvisning: Cylinder
	; Bitparameter=0 (ingen bit inställd) → Värden för längd och bearbetningsmått är inkrementella, råämne på huvudspindel
	; Referenspunkt (Z0)=0
	; Längd (Z1)=-200
	; Bearbetningsmått (ZB)=-150
	; Ytterdiameter (d0)=100

Exempel 2: Rörformigt råämne på en svarvmaskin



Programkod	Kommentar
...	
WORKPIECE(,,, "PIPE",256,0,-200,-150,100,80)	; Råämnedefinition:
	; Råämneshänvisning: Rör
	; Bitparameter=256 (Bit8=1) → Innerdiameter är absolut, längd och bearbetningsmått är inkrementella, råämne på huvudspindel
	; Referenspunkt (Z0)=0
	; Längd (Z1)=-200
	; Bearbetningsmått (ZB)=-150
	; Ytterdiameter (d0)=100
	; Innerdiameter (d1)=80
...	

18.13 Koppla om språkmode (G290, G291)

Styrningen erbjuder möjlighet att läsa in och genomarbeta detaljprogram från externa CNC-system. Förutsättning är att motsvarande NC-språkmode (ISO-dialekt) fastlades vid idrifttagningen.

Litteratur:

Funktionshandbok ISO-dialekter

ISO-dialektmode kan aktiveras åtskilt för varje kanal. Till exempel kan kanal 1 gå i ISO-dialektmode, under det att kanal 2 är aktiv i SINUMERIK-mode.

Omkopplingen mellan SINUMERIK-mode och ISO-dialektmode görs i NC-programmet via kommandona i G-gruppen 47. Det aktiva verktyget, verktygskompenseringarna och nollpunktsförflyttningarna påverkas inte av omkopplingen.

Syntax

```
G291
...
G290
```

Betydelse

G290:	Aktivera SINUMERIK-språkmode	
	Ensam i blocket:	ja
	Verkan:	modal
G291:	Aktivera ISO-språkmode	
	Ensam i blocket:	ja
	Verkan:	modal

Villkor

SINUMERIK-Modus

- Förinställningen av G-kommandot kan fastläggas för varje kanal via ett maskindatum.
- I SINUMERIK-mode kan inga språkkommandon programmeras från ISO-dialekterna.

ISO-dialektmode

- ISO-dialektmode som grundinställning för styrningen kan ställas in med maskindata. Styrningen startas sedan standardmässigt i ISO-dialektmode.
- Endast G-kommandon från ISO-dialekten kan programmeras. Programmeringen av SINUMERIK-funktioner är inte möjligt i dialektmode.
- En blandning av ISO-dialekt- och SINUMERIK-språk i samma NC-block är inte möjligt.
- En omkoppling via G-kommandon mellan ISO-dialekt-M (fräsa) och ISO-dialekt-T (svarva) är inte möjligt.

- Underprogram, som är programmerade i SINUMERIK-mode, kan anropas.
- Om SINUMERIK-funktioner ska användas, måste först kopplas om till SINUMERIK-mode (se exempel).

Exempel

Kompression av linjärblock i ISO-dialektmode

Programkod	Kommentar
N5 G290	; Aktivera SINUMERIK-språkmode.
N10 COMPON	; COMPON är ett kommando i Siemens-språket, som aktiverar en kompressorfunktion, som ersätter på varandra följande linjärblock med polynom-block med så stora banlängder som möjligt.
N15 G291	; Aktivera ISO-språkmode.
N20 G01 X100 Y100 F1000	; Eftersom COMPON aktiverades i SINUMERIK-mode, kan också linjärblock i ISO-dialektmode komprimeras med denna funktion.
...	

Egna avspåningsprogram

19.1 Stödjande funktioner för avspåningen

För avspåningen erbjuds färdiga bearbetningscykler. Därutöver har du möjlighet att med de i det följande uppförda funktionerna upprätta egna avspåningsprogram:

- Upprätta konturtabell (CONTPRON)
- Upprätta kodad konturtabell (CONTDCON)
- Koppla från konturförberedelse (EXECUTE)
- Fastställa skärningspunkt mellan två konturelement (INTERSEC)
(Endast för tabeller som upprättades med CONTPRON.)
- Blockvis genomarbeta konturelement i en tabell (EXECTAB)
(Endast för tabeller som upprättades med CONTPRON.)
- Beräkna cirkeldata (CALCDAT)

Märk

Du kan använda dessa funktioner inte bara till avspåning utan universiellt.

Förutsättningar

Före anrop av funktionerna CONTPRON eller CONTDCON måste:

- en startpunkt uppsökas, som tillåter en kollisionsfri bearbetning.
- skärradiekompenseringen med G40 vara fränkopplad.

19.2 Upprätta konturtabell (CONTPRON)

Med CONTPRON tillkoppling av konturförberedelse. De efterföljande anropade NC-blocken genomarbetas inte, delas upp i enskilda rörelser och läggs i konturtabellen. Varje konturelement motsvarar en tabellrad i konturtabellens tvådimensionella fält. Antalet registrerade baksnitt returneras.

Syntax

Koppla till konturförberedelse:

```
CONTPRON (<Konturtabelle>, <Bearbeitungsart>, <Hinterschnitte>,
<Bearbeitungsrichtung>)
```

Koppla från konturförberedelse och återvända till den normala genomarbetningsmoden:

```
EXECUTE (<FEHLER>)
```

Se " Koppla från konturförberedelse (EXECUTE) (Sida 689) "

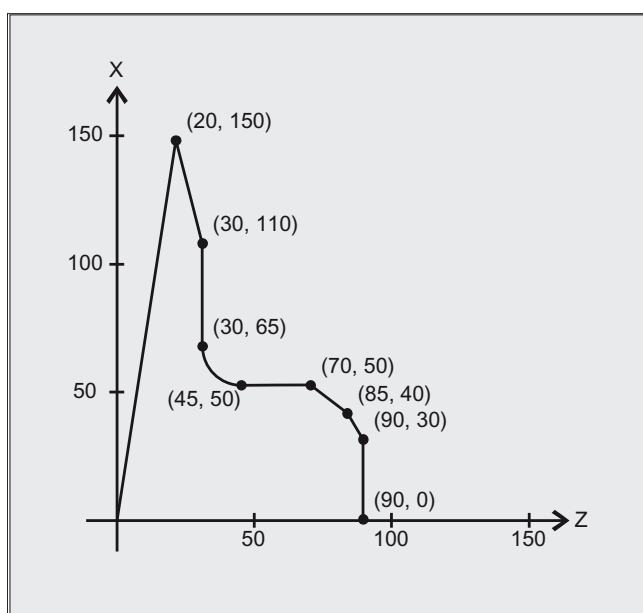
Betydelse

CONTPRON:	Fördefinierad procedur för tillkoppling av konturförberedelsen för upprättande av en konturtabell		
<Konturtabelle>:	Namn på konturtabellen		
<Bearbeitungsart>:	Parameter för bearbetningstypen		
	Typ:	CHAR	
	Värde:	"G":	Längssvarva: Innerbearbetning
		"L":	Längssvarva: Ytterbearbetning
		"N":	Plansvarva: Innerbearbetning
"P":		Plansvarva: Ytterbearbetning	
<Hinterschnitte>:	Resultatvariabel för antalet uppträdande baksnittselement		
	Typ:	INT	
<Bearbeitungsrichtung>:	Parameter för bearbetningsriktningen		
	Typ:	INT	
	Värde:	0	Konturförberedelse framåt (standardvärde)
		1	Konturförberedelse i båda riktningarna

Exempel 1

Upprätta en konturtabell med:

- namnet "KTAB"
- max. 30 konturelement (cirklar, linjer)
- en variabel för antalet uppträdande baksnittselement
- en variabel för felmeddelanden

**NC-program:**

Programkod	Kommentar
N10 DEF REAL KTAB[30,11]	; Konturtabell med namnet KTAB och max. 30 konturelement, parametervärde 11 (spalttal i tabellen) är en fast storhet.
N20 DEF INT ANZHINT	; Variabel för antalet baksnittselement med namnet ANZHINT.
N30 DEF INT FEHLER	; Variabel för felsvaret (0=inget fel, 1=fel).
N40 G18	
N50 CONTPRON(KTAB,"G",ANZHINT)	; Koppla till konturförberedelse.
N60 G1 X150 Z20	; N60 till N120: Konturbeskrivning
N70 X110 Z30	
N80 X50 RND=15	
N90 Z70	
N100 X40 Z85	
N110 X30 Z90	
N120 X0	
N130 EXECUTE(FEHLER)	; Avsluta ifyllning av konturtabellen, omkoppling till normal programdrift.
N140 ...	; Ytterligare bearbetning av tabellen.

Konturtabell KTAB:

Index Rad	Spalt									
(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

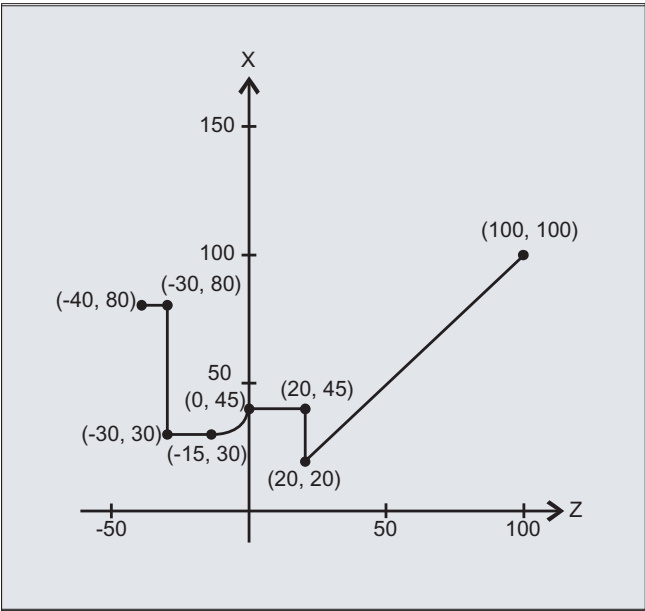
Förklaring av spaltinnehållet:

- (0) Pekare på nästa konturelement (på dess radnummer)
- (1) Pekare på föregående kontruellement
- (2) Kodning av konturmode för rörelsen
Möjliga värden för 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) Begynnelsepunkt för konturelementen
(3) = abskissa, (4) = ordinata i det aktuella planet
- (5), (6) Slutpunkt för konturelementen
(5) = abskissa, (6) = ordinata i det aktuella planet
- (7) Max-/min-pekare: kännetecknar lokala maxima och minima i konturen
- (8) Maximalt värde mellan konturelement och abskissa (vid längsbearbetning) resp. ordinata (vid planbearbetning). Vinkeln är beroende av den programmerade bearbetningstypen.
- (9), (10) Medelpunktskoordinater för konturelementet när det är ett cirkelblock.
(9) = abskissa, (10) = ordinata

Exempel 2

Upprätta en konturtabell med

- namnet KTAB
- max. 92 konturelement (cirklar, linjer)
- driftssätt: längssvarvning, ytterbearbetning
- förberedelse framåt och bakåt



NC-program:

Programkod	Kommentar
N10 DEF REAL KTAB[92,11]	; Konturtabell med namnet KTAB och max. 92 konturelement, parametervärde 11 är en fast storhet.
N20 DEF CHAR BT="L"	; Driftssätt för CONTPRON: Längssvarvning, ytterbearbetning
N30 DEF INT HE=0	; Antal baksnittselement=0
N40 DEF INT MODE=1	; Förberedelse framåt och bakåt
N50 DEF INT ERR=0	; Felsvar
...	
N100 G18 X100 Z100 F1000	
N105 CONTPRON(KTAB,BT,HE,MODE)	; Koppla till konturförberedelse.
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)	: Avsluta ifyllning av konturtabellen, omkoppling till normal programdrift.
...	

Konturtabell KTAB:

Efter avslutad konturförberedelse står konturen i båda riktningarna till förfogande.

Index	Spalt										
Rad	(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

Förklaring av spaltinnehållet och anmärkningar till raderna 0, 1, 6, 8, 83, 85 och 91

De i exempel 1 nämnda förklaringarna av spaltinnehållet gäller.

Alltid i tabellrad 0:

- 1) Föregångare: Rad n innehåller konturslutet framåt
- 2) Efterföljare: Rad n är konturtabellslut framåt

En gång var inom konturelementen framåt:

- 3) Föregångare: Konturbörjan (framåt)
- 4) Efterföljare: Konturslut (framåt)

Alltid på rad konturtabellslut (framåt) +1:

- 5) Föregångare: Antal baksnitt framåt
- 6) Efterföljare: Antal baksnitt bakåt

En gång var inom konturelementen bakåt:

- 7) Efterföljare: Konturslut (bakåt)
- 8) Föregångare: Konturbörjan (bakåt)

Alltid i sista tabellraden:

- 9) Föregångare: Rad n är konturtabellbörjan (bakåt)

10) Efterföljare: Rad n innehåller konturbörjan (bakåt)

Ytterligare informationer

Tillåtna förflyttningskommandon, koordinatsystem

För konturprogrammering är följande G-kommandon tillåtna:

- G-grupp 1: G0, G1, G2, G3

Dessutom möjliga är:

- Rundning och avfasning
- Cirkelprogrammering via CIP och CT

Funktionerna Spline, Polynom och Gänga leder till fel.

Ändringar i koordinatsystemet genom tillkoppling av en frame är inte tillåtet mellan CONTPRON och EXECUTE. Samma gäller för ett byte mellan G70 och G71 resp. G700 och G710.

Ett byte av geometriaxlarna med GEOAX under förberedelsen av konturtabellen leder till ett larm.

Baksnittselement

Konturbeskrivningen av de enskilda baksnittselementen kan valfritt göras i ett underprogram eller i enskilda block.

Avspåning oberoende av den programmerade konturriktningen

Konturförberedelsen med CONTPRON utvidgades så att konturtabellen efter ditt anrop står till förfogande oberoende av den programmerade riktningen.

19.3 Upprätta kodad konturtabell (CONTDCON)

Vid den med `CONTDCON` tillkopplade konturförberedelsen blir de efterföljande anropade NC-blocken lagrade i en 6-spaltig konturtabell kodade för minnesbesparing. Varje konturelement motsvarar en tabellrad i konturtabellen. Med kännedom om de nedan angivna kodningsreglerna kan du t.ex. sammanställa DIN-kodprogram för cykler ur tabellraderna. I tabellraden med nummer 0 sparas data från utgångspunkten.

Syntax

Koppla till konturförberedelse:

`CONTDCON (<Konturtable>, <Bearbeitungsrichtung>)`

Koppla från konturförberedelse och återvända till den normala genomarbetsmoden:

`EXECUTE (<FEHLER>)`

Se " Koppla från konturförberedelse (EXECUTE) (Sida 689) "

Betydelse

<code>CONTDCON:</code>	Fördefinierad procedur för tillkoppling av konturförberedelsen för upprättande av en koderad konturtabell		
<code><Konturtable>:</code>	Namn på konturtabellen		
<code><Bearbeitungsrichtung>:</code>	Parameter för bearbetningsriktning		
	Typ:	INT	
	Värde:	0	Konturförberedelse enligt följden av konturblock (standardvärde)
		1	otillåten

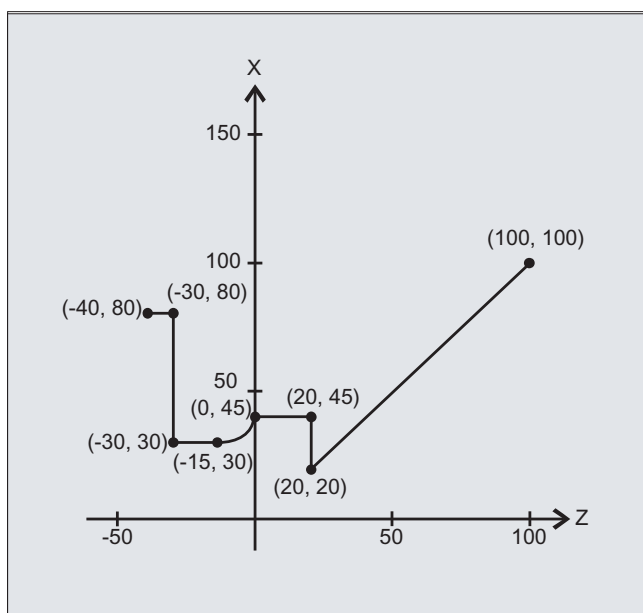
Märk

De för `CONTDCON` tillåtna G-kommandona i den programdel som ska bli tabell, är mer omfattande än de vid `CONTPRON`. Därutöver sparas också matningar och matningstyp per konturstycke.

Exempel

Upprätta en konturtabell med:

- Namnet "KTAB"
- Konturelementen (cirklar, linjer)
- Driftssätt: Svarva
- Bearbetningsriktning: framåt

**NC-program:**

Programkod	Kommentar
N10 DEF REAL KTAB[9,6]	; Konturtabell med namnet KTAB och 9 tabellrader. Dessa tillåter 8 konturblock. Parametervärdet 6 (spalttal i tabellen) är en fast storhet.
N20 DEF INT MODE = 0	; Variabel för bearbetningsriktningen. Standardvärde 0: endast i programmerad riktning för konturen.
N30 DEF INT ERROR = 0	; Variabel för felsvar.
...	
N100 G18 G64 G90 G94 G710	
N101 G1 Z100 X100 F1000	
N105 CONTDCON (KTAB, MODE)	; Anrop konturförberedelse (MODE får utelämnas).
N110 G1 Z20 X20 F200	; Konturbeskrivning.
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)	; Avsluta ifyllning av konturtabellen, omkoppling till normal programdrift.
...	

Konturtabell KTAB:

	Spaltindex					
	0	1	2	3	4	5
Radindex	Kontur-mode	Slutpunkt abskissa	Slutpunkt ordinata	Medelpunkt abskissa	Medelpunkt ordinata	Matning
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

Förklaring av spaltinnehållet:Rad 0: Kodning för **startpunkten**:Spalt 0: 10^0 (entalsplats): $G_0 = 0$ 10^1 (tiotalssplats): $G_{70} = 0$, $G_{71} = 1$, $G_{700} = 2$, $G_{710} = 3$

Spalt 1: Startpunkt abskissa

Spalt 2: Startpunkt ordinata

Spalt 3-4: 0

Spalt 5: Radindex för det sista konturstycket i tabellen

Rader 1-n: Poster för **konturstyckena**Spalt 0: 10^0 (entalsplats): $G_0 = 0$, $G_1 = 1$, $G_2 = 2$, $G_3 = 3$ 10^1 (tiotalssplats): $G_{70} = 0$, $G_{71} = 1$, $G_{700} = 2$, $G_{710} = 3$ 10^2 (hundratalssplats): $G_{90} = 0$, $G_{91} = 1$ 10^3 (tusentalssplats): $G_{93} = 0$, $G_{94} = 1$, $G_{95} = 2$, $G_{96} = 3$ 10^4 (tiusentalssplats): $G_{60} = 0$, $G_{44} = 1$, $G_{641} = 2$, $G_{642} = 3$ 10^5 (hundrausentalssplats): $G_9 = 1$

Spalt 1: Slutpunkt abskissa

Spalt 2: Slutpunkt ordinata

Spalt 3: Medelpunkt abskissa vid cirkelinterpolering

Spalt 4: Medelpunkt ordinata vid cirkelinterpolering

Spalt 5: Matning

Ytterligare informationer

Tillåtna förflyttningskommandon, koordinatsystem

För konturprogrammering är följande G-grupper och G-kommandon tillåtna:

G-grupp 1:	G0, G1, G2, G3
G-grupp 10:	G60, G64, G641, G642
G-grupp 11:	G9
G-grupp 13:	G70, G71, G700, G710
G-grupp 14:	G90, G91
G-grupp 15:	G93, G94, G95, G96, G961

Dessutom möjliga är:

- Rundning och avfasning
- Cirkelprogrammering via CIP och CT

Funktionerna Spline, Polynom och Gänga leder till fel.

Ändringar i koordinatsystemet genom tillkoppling av en frame är inte tillåtet mellan CONTDCON och EXECUTE. Samma gäller för ett byte mellan G70 och G71 resp. G700 och G710.

Ett byte av geometriaxlarna med GEOAX under förberedelsen av konturtabellen leder till ett larm.

Bearbetningsriktning

Den med CONTDCON skapade konturtabellen är avsedd för avspåning i den programmerade riktningen för konturen.

19.4 Fastställa skärningspunkt mellan två konturelement (INTERSEC)

INTERSEC fastställer skärningspunkten för två normerade konturelement ur med CONTPRON skapade konturtabeller.

Syntax

```
<Status>=INTERSEC(<Konturtabelle_1>[<Konturelement_1>],  
<Konturtabelle_2>[<Konturelement_2>],<Schnittpunkt>,<Bearbeitungsart>)
```

Betydelse

INTERSEC:	Fördefinierad funktion för fastställande av skärningspunkten för två konturelement ur med CONTPRON skapade konturtabeller		
<Status>:	Variabel för skärningspunktstatus		
	Typ:	BOOL	
	Värde:	TRUE	Skärningspunkt hittad
		FALSE	Ingen skärningspunkt hittad
<Konturtabelle_1>:	Namn på den första konturtabellen		
<Konturelement_1>:	Nummer på konturelementet i den första konturtabellen		
<Konturtabelle_2>:	Namn på den andra konturtabellen		
<Konturelement_2>:	Nummer på konturelementet i den andra konturtabellen		
<Schnittpunkt>:	Skärningspunkt-koordinater i det aktiva planet (G17 / G18 / G19)		
	Typ:	REAL	
<Bearbeitungsart>:	Parameter för bearbetningstypen		
	Typ:	INT	
	Värde:	0	Skärningspunktberäkning i det med parameter 2 aktiva planet (standardvärde)
		1	Skärningspunktberäkning oberoende av det överlämnade planet

Märk

Beakta att variablerna måste vara definierade innan de används.

Överföringen av konturerna kräver respekterandet av de med CONTPRON definierade värdena:

Parameter	Betydelse
2	Kodning av konturmode för rörelsen
3	Kontur-begynnelsepunkt abskissa
4	Kontur-begynnelsepunkt ordinata
5	Kontur-slutpunkt abskissa
6	Kontur-slutpunkt ordinata
9	Medelpunktskoordinat för abskissan (endast vid cirkel-kontur)
10	Medelpunktskoordinat för ordinatan (endast vid cirkel-kontur)

Exempel

Fastställa skärningspunkt för konturelement 3 i tabellen TABNAME1 och konturelement 7 i tabellen TABNAME2. Skärningspunkt-koordinaterna i det aktiva planet läggs i variabeln ISCOORD (1:a element = abskissa, 2:a element = ordinata). Existerar ingen skärningspunkt, görs ett hopp till KEINSCH (ingen skärningspunkt hittad).

Programkod	Kommentar
DEF REAL TABNAME1[12,11]	; Konturtabell 1
DEF REAL TABNAME2[10,11]	; Konturtabell 2
DEF REAL ISCOORD[2]	; Variabel för skärningspunkt-koordinater.
DEF BOOL ISPOINT	; Variabel för skärningspunktstatus.
DEF INT MODE	; Variabel för bearbetningstyp.
...	
MODE=1	; Beräkning oberoende av det aktiva planet.
N10 ISPOINT=INTERSEC(TABNAME1[3],TABNAME2[7], ISCOORD,MODE)	; Anrop skärningspunkt för konturelementen.
N20 IF ISPOINT==FALSE GOTOF KEINSCH	; Hopp till KEINSCH.
...	

19.5 Blockvis bortkörning av konturelement från en tabell (EXECTAB)

Med EXECTAB kan du blockvis köra bort konturelement från en tabell, som t.ex. skapades med CONTRON.

Syntax

```
EXECTAB (<Konturtabelle> [<Konturelement>])
```

Betydelse

EXECTAB:	Fördefinierad procedur för bortkörning av ett konturelement
<Konturtabelle>:	Namn på konturtabellen
<Konturelement>:	Nummer på konturelementet

Exempel

Konturelementen 0 till 2 i tabellen KTAB ska köras bort blockvis.

Programkod	Kommentar
N10 EXECTAB(KTAB[0])	; Förflytta element 0 i tabellen KTAB.
N20 EXECTAB(KTAB[1])	; Förflytta element 1 i tabellen KTAB.
N30 EXECTAB(KTAB[2])	; Förflytta element 2 i tabellen KTAB.

19.6 Beräkna cirkeldata (CALCDAT)

Med **CALCDAT** kan du ur tre eller fyra kända cirkelpunkter beräkna radien och cirkelmedelpunkt-koordinaterna. De angivna punkterna måste vara olika.

Vid 4 punkter som inte ligger exakt på cirkeln bildas för cirkelmedelpunkt och radie ett medelvärde.

Märk

Räkneföreskrift för medelvärdesberäkning

Cirkelbågsberäkningen utförs 4 x:

1. med cirkelpunkt 1, 2, 3
2. med cirkelpunkt 1, 2, 4
3. med cirkelpunkt 1, 3, 4
4. med cirkelpunkt 2, 3, 4

Cirkelpunkt-koordinaterna abskissavärde och ordinatavärde beräknas genom att abskissavärdena resp. ordinatavärdena för de fyra cirkelbågberäkningarna adderas och delas med 4.

Radien beräknas genom att dra roten ur summan för de fyra radierna i cirkelbågberäkningarna och multiplicera resultatet med 0,5.

Syntax

```
<Status>=CALCDAT (<Kreispunkte> [<Anzahl>, <Art>], <Anzahl>, <Ergebnis>)
```

Betydelse

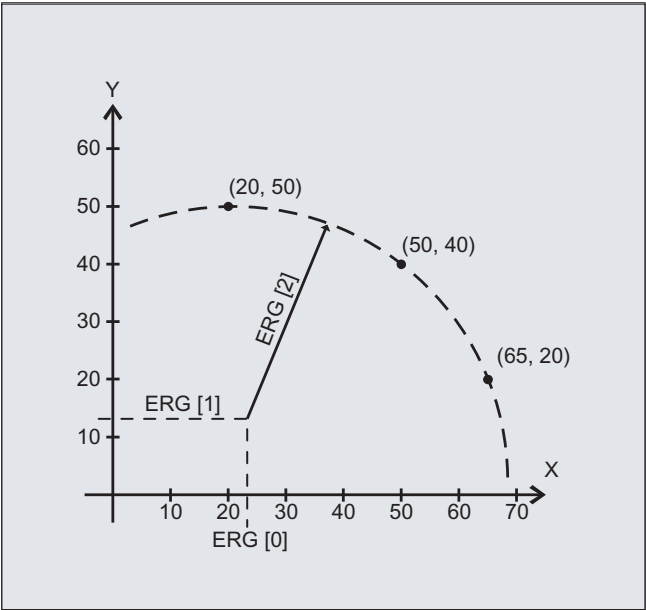
CALCDAT:	Fördefinierad funktion för beräkning av radie och medelpunkt-koordinater för en cirkel av 3 eller 4 punkter		
<Status>:	Variabel för cirkelberäkningstatus		
	Typ:	BOOL	
	Värde:	TRUE	De angivna punkterna ligger på en cirkel.
		FALSE	De angivna punkterna ligger inte på en cirkel.
<Kreispunkte>[]:	Variabel för angivnade av cirkelpunkterna med parametrar:		
	<Anzahl>:	Antal cirkelpunkter (3 eller 4)	
	<Art>:	Typ av koordinatuppgift, t.ex. 2 för 2-punkt-koordinater	
<Anzahl>:	Parameter för antalet till beräkningen använda punkter (3 eller 4)		
<Ergebnis>[3]:	Variabel för resultat:		
	Uppgift av cirkelmedelpunkt-koordinater och radie		
	0	Cirkelmedelpunkt-koordinat: Abskissavärde	
	1	Cirkelmedelpunkt-koordinat: Ordinativärde	
	2	Radie	

Märk

Beakta att variablerna måste vara definierade innan de används.

Exempel

För tre punkter ska fastställas om de ligger på ett cirkelavsnitt.



Programkod	Kommentar
N10 DEF REAL PKT[3,2]=(20,50,50,40,65,20)	; Variabel för angivande av cirkelpunkter.
N20 DEF REAL ERG[3]	; Variabel för resultat
N30 DEF BOOL STATUS	; Variabel för status.
N40 STATUS=CALCDAT(PKT,3,ERG)	; Anrop av de fastställda cirkeldata.
N50 IF STATUS == FALSE GOTOF ERROR	; Hopp till fel.

19.7 Koppla från konturförberedelse (EXECUTE)

Med **EXECUTE** kopplas konturförberedelsen från och samtidigt tillbaka till den normala genomarbetsmoden.

Syntax

EXECUTE (<FEHLER>)

Betydelse

EXECUTE:	Fördefinierad procedur för att avsluta konturförberedelsen	
<Fehler>:	Variabel för felsvar	
	Typ:	INT
	Värdet för variabeln visar om konturen kunde förberedas felfritt:	
	0	Fel
	1	Inget fel

Exempel

Programkod

```
...  
N30  CONTPRON (...)  
N40  G1 X... Z...  
...  
N100 EXECUTE (...)  
...
```


Programmera cykler externt

20.1 Teknologiska cykler

20.1.1 Inledning

Innehåll

Detta kapitel innehåller beskrivningen till cyklerna för teknologierna svarva, fräsa, slipa.

Uppbyggnad

Beskrivningen för en cykel är uppbyggd på följande sätt:

- **Syntax**
Cykelnamn och anropsordningsföljd för överföringsparametrarna
- **Parametrar**
Tabell för förklaring av de enskilda parametrarna

Parameterbeskrivning

I tabellen finns följande data angivna till en parameter: Namn, beskrivning, värdeområde och beroende av andra parametrar.

Spalten för hänvisning till parametern i masken tjänar till att göra det lättare att åter hitta av styrningen programmerade värden vid återöversättningen av externt genererade cykelanrop.

Parameter "endast för ytan"

I tabellen är parametrar markerade med "endast för ytan". Dessa är inte av betydelse för cykelns funktion. De behövs endast för att fullständigt kunna återöversätta cykelanrop. Är de inte programmerade kan cykeln trots detta återöversättas, fälten är då lämpligt markerade med färg och måste vara ifyllda i masken.

Parameter "reserverad"

Parametrar som beskrivs med "reserverad", måste programmeras med värdet 0 eller tomkomma, för att tillordningen av de följande anropsparametrarna stämmer med de interna cykelparametrarna. Undantag: vid stringparametern värde "" eller tomkomma.

Upprepa cykler på positionsmönster

Borr- och fräscykler kan upprepas på positionsmönster (modala anrop). Före cykeln ska då i samma rad skrivas `MCALL`, t.ex. `MCALL CYCLE83 ( . . . )`.

Märk

Är vissa överföringsparametrar (t.ex. <_VARI>, <_GMODE>, <_DMODE>, <_AMODE>) indirekt programmerade som parametrar, öppnas inmatningsmasken vid återöversättningen, men kan inte sparas, eftersom det till vissa urvalsutor inte finns någon entydig tillordning.

20.1.2 Teknologis specifik översikt

I den följande översiktstabellen finns alla tillgängliga externa programmerbara teknologiska cykler uppförda och tillordnade respektive teknologi:

Teknologi	Teknologisk cykel
Borra	• CYCLE81 - borra, centrera (Sida 731) • CYCLE82 - borra, plansänka (Sida 732) • CYCLE85 - brotscha (Sida 741) • CYCLE86 - ursvarvning (Sida 742) • CYCLE83 - djuphålsborra (Sida 735) • CYCLE830 - djuphålsborra 2 (Sida 765) • CYCLE84 - Gängtappning utan flytande gänghållare (Sida 738) • CYCLE840 - gängtappning med flytande gänghållare (Sida 774) • CYCLE78 - fräsa hålgänga (Sida 727) • CYCLE802 - valfria positioner (Sida 763) • HOLES1 - hålrad (Sida 694) • CYCLE801 - gitter eller ram (Sida 761) • HOLES2 - hålrad (Sida 694)
Svarva	• CYCLE951- avspåna (Sida 785) • CYCLE930 - Instick (Sida 780) • CYCLE940 - fristick former (Sida 782) • CYCLE99 - gängsvarvning (Sida 751) • CYCLE98 - kopplade gängor (Sida 747) • CYCLE92 - avstick (Sida 743)
Kontursvarva	• CYCLE62 - konturanrop (Sida 713) • CYCLE952 - kontursticka (Sida 788)

Teknologi	Teknologisk cykel
Fräsa	• CYCLE61- planfräsa (Sida 711) • POCKET3 - fräsa fyrkantficka (Sida 696) • POCKET4 - fräsa cirkelsficka (Sida 699) • CYCLE76 - fräsa fyrkanttapp (Sida 723) • CYCLE77 - fräsa cirkeltapp (Sida 725) • CYCLE79 - flerkant (Sida 729) • SLOT1 - längsspår (Sida 701) • SLOT2 - cirkelspår (Sida 704) • CYCLE899 - fräsa öppet spår (Sida 777) • LONGHOLE - långhål (Sida 706) • CYCLE70 - gängfräsning (Sida 717) • CYCLE60 - gravrycykel (Sida 708)
Konturfräsning	• CYCLE62 - konturanrop (Sida 713) • CYCLE72 - banfräsa (Sida 719) • CYCLE63 - fräsa konturficka (Sida 714) • CYCLE64 - förborra konturficka (Sida 716)
Slipning	• CYCLE495 - profilera (Sida 756) • CYCLE435 - ställa in koordinatsystem för skärpverktyg (Sida 756) • CYCLE4071 - längsslipning med ansättning i vändpunkten (Sida 794) • CYCLE4072 - längsslipning mit ansättning i vändpunkten (Sida 795) • CYCLE4073 - längsslipning med kontinuerlig ansättning (Sida 799) • CYCLE4074 - Längsslipning med kontinuerlig ansättning och avbrottssignal (Sida 800) • CYCLE4075 - flatslipning med ansättning i vändpunkten (Sida 803) • CYCLE4077 - flatslipning med ansättning i vändpunkten och avbrottssignal (Sida 806) • CYCLE4078 - flatslipning med kontinuerlig ansättning (Sida 809) • CYCLE4079 - flatslipning med intermittent ansättning (Sida 811)
Övrigt	• CYCLE800 - vridning (Sida 758) • CYCLE832 - High Speed Settings (Sida 771)
Alla	• GROUP_BEGIN - början programblock (Sida 813) • GROUP_END - slut programblock (Sida 814) • GROUP_ADDEND - slut inkörningstillägg (Sida 814)

20.1.3 HOLES1 - hålråd

Syntax

HOLES1(<SPCA>, <SPCO>, <STA1>, <FDIS>, <DBH>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
1	X0	<SPCA>	REAL	Referenspunkt för hålråd i den 1:a axeln (abs)		
2	Y0	<SPCO>	REAL	Referenspunkt för hålråd i den 2:a axeln (abs)		
3	α 0	<STA1>	REAL	Grund-rotationsvinkel (vinkel till 1:a axeln)		
4	L0	<FDIS>	REAL	Avstånd 1:a hålet från referenspunkt		
5	L	<DBH>	REAL	Avstånd mellan hålen		
6	N	<NUM>	INT	Antal hål		
7		<_VARI>	INT	reserverat		
8		<_UMODE>	INT	reserverat		
9		<_HIDE>	STRING [200]	Gömda positioner - max. 198 tecken - Uppgift över det fortlöpande positionsnumret, t.ex. "1,3" (positioner 1 och 3 utförs inte)		
10		<_NSP>	INT	reserverat		
11		<_DMODE>	INT	Indikeringsmode		
				ENTAL:	Bearbetningsplan G17/18/19	
					0 =	Kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
					1 =	G17 (endast aktiv i cykeln)
					2 =	G18 (endast aktiv i cykeln)
	3 =	G19 (endast aktiv i cykeln)				

20.1.4 HOLES2 - hålråd

Syntax

HOLES2(<CPA>, <CPO>, <RAD>, <STA1>, <INDA>, <NUM>, <_VARI>,
<_UMODE>, <_HIDE>, <_NSP>, <_DMODE>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	X0	<CPA>	REAL	Medelpunkt för hålcirkel i den 1:a axeln (abs) Referenspunkt i den 1:a axeln
2	Y0	<CPO>	REAL	Medelpunkt för hålcirkel i den 2:a axeln (abs) Referenspunkt i den 2:a axeln
3	R	<RAD>	REAL	Radie för hålcirkeln
4	$\alpha 0$	<STA1>	REAL	Begynnelsevinkel eller 1:a roterande axelposition:
5	$\alpha 1$	<INDA>	REAL	Stegkopplingsvinkel (endast vid hålcirkel)
				< 0 = Medurs > 0 = Moturs
6	N	<NUM>	INT	Antal positioner
7		<_VARI>	INT	Bearbetningstyp
				ENTAL: reserverat
				TIOTAL: Positioneringstyp
				0 = Uppsöka position - linjärt
				1 = Uppsöka position - på cirkelbana
				HUNDRATAL: reserverat
				TUSENTAL: Cirkelmönster
				0 = Kompatibilitetsmode, om INDA = 0 då helcirkel, INDA <> 0 då delcirkel)
				1 = Helcirkel
				2 = Delcirkel
				TIOTUSENTAL: Positionsmönster med roterande axel
				0 = XY (utan roterande axel) (vid XY)
				1 = XA (X-axel och roterande axel runt X) (endast vid XA)
				2 = YB (Y-axel och roterande axel runt Y) (endast vid YB)
				3 = ZC (Z-axel och roterande axel runt C) (endast vid ZC)
				ENMILJONTAL + HUNDRATUSENTAL: Offset (vid flera roterande axlar runt samma axel; när index för stort, då 1:a axeln)
				00 = 1. A, B eller C-axel
				01 = 2. A, B eller C-axel
				...
				10 = 20. A, B eller C-axel
8		<_UMODE>	INT	reserverat
9		<_HIDE>	STRING [200]	reserverat
10		<_NSP>	INT	reserverat

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
11		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/18/19
				0 = Kompatibilitet, det före cykelanrop verk- samma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)

20.1.5 POCKET3 - fräsa fyrkantficka

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<_RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Fickdjup (abs/ink), se <_AMODE>)
5	L	<_LENG>	REAL	Ficklängd (ink, matas in med förtecken)
6	W	<_WID>	REAL	Fickbredd (ink, matas in med förtecken)
7	R	<_CRAD>	REAL	Hörnradie för fickan
8	X0	<_PA>	REAL	Referenspunkt, 1:a axeln (abs)
9	YO	<_PO>	REAL	Referenspunkt, 2:a axeln (abs)
10	α0	<_STA>	REAL	Vridvinkel, vinkel mellan längsaxel (L) och 1:a axeln
11	DZ	<_MID>	REAL	Maximal djupansättning
12	UXY	<_FAL>	REAL	Finbearbetningsmån i planet
13	UZ	<_FALD>	REAL	Finbearbetningsmån djup
14	F	<_FFP1>	REAL	Matning i planet
15	FZ	<_FFD>	REAL	Ansättningsmatning djup
16		<_CDIR>	INT	Fräsriktning:
				0 = Medmatning 1 = Motmatning

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
17		<_VARI>	INT	Bearbetningstyp
				ENTAL:
				1 = grovbearbetning
				2 = finbearbetning
				4 = finbearbetning av kant
				5 = fasning
				TIOTAL:
				0 = förborrad, ansättning med G0
				1 = lodrät, ansättning med G1
				2 = helikal
				3 = pendla på ficklängdsaxeln
				HUNDRATAL: reserverat
18	DXY	<_MIDA>	REAL	Maximal ansättning i planet, enhet se <_AMODE>
19	L1	<_AP1>	REAL	Längd för förbearbetningen (ink)
20	W1	<_AP2>	REAL	Bredd för förbearbetningen (ink)
21	AZ	<_AD>	REAL	Djup för förbearbetningen (ink)
22	ER	<_RAD1>	REAL	Radie för helixbanan vid helikal nermatning
	EW			Maximal nermatningsvinkel för pendlande
23	EP	<_DP1>	REAL	Helixstigning vid helikal nermatning
24		<_UMODE>	INT	reserverat
25	FS	<_FS>	REAL	Fasbredd (ink)
26	ZFS	<_ZFS>	REAL	Nermatningsdjup (verktygsspets) vid fasning (abs/ink) se <_AMODE>

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
27		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TIOTAL: reserverat
				HUNDRATAL: Val bearbetning/enda startpunktberäkning
				0 = kompatibilitetsmode
				1 = normal bearbetning
				TUSENTAL: Måttställning via mitt/hörn
				0 = kompatibilitetsmode
				1 = måttställning via mitt
				2 = måttställning hörnpunkt, fickläge +LENG/+WID
				3 = måttställning hörnpunkt, fickläge -LENG/+WID
				4 = måttställning hörnpunkt, fickläge +LENG/-WID
				5 = måttställning hörnpunkt, fickläge -LENG/-WID
				TIOTUSENTAL: Komplettbearbetning/efterbearbetning
28		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL: Matningstyp: G-grupp (G94/G95) för yt- och djupmatning
				0 = kompatibilitetsmode
				1 = G-kommando som före cykelanrop. G94/G95 för yt- och djupmatning lika
				HUNDRATAL: --- reserverat
				TUSENTAL: --- reserverat
				TIOTUSENTAL: Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
29		<_AMODE>	INT	Alternativmode
				ENTAL:
				Fickdjup (Z1)
				0 = absolut (kompatibilitetsmode)
				1 = inkrementell
				TIOTAL:
				Enhet för ansättning i planet (DXY)
				0 = mm
				1 = % av verktygsdiametern
				HUNDRATAL:
				Nermatningsdjup vid fasning (ZFS)
				0 = absolut
				1 = inkrementell

20.1.6 POCKET4 - fräsa cirkelsficka

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<_RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Fickdjup (abs/ink), se <_AMODE>
5	Ø	<_CDIAM>	REAL	Fickdiameter eller fickradie, se <_DMODE>
6	X0	<_PA>	REAL	Referenspunkt, 1:a axeln (abs)
7	Y0	<_PO>	REAL	Referenspunkt, 2:a axeln (abs)
8	DZ	<_MID>	REAL	Maximal djupansättning, se <_VARI> = planvis Maximal helixstigning, se <_VARI> = helikal
9	UXY	<_FAL>	REAL	Finbearbetningsmån i planet
10	UZ	<_FALD>	REAL	Finbearbetningsmån djup
11	F	<_FFP1>	REAL	Matning för ytbearbetning
12	FZ	<_FFD>	REAL	Ansättningsmatning djup
13		<_CDIR>	INT	Fräsriktning
				0 = Medmatning 1 = Motmatning

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
14		<_VARI>	INT	Bearbetningstyp
				ENTAL:
				Bearbetning
				1 = grovbearbetning
				2 = finbearbetning
				4 = finbearbetning av kant
				5 = fasning
				TIOTAL:
				Ansättningstyp (grovbearbetning och finbearbetning)
				0 = förborrad, ansättning med G0 (ficka är förbearbetad)
15	DXY	<_MIDA>	REAL	Maximal ansättning i planet, se <_AMODE>, 0 = 0,8 x verktygsdiameter
				16 Ø
				<_AP1>
				REAL
				Diameter/radie för förbearbetningen (ink)
				17 AZ
				<_AD>
				REAL
				Djup för förbearbetningen (ink)
				18 ER
19	EP	<_RAD1>	REAL	Radie för helixbanan vid helikal nermatning
				<_DP1>
				REAL
				Helixstigning vid nermatning på helixbana
				20
				<_UMODE>
				INT
				reserverat
				21 FS
				<_FS>
22	ZFS	<_ZFS>	REAL	Fasbredd (ink)
				22 ZFS
				<_ZFS>
				REAL
				Nermatningsdjup (verktygsspets) vid fasning (abs/ink) se <_AMODE>
				23
				<_GMODE>
				INT
				Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL:
23		<_GMODE>	INT	reserverat
				TIOTAL:
				reserverat
				HUNDRATAL:
				Bearbetning/startpunktberäkning
				0 = kompatibilitetsmode
				1 = normal bearbetning
				TUSENTAL:
				reserverat
				TIOTUSENTAL:
23		<_GMODE>	INT	Komplettbearbetning/efterbearbetning
				0 = kompatibilitetsmode (<_AP1> och <_AD> behandla som tidigare)
				1 = komplettbearbetning
				2 = efterbearbetning

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
24		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL:
				Matningstyp: G-grupp (G94/G95) för yt- och djupmatning
				0 = kompatibilitetsmode
				1 = G-kommando som före cykelanrop. G94/G95 för yt- och djupmatning lika
				HUNDRATAL:
25		<_AMODE>	INT	Alternativmode
				ENTAL:
				Fickdjup (Z1)
				0 = absolut (kompatibilitetsmode)
				1 = inkrementell
				TIOTAL:
				Enhet för ansättningsbredd (DXY)
				0 = mm
				1 = % av verktygsdiametern
				HUNDRATAL:
				Nermatningsdjup vid fasning (ZFS)
				0 = absolut
				1 = inkrementell

20.1.7 SLOT1 - längsspår

Syntax

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>)

Parametrar

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse		
1	RP	<RTP>	REAL	Återgångsplan (abs)		
2	Z0	<RFP>	REAL	Referenspunkt för verktygsaxeln (abs)		
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)		
4	Z1	<_DP>	REAL	Spårdjup (abs)		
5		<_DPR>	REAL	Spårdjup (ink), relaterat till Z0 (mata in utan förtecken)		
6		<NUM>	INT	Antal spår = 1		
7	L	<LENG>	REAL	Spårets längd		
8	W	<WID>	REAL	Spårets bredd		
9	X0	<_CPA>	REAL	Referenspunkt, 1:a axeln i planet		
10	Y0	<_CPO>	REAL	Referenspunkt, 2:a axeln i planet		
11		<RAD>	REAL	reserverat		
12	α	<STA1>	REAL	Rotationsvinkel		
13		<INDA>	REAL	reserverat		
14	FZ	<FFD>	REAL	Ansättningsmatning djup		
15	F	<FFP1>	REAL	Matning		
16	DZ	<_MID>	REAL	Maximal djupansättning		
17		<CDIR>	INT	Fräsriktning	0 =	Medmatning
					1 =	Motmatning
18	UXY	<_FAL>	REAL	Finbearbetningsmån plan eller spårkant		
19		<VARI>	INT	Bearbetningstyp		
				ENTAL:		
					0 =	reserverat
					1 =	grovbearbetning
					2 =	finbearbetning
					4 =	finbearbetning av kant (bearbeta endast kanten)
					5 =	fasning
				TIOTAL:	Uppsökning	
					0 =	förborrad, ansättning med G0 (spår är förbearbetat)
					1 =	lodrät, ansättning med G1
					2 =	helikal
					3 =	pendlande
				HUNDRATAL:	reserverat	
20	DZF	<_MIDF>	REAL	reserverat		
21	FF	<FFP2>	REAL	reserverat		
22	SF	<SSF>	REAL	reserverat		

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
23	UZ	<_FALD>	REAL	Finbearbetningsmån djup
24	ER	<_STA2>	REAL	Radie för helixbanan vid helikal nermatning
	EW			Maximal nermatningsvinkel för pendlande
25	EP	<_DP1>	REAL	Nermatningsdjup per varv för helix
26		<_UMODE>	INT	reserverat
27	FS	<_FS>	REAL	Fasbredd (ink) vid fasning
28	ZFS	<_ZFS>	REAL	Nermatningsdjup (verktygsspets) vid fasning (abs/ink) se <_AMODE>
29		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivrädena)
				ENTAL: reserverat
				TIOTAL: reserverat
				HUNDRATAL: Val bearbetning eller endast startpunkt-beräkning
				1 = normal bearbetning
				TUSENTAL: Måttställning referenspunkt, spårläge
				0 = Centrum
				1 = vänster invändigt +L
				2 = höger invändigt -L
				3 = vänster kant +L
				4 = höger kant -L
30		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL: reserverat
				HUNDRATAL: reserverat
				TUSENTAL: Kännetecken SW-version
				1 = Funktionsutvidgning SLOT1
				TIOTUSENTAL: Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
31		<_AMODE>	INT	Alternativmode
				ENTAL:
				Slutdjup Z1 (abs/ink)
				0 = kompatibilitet
				1 = Z1 (ink)
				2 = Z1 (abs)
				TIOTAL:
				reserverat
				HUNDRATAL:
				Nermatningsdjup vid fasning ZFS
				0 = ZFS (abs)
				1 = ZFS (ink)

Märk

Cykeln är utrustad med nya funktioner jämfört med tidigare SW-versioner. Det har till följd att vissa parametrar i inmatningsmasken inte längre visas (<NUM>, <RAD>, <INDA>). Flera spår på ett positionsmönster är programmerbara med hjälp av "MCALL" och anrop av det önskade positionsmönstret, t.ex. HOLES2.

20.1.8 SLOT2 - cirkelspår

Syntax

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>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<RTP>	REAL	Återgångsplan (abs)
2	Z0	<RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Spårdjup (abs)
5		<_DPR>	REAL	Spårdjup (ink), relaterat till Z0 (mata in utan förtecken)
6	N	<NUM>	INT	Antal spår
7	$\alpha 1$	<AFSL>	REAL	Spårets öppningsvinkel
8	W	<WID>	REAL	Spårets bredd
9	X0	<_CPA>	REAL	Referenspunkt = medelpunkt för cirkeln, 1:a axeln i planet
10	Y0	<_CPO>	REAL	Referenspunkt = medelpunkt för cirkeln, 2:a axeln i planet
11	R	<RAD>	REAL	Cirkelns radie

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
12	$\alpha 0$	<STA1>	REAL	Begynnelsevinkel
13	$\alpha 2$	<INDA>	REAL	Framkopplingsvinkel
14	FZ	<FFD>	REAL	Ansättningsmatning djup
15	F	<FFP1>	REAL	Matning
16	DZ	<_MID>	REAL	Maximal djupansättning
17		<CDIR>	INT	Fräsriktning
				0 = Medmatning
				1 = Motmatning
18	UXY	<_FAL>	REAL	Finbearbetningsmån plan eller spårkant
19		<VARI>	INT	Bearbetningstyp
				ENTAL:
				0 = komplettbearbetning
				1 = grovbearbetning
				2 = finbearbetning
				3 = finbearbetning av kant
				5 = fasning
				TIOTAL:
				0 = mellanpositionera med G0-linje
				1 = mellanpositionera på cirkelbana
				HUNDRATAL:
				reserverat
				TUSENTAL:
				0 = kompatibilitetsmode, när <INDA> = 0 då helcirkel, <INDA> <> 0 då delcirkel)
				1 = Helcirkel
				2 = Delcirkel
20	DZF	<_MIDF>	REAL	reserverat
21		<FFP2>	REAL	reserverat
22		<SSF>	REAL	reserverat
23	FF	<_FFCP>	REAL	reserverat
24		<_UMODE>	INT	reserverat
25	FS	<_FS>	REAL	Fasbredd (ink)
26	ZFS	<_ZFS>	REAL	Nermatningsdjup (verktygsspets) vid fasning (abs/ink) se <_AMODE>
27		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivrärderna)
				ENTAL:
				reserverat
				TIOTAL:
				reserverat
				HUNDRATAL:
				Val bearbetning eller endast startpunktberäkning
				0 = kompatibilitetsmode
				1 = normal bearbetning

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
28		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL:
				reserverat
				HUNDRATAL:
				reserverat
29		<_AMODE>	INT	Alternativmode
				ENTAL:
				Slutdjup Z1 (abs/ink)
				0 = kompatibilitet
				1 = Z1 (ink)
				2 = Z1 (abs)
				TIOTAL:
				reserverat
				HUNDRATAL:
				Nermatningsdjup vid fasning ZFS
				0 = ZFS (abs)
				1 = ZFS (ink)

20.1.9 LONGHOLE - långhål

Syntax

LONGHOLE(<RTP>, <RFP>, <SDIS>, <_DP>, <_DPR>, <NUM>, <LENG>, <_CPA>, <_CPO>, <RAD>, <STAL>, <INDA>, <FFD>, <FFP1>, <MID>, <_VARI>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
4	Z1	<_DP>	REAL	Långhålsdjup (abs)
5		<_DPR>	REAL	Långhålsdjup (ink), relaterat till Z0 (mata in utan förtecken)
6		<NUM>	INT	Antal långhål = 1
7	L	<LENG>	REAL	Längd långhål
8	X0	<_CPA>	REAL	Referenspunkt, 1:a axeln i planet
9	Y0	<_CPO>	REAL	Referenspunkt, 2:a axeln i planet
10		<RAD>	REAL	reserverat
11	α0	<STA1>	REAL	Rotationsvinkel
12		<INDA>	REAL	reserverat
13	FZ	<FFD>	REAL	Ansättningsmatning djup
14	F	<FFP1>	REAL	Matning
15	DZ	<MID>	REAL	Maximal djupansättning
16		<_VARI>	INT	Bearbetningstyp ENTAL: Ansättningstyp 1 = lodrät, ansättning med G1 3 = pendlande HUNDRATAL: reserverat
17		<_UMODE>	INT	reserverat
18		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena) ENTAL: reserverat TIOTAL: reserverat HUNDRATAL: Val bearbetning eller endast startpunktberäkning 0 = kompatibilitetsmode 1 = normal bearbetning TUSENTAL: Måttsättning referenspunkt, spårläge 0 = Centrum 1 = vänster invändigt +L 2 = höger invändigt -L 3 = vänster kant +L 4 = höger kant -L

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
19		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL:
				Matningstyp: G-grupp (G94/G95) för yt- och djupmatning
				0 = kompatibilitetsmode
				1 = G-kommando som före cykelanrop. G94/G95 för yt- och djupmatning lika
				HUNDRATAL:
				reserverat
				TUSENTAL:
				Kännetecken SW-version
				1 = funktionsutvidgning LONGHOLE (måttställning referenspunkt)
20		<_AMODE>	INT	Alternativmode
				ENTAL:
				Slutdjup Z1 (abs/ink)
				0 = kompatibilitet
				1 = Z1 (ink)
				2 = Z1 (abs)

Märk

Cykeln är utrustad med nya funktioner jämfört med tidigare SW-versioner. Det har till följd att vissa parametrar i inmatningsmasken inte längre visas (<NUM>, <RAD>, <INDA>). Flera spår på ett positionsmönster är programmerbara med hjälp av "MCALL" och anrop av det önskade positionsmönstret, t. ex. HOLES2.

20.1.10 CYCLE60 - gravrycykel

Syntax

```
CYCLE60 (<_TEXT>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_PA>,
<_PO>, <_STA>, <_CP1>, <_CP2>, <_WID>, <_DF>, <_FFD>, <_FFP1>,
<_VARI>, <_CODEP>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>)
```


Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1		<_TEXT>	STRING [200]	Text som ska graveras (maximalt 100 tecken)
2	RP	<_RTP>	REAL	Återgångsplan (abs)
3	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
4	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referensplanet, mata in utan förtecken)
5	Z1	<_DP>	REAL	Djup (abs), se <_AMODE>
6	Z1	<_DPR>	REAL	Djup (ink), se <_AMODE>
7	X0	<_PA>	REAL	Referenspunkt, 1:a Axeln i planet (abs) - rätvinklig, se <_VARI>
	R			Referenspunkt, längd (radie) - polär, se <_VARI>
8	Y0	<_PO>	REAL	Referenspunkt, 2:a Axeln i planet (abs) - rätvinklig, se <_VARI>
	$\alpha 0$			Referenspunkt, vinkel relaterad till den 1:a axeln - polär, se <_VARI>
9	$\alpha 1$	<_STA>	REAL	Textriktning, vinkel för textlinjen relaterad till den 1:a axeln, se <_VARI>
10	XM	<_CP1>	REAL	Medelpunkt för cirkeln, 1:a Axeln i planet (abs) - rätvinklig, se <_VARI>
	LM			Medelpunkt till textcirkeln, längd (radie) relaterad till WNP - polär, se <_VARI>
11	YM	<_CP2>	REAL	Medelpunkt för cirkeln, 2:a axeln i planet (abs) - rätvinklig, se <_VARI>
	αM			Medelpunkt till textcirkeln, vinkel relaterad till den 1:a axeln - polär, se <_VARI>
12	W	<_WID>	REAL	Teckenhöjd (mata in utan förtecken)
13	DX1	<_DF>	REAL	Teckenavstånd / total bredd, se <_VARI>
	$\alpha 2$			Öppningsvinkel, se <_VARI>
14	FZ	<_FFD>	REAL	Ansättningsmatning djup, se <_DMODE>
15	F	<_FFP1>	REAL	Matning för ytbearbetning

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
16		<_VARI>	INT	Bearbetning (uppriktning och referenspunkt för gravyrtexten)
				ENTAL:
				Referenspunkt
				0 = rätvinklig
				1 = polär
				TIOTAL:
				Textuppriktning
				0 = Text på en linje
				1 = Text på en cirkelbåge uppe
				2 = Text på en cirkelbåge nere
				HUNDRATAL:
				reserverat
				TUSENTAL:
				Referenspunkt för texten horisontal
				0 = vänster
				1 = koncentrisk
				2 = höger
				TIOTUSENTAL:
				Referenspunkt för texten vertikal
				0 = nere
				1 = koncentrisk
				2 = uppe
				HUNDRATUSENTAL:
				Textlängd
				0 = Teckenavstånd
				1 = Total bredd för texten (endast vid linjär text)
				2 = Öppningsvinkel (endast vid text på cirkelbåge)
				ENMILJONTAL:
				Cirkelmedelpunkt
				0 = rätvinklig (kartesisk)
				1 = polär
				TIOMILJONTAL:
				Spegelskrift
				0 = kompatibilitet
				1 = Spegelskrift TILL
				2 = Spegelskrift FRÅN
17		<_CODEP>	INT	Nummer för codepage för skriften (för tillfället endast 1252)
18		<_UMODE>	INT	reserverat
19		_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL:
				reserverat
				TIOTAL:
				reserverat
				HUNDRATAL:
				Val bearbetning/enda
				startpunktberäkning
				0 = kompatibilitetsmode
				1 = normal bearbetning

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
20		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL:
				Matningstyp: G-grupp (G94/G95) för yt- och djupmatning
				0 = kompatibilitetsmode
				1 = G-kommando som före cykelanrop. G94/G95 för yt- och djupmatning lika
21		<_AMODE>	INT	Alternativmode
				ENTAL:
				Slutdjup (<_DP>, <_DPR>)
				0 = kompatibilitet
				1 = inkrementell (<_DPR>)
				2 = absolut (<_DP>)

20.1.11 CYCLE61- planfräsa

Syntax

```
CYCLE61 (<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_PA>, <_PO>, <_LENG>,
<_WID>, <_MID>, <_MIDA>, <_FALD>, <_FFP1>, <_VARI>, <_LIM>,
<_DMODE>, <_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<_RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln, höjd råämne (abs)
3	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Höjd färdig detalj (abs/ink), se <_AMODE>
5	X0	<_PA>	REAL	Hörnpunkt 1 i den 1:a axeln (abs)
6	Y0	<_PO>	REAL	Hörnpunkt 1 i den 2:a axeln (abs)
7	X1	<_LENG>	REAL	Hörnpunkt 2 i den 1:a axeln, se <_AMODE>
8	Y1	<_WID>	REAL	Hörnpunkt 2 i den 2:a axeln, se <_AMODE>
9	DZ	<_MID>	REAL	Maximal djupansättning
10	DXY	<_MIDA>	REAL	Maximal ansättning i planet, (enhet se <_AMODE>)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
11	UZ	<_FALD>	REAL	Finbearbetningsmån djup		
12	F	<_FFP1>	REAL	Arbetsmatning		
13		<_VARI>	INT	Bearbetningstyp		
				ENTAL:	Bearbetning	
					1 =	grovbearbetning
					2 =	finbearbetning
				TIOTAL:	Bearbetningsriktning	
					1 =	parallell till 1:a axeln. en riktning
					2 =	parallell till 2:a axeln. en riktning
					3 =	parallell till 1:a axeln, växlande riktning
					4 =	parallell till 2:a axeln, växlande riktning
14		<_LIM>	INT	Begränsningar		
				ENTAL:	Begränsning 1:a axeln minus	
					0 =	nej
					1 =	ja
				TIOTAL:	Begränsning 1:a axeln plus	
					0 =	nej
					1 =	ja
				HUNDRATAL:	Begränsning 2:a axeln minus	
					0 =	nej
					1 =	ja
				TUSENTAL:	Begränsning 2:a axeln plus	
					0 =	nej
					1 =	ja
15		<_DMODE>	INT	Indikeringsmode		
				ENTAL:	Bearbetningsplan G17/18/19	
					0 =	kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
					1 =	G17 (endast aktiv i cykeln)
					2 =	G18 (endast aktiv i cykeln)
					3 =	G19 (endast aktiv i cykeln)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
16		<_AMODE>	INT	Alternativmode
				ENTAL:
				Slutdjup (<_DP>)
				0 = absolut
				1 = inkrementell
				TIOTAL:
				Enhet för ansättning i planet (<_MIDA>)
				0 = mm
				1 = % av verktygsdiameteren
				HUNDRATAL:
				reserverat
				TUSENTAL:
				Längd på ytan
				0 = inkrementell
				1 = absolut
				TIOTUSENTAL:
				Bredd på ytan
				0 = inkrementell
				1 = absolut

20.1.12 CYCLE62 - konturanrop

Syntax

CYCLE62 (<_KNAME>,<_TYPE>,<_LAB1>,<_LAB2>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	PRG/CON	<_KNAME>	STRING [140]	Konturnamn eller underprogramnamn, måste inte programmeras vid _TYPE = 2
2		<_TYPE>	INT	Bestämning av konturinmatning
				0 = Underprogram
				1 = Konturnamn
				2 = Labels
				3 = Labels i underprogram
3	LAB1	<_LAB1>	STRING[32]	Label 1, konturbörjan
4	LAB2	<_LAB2>	STRING[32]	Label 2, konturslut

20.1.13 CYCLE63 - fräsa konturficka

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
1	PRG	< _PRG>	STRING [100]	Namn urfräsningsprogram		
2		< _VARI>	INT	Bearbetningstyp		
				ENTAL:	Teknologisk bearbetning	
					1 =	grovbearbetning
					3 =	finbearbetning av botten
					4 =	finbearbetning av kant
					5 =	fasning
				TIOTAL:	Ansättningstyp	
					0 =	Nermatning koncentrisk
					1 =	Nermatning helikal
					2 =	Nermatning pendlande
				HUNDRATAL:	reserverat	
				TUSENTAL:	Lyftningsmode	
					0 =	lyfta till återgångsplanet
					1 =	lyfta på referenspunkt + säkerhetsavstånd
TIOTUSENTAL:	Startpunkt vid grovbearbetning och finbearbetning botten					
	0 =	auto				
	1 =	manuell				
3	RP	< _RP>	REAL	Återgångsplan (abs)		
4	Z0	< _Z0>	REAL	Referenspunkt för verktygsaxeln (abs)		
5	SC	< _SC>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)		
6	Z1	< _Z1>	REAL	Slutdjup, (se < _AMODE> ENTAL)		
7	F	< _F>	REAL	Matning i planet grovbearbetning/finbearbetning		
8	FZ	< _FZ>	REAL	Ansättningsmatning djup		
9	DXY	< _DXY>	REAL	Ansättning plan - enhet, (se < _AMODE> TIOTAL)		
10	DZ	< _DZ>	REAL	Ansättning djup		
11	UXY	< _UXY>	REAL	Finbearbetningsmån i planet		
12	UZ	< _UZ>	REAL	Finbearbetningsmån djup		

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
13		<_CDIR>	INT	Fräsriktning	0 =	Medmatning
					1 =	Motmatning
14	XS	<_XS>	REAL	Startpunkt X, absolut		
15	YS	<_YS>	REAL	Startpunkt Y, absolut		
16	ER	<_ER>	REAL	Nermatning helikal: Radie		
17	EP	<_EP>	REAL	Nermatning helikal: Stigning		
18	EW	<_EW>	REAL	Nermatning pendlande: Maximal nermatningsvinkel		
19	FS	<_FS>	REAL	Fasbredd (ink) vid fasning		
20	ZFS	<_ZFS>	REAL	Nermatningsdjup verktygsspets vid fasning, (se <_AMODE> HUNDRA-TAL)		
21	TR	<_TR>	STRING[32]	Referensverktygsnamn vid restmaterialbearbetning		
22	DR	<_DR>	INT	Referensverktyg D-nummer vid restmaterialbearbetning		
23		<_UMODE>	INT	reserverat		
24		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)		
				ENTAL:	reserverat	
				TIOTAL:	reserverat	
				HUNDRATAL:	Val bearbetning/endast startpunktberäkning	
					0 =	normal bearbetning (ingen kompatibilitetsmode nödvändig)
					1 =	normal bearbetning
					2 =	reserverat
25		<_DMODE>	INT	Indikeringsmode		
				ENTAL:	Bearbetningsplan G17/G18/G19	
					0 =	kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
					1 =	G17 (endast aktiv i cykeln)
					2 =	G18 (endast aktiv i cykeln)
					3 =	G19 (endast aktiv i cykeln)
				TIOTAL:	reserverat	
				HUNDRATAL:	Teknologimodus	
					1 =	Ficka
					2 =	Tapp
				TUSENTAL:	Bearbeta restmaterial	
					0 =	nej
					1 =	ja
				TIOTUSENTAL:	Teknologiskalning inom cykelmaskerna (Sida 814)	
					0 =	inmatning: komplett
					1 =	inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
26		<_AMODE>	INT	Alternativmode
				ENTAL:
				Slutdjup (Z1)
				0 = absolut (kompatibilitetsmode)
				1 = inkrementell
				TIOTAL:
				Enhet för ansättning i planet (DXY)
				0 = mm
				1 = % av verktygsdiametern
				HUNDRATAL:
				Nermatningsdjup vid fasning (ZFS)
				0 = absolut
				1 = inkrementell
				TUSENTAL:

				reserverat

20.1.14 CYCLE64 - förborra konturficka

Syntax

```
CYCLE64 (<_PRG>, <_VARI>, <_RP>, <_Z0>, <_SC>, <_Z1>, <_F>, <_DXY>,
<_UXY>, <_UZ>, <_CDIR>, <_TR>, <_DR>, <_UMODE>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	PRG	<_PRG>	STRING [100]	Namn borr-/centrerprogram
2		<_VARI>	INT	Bearbetningstyp
				ENTAL:
				reserverat
				TIOTAL:
				reserverat
				HUNDRATAL:
				reserverat
				TUSENTAL:
				Lyftningsmode
				0 = lyfta till återgångsplanet
				1 = lyfta på referenspunkt + säkerhetsavstånd
3	RP	<_RP>	REAL	Återgångsplan (abs)
4	Z0	<_Z0>	REAL	Referenspunkt (abs)
5	SC	<_SC>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
6	Z1	<_Z1>	REAL	Borr-/centrerdjup, (se <_AMODE> ENTAL)
7	F	<_F>	REAL	Matning borrar/centrering
8	DXY	<_DXY>	REAL	Ansättning plan - enhet, (se <_AMODE> TIOTAL)
9	UXY	<_UXY>	REAL	Finbearbetningsmån i planet

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
10	UZ	< _UZ>	REAL	Finbearbetningsmån djup		
11		< _CDIR>	INT	Fräsriktning	0 =	Medmatning
					1 =	Motmatning
12	TR	< _TR>	STRING[20]	Referensverktygsnamn		
13	DR	< _DR>	INT	Referensverktyg D-nummer		
14		< _UMODE>	INT	reserverat		
15		< _GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)		
				ENTAL:	reserverat	
				TIOTAL:	reserverat	
				HUNDRATAL:	Val bearbetning/enda startpunktberäkning	
					0 =	normal bearbetning (ingen kompatibilitetsmode nödvändig)
					1 =	normal bearbetning
					2 =	reserverat
25		< _DMODE>	INT	Indikeringsmode		
				ENTAL:	Bearbetningsplan G17/G18/G19	
					0 =	kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
					1 =	G17 (endast aktiv i cykeln)
					2 =	G18 (endast aktiv i cykeln)
					3 =	G19 (endast aktiv i cykeln)
				TIOTAL:	Teknologimodus)	
					1 =	Förborring
					2 =	Centrering
26		< _AMODE>	INT	Alternativmode		
				ENTAL:	Borr-/centrerdjup Z1	
					0 =	absolut (kompatibilitetsmode)
					1 =	inkrementell
				TIOTAL:	Enhet för ansättning i planet (DXY)	
					0 =	mm
					1 =	% av verktygsdiametern

20.1.15 CYCLE70 - gängfräsning

Syntax

```
CYCLE70(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DIATH>, <_H1>, <_FAL>,
<_PIT>, <_NT>, <_MID>, <_FFR>, <_TYPTH>, <_PA>, <_PO>, <_NSP>,
<_VARI>, <_PITA>, <_PITM>, <_PTAB>, <_PTABA>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
1	RP	<_RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Gänglängd (abs, ink), se <_AMODE> Ta hänsyn till utgången vid hålets botten (minimum halv stigning)
5	Ø	<_DIATH>	REAL	Märkdiameter för gängen
6	H1	<_H1>	REAL	Gängdjup
7	U	<_FAL>	REAL	Finbearbetningsmån
8	P	<_PIT>	REAL	Gängstigning (val <_PITA>: mm, inch, MODUL, gängor/tum)
9	NT	<_NT>	INT	Antal tänder på skärplattan Verktöglängd alltid relaterad till den undre tanden!
10	DXY	<_MID>	REAL	Maximal ansättning per skär <_MID> > <_H1>: allt med ett skär
11	F	<_FFR>	REAL	Fräsmatning
12		<_TYPH>	INT	Gängtyp 0 = Invändig gänga 1 = Utvändig gänga
13	X0	<_PA>	REAL	Cirkelns medelpunkt, 1:a axeln (abs)
14	Y0	<_PO>	REAL	Cirkelns medelpunkt, 2:a axeln (abs)
15	αS	<_NSP>	REAL	Startvinkel (gंगा med flera ingångar)
16		<_VARI>	INT	Bearbetningstyp ENTAL: 1 = grovbearbetning 2 = finbearbetning TIOTAL: 1 = upifrån och ner 2 = nerifrån och upp HUNDRATAL: 0 = Höergंगा 1 = Vänstergंगा
17		<_PITA>	INT	Värdering av gängstigningen 0 = kompatibilitetsmode 1 = stigning i mm 2 = stigning i gängor per tum (TPI) 3 = stigning i inch 4 = stigning som MODUL
18		<_PITM>	STRING[15]	String som flagga för inmatning av gängstigning (endast för ytan)
19		<_PTAB>	STRING[20]	String för gängtabell ("", "ISO", "BSW", "BSP", "UNC") (endast för ytan)
20		<_PTABA>	STRING[20]	String för val i gängtabellen (t.ex. "M 10", "M 12", ...) (endast för ytan)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
21		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TIOTAL: reserverat
				HUNDRATAL: Bearbetning/startpunktberäkning
				0 = kompatibilitetsmode 1 = normal bearbetning
22		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln) 3 = G19 (endast aktiv i cykeln)
23		<_AMODE>	INT	Alternativmode
				ENTAL: Gänglängd (<_DP>)
				0 = absolut 1 = inkrementell

20.1.16 CYCLE72 - banfräsa

Syntax

```
CYCLE72 (<_KNAME>, <_RTP>, <_RFP>, <_SDIS>, <_DP>, <_MID>, <_FAL>,
<_FALD>, <_FFP1>, <_FFD>, <_VARI>, <_RL>, <_AS1>, <_LP1>, <_FF3>,
<_AS2>, <_LP2>, <_UMODE>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1		<_KNAME>	STRING [141]	Namn på kontur-underprogrammet
2	RP	<_RTP>	REAL	Återgångsplan (abs)
3	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
4	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
5	Z1	<_DP>	REAL	Slutpunkt, slutdjup (abs/ink), se <_AMODE>
6	DZ	<_MID>	REAL	Maximal djupansättning (ink, mata in utan förtecken)
7	UXY	<_FAL>	REAL	Finbearbetningsmån plan (ink), avmätt vid kantkonturen
8	UZ	<_FALD>	REAL	Finbearbetningsmån djup (ink), avmätt på botten mata in utan förtecken)
9	FX	<_FFP1>	REAL	Matning vid konturen

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
10	FZ	<_FFD>	REAL	Matning för djupansättning (eller ansättning i rymden)		
11		<_VARI>	INT	Bearbetningstyp		
				ENTAL:	Bearbetning	
					1 =	grobearbetning
					2 =	finbearbetning
					5 =	fasning
				TIOTAL:		
					0 =	mellanväg med G0
					1 =	mellanväg med G1
				HUNDRATAL:	Tillbakamatning vid konturslut	
					0 =	återgång på konturslut till referenspunkt
					1 =	Återgång på konturslut till referenspunkt + <_SDIS>
					2 =	Tillbakamatning vid konturslut med <_SDIS>
					3 =	ingen återgång på konturslut, nästa startpunkt uppsöks med konturmatning
				TUSENTAL:	reserverat	
				TIOTUSENTAL:	Bearbeta kontur	
					0 =	Bearbeta kontur fram
1 =	Bearbeta kontur bakåt Inskränkningar vid bakåt: • Max. 170 konturelement (inklusive faser eller avrundningar) • Endast värden i planet (X/Y) och F utvärderas					
12		<_RL>	INT	Bearbetningsriktning		
					40 =	koncentrisk till konturen (G40, fram- och bortkörning: linje eller lodrätt)
					41 =	vänster om konturen (G41, fram- och bortkörning: linje eller cirkel)
					42 =	höger om konturen (G42, fram- och bortkörning: linje eller cirkel)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
13		<_AS1>	INT	Kontur-framkörningsrörelse
				ENTAL:
				1 = linje
				2 = kvartscirkel
				3 = halvcirkel
				4 = lodrät fram- och bortkörning
				TIOTAL:
				0 = sista rörelse, i planet
				1 = sista rörelse, i rymden
14	L1	<_LP1>	REAL	Framkörningsväg, eller framkörningsradie (ink, mata in utan förtecken)
15	FZ	<_FF3>	REAL	Matning för mellansträckor (G94/G95 som för kontur)
16		<_AS2>	INT	Kontur-bortkörningsrörelse (inte vid lodrät fram-/bortkörning)
				ENTAL:
				1 = linje
				2 = kvartscirkel
				3 = halvcirkel
				TIOTAL:
				0 = sista rörelse, i planet
				1 = sista rörelse, i rymden
17	L2	<_LP2>	REAL	Bortkörningsväg, eller bortkörningsradie (ink, mata in utan förtecken)
18		<_UMODE>	INT	reserverat
19	FS	<_FS>	REAL	Fasbredd (ink)
20	ZFS	<_ZFS>	REAL	Nermatningsdjup (verktygsspets) vid fasning (abs/ink) se <_AMODE>
21		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TIOTAL: reserverat
				HUNDRATAL: Val bearbetning/endast startpunktberäkning
				0 = kompatibilitetsmode
				1 = normal bearbetning

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
22		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL:
				Matningstyp: G-grupp (G94/G95) för yt- och djupmatning
				0 = kompatibilitetsmode
23		<_AMODE>	INT	TOTAL:
				0 = Kompatibilitetsmode: Konturnamn står i <_KNAME>
				1 = konturnamn programmeras i CYCLE62 och överlämnas till _SC_CONT_NAME
				TUSENTAL:
				0 =
				1 =
				Alternativmode
				ENTAL:
				Slutpunkt Z1 (<_DP>)
				0 = abssolut (kompatibilitetsmode)
				1 = inkrementell
				TIOTAL:
				Enhet för ansättning i planet
				0 = mm (inch)
				1 = reserverat
				HUNDRATAL:
				Nermatningsdjup vid fasning (<_ZFS>)
				0 = absolut
				1 = inkrementell

Märk

Är följande överföringsparametrar programmerade indirekt (som parameter), återöversätts inte inmatningsmasken:

<_VARI>, <_RL>, <_AS1>, <_AS2>, <_UMODE>, <_GMODE>, <_DMODE>, <_AMODE>

20.1.17 CYCLE76 - fräsa fyrkanttapp

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<_RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Tappdjup (abs)
5		<_DPR>	REAL	Tappdjup (ink), relaterat till Z0 (mata in utan förtecken)
6	L	<_LENG>	REAL	Tapplängd, se <_GMODE> (matas in utan förtecken)
7	W	<_WID>	REAL	Tappbredd, se <_GMODE> (matas in utan förtecken)
8	R	<_CRAD>	REAL	Hörnradie för tappen (mata in utan förtecken)
9	X0	<_PA>	REAL	Referenspunkt tapp, 1:a axeln i planet (abs)
10	Y0	<_PO>	REAL	Referenspunkt tapp, 2:a axeln i planet (abs)
11	$\alpha 0$	<_STA>	REAL	Vridvinkel, vinkel mellan längsaxel (L) och 1:a axeln i planet
12	DZ	<_MID>	REAL	Maximal djupansättning (ink, mata in utan förtecken)
13	UXY	<_FAL>	REAL	Finbearbetningsmån plan (ink), avmätt vid kantkonturen
14	UZ	<_FALD>	REAL	Finbearbetningsmån djup (ink), avmätt på botten (mata in utan förtecken)
15	FX	<_FFP1>	REAL	Matning vid konturen
16	FZ	<_FFD>	REAL	Ansättningsmatning djup
17		<_CDIR>	INT	Fräsriktning (mata in utan förtecken)
				ENTAL:
				0 = Medmatning
				1 = Motmatning
18		<_VARI>	INT	Bearbetning
				ENTAL:
				1 = grovbearbetning
				2 = finbearbetning
				5 = fasning
19	L1	<_AP1>	REAL	Längd på råämnestappen
20	W1	<_AP2>	REAL	Bredd på råämnestappen
21	FS	<_FS>	REAL	Fasbredd (ink)
22	ZFS	<_ZFS>	REAL	Nermatningsdjup (verktygsspets) vid fasning (abs, ink), se <_AMODE>

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
23		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TIOTAL: reserverat
				HUNDRATAL: Val bearbetning eller endast startpunktberäkning
				0 = kompatibilitetsmode
				1 = normal bearbetning
				TUSENTAL: Måttställning av tappen vid mitt eller hörn
				0 = kompatibilitetsmode
				1 = måttställning via mitt
				2 = Måttställning hörnpunkt, tapp +L +W
				3 = Måttställning hörnpunkt, tapp -L +W
				4 = Måttställning hörnpunkt, tapp +L -W
				5 = Måttställning hörnpunkt, tapp -L -W
				TIOTUSENTAL: Komplettbearbetning eller efterbearbetning
24		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL: --- reserverat
				HUNDRATAL: --- reserverat
				TUSENTAL: --- reserverat
				TIOTUSENTAL: Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
25		<_AMODE>	INT	Alternativmode
				ENTAL:
				Slutdjup Z1 (DP)
				0 = kompatibilitet
				1 = inkrementell
				2 = absolut
				TIOTAL: reserverat
				HUNDRATAL:
				Nermatningsdjup vid fasning (ZFS)
				0 = absolut
				1 = inkrementell

20.1.18 CYCLE77 - fräsa cirkeltapp

Syntax

```
CYCLE77 (<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_DPR>, <_CDIAM>, <_PA>,
<_PO>, <_MID>, <_FAL>, <_FALD>, <_FFP1>, <_FFD>, <_CDIR>, <_VARI>,
<_AP1>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<_RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Tappdjup (abs)
5		<_DPR>	REAL	Tappdjup (ink), relaterat till Z0 (mata in utan förtecken)
6	Ø	<_CDIAM>	REAL	Diameter för tappen (mata in utan förtecken)
7	X0	<_PA>	REAL	Referenspunkt tapp, 1:a axeln i planet (abs)
8	Y0	<_PO>	REAL	Referenspunkt tapp, 2:a axeln i planet (abs)
9	DZ	<_MID>	REAL	Maximal djupansättning (ink, mata in utan förtecken)
10	UXY	<_FAL>	REAL	Finbearbetningsmån plan (ink), avmätt vid kantkonturen
11	UZ	<_FALD>	REAL	Finbearbetningsmån djup (ink), avmätt på botten (mata in utan förtecken)
12	FX	<_FFP1>	REAL	Matning vid konturen
13	FZ	<_FFD>	REAL	Ansättningsmatning djup
14		<_CDIR>	INT	Fräsriktning (mata in utan förtecken)
				ENTAL:
				0 = Medmatning
				1 = Motmatning

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
15		<_VARI>	INT	Bearbetningstyp
				ENTAL:
				Bearbetning
				1 = skrubbearbetning till arbetsmån för finbearbetning
				2 = finbearbetning (arbetsmån X/Y/ Z=0)
				5 = fasning
16	Ø1	<_AP1>	REAL	Rååmnestappens diameter
17	FS	<_FS>	REAL	Fasbredd (ink)
18	ZFS	<_ZFS>	REAL	Nermatningsdjup (verktygsspets) vid fasning (abs/ink) se <_AMODE>
19		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL:
				reserverat
				TIOTAL:
				reserverat
				HUNDRATAL:
				Val bearbetning/enda startpunktberäkning
				0 = kompatibilitetsmode
				1 = normal bearbetning
				TUSENTAL:
				reserverat
				TIOTUSENTAL:
				Komplettbearbetning/efterbearbetning
				0 = Kompatibilitetsmode (<_AP1> behandla som tidigare)
				1 = komplettbearbetning
				2 = efterbearbetning
20		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL:

				reserverat
				HUNDRATAL:

				reserverat
				TUSENTAL:

				reserverat
				TIOTUSENTAL:
				Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
21		<_AMODE>	INT	Alternativmode
				ENTAL:
				Slutdjup Z1 (DP)
				0 = absolut (kompatibilitetsmode)
				1 = inkrementell
				2 = absolut
				TIOTAL: reserverat
		<_AMODE>	INT	HUNDRATAL:
				Nermatningsdjup vid fasning (ZFS)
				0 = absolut
				1 = inkrementell

20.1.19 CYCLE78 - fräsa hålgänga

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<_RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Slutborrdjup (abs/ink), se <_AMODE>
5		<_ADPR>	REAL	Förborrningsdjup med reducerad bormatning (ink), med <_VARI> TIOTUSENTAL verksam
6	D	<_FDPR>	REAL	Maximal djupansättning (ink) $D \geq Z1 \Rightarrow$ en ansättning på slutborrdjup $D < Z1 \Rightarrow$ djupborrcykel med flera ansättningar och urspåning
7	ZR	<_LDPR>	REAL	Restborrdjup vid genomborrning (ink), med matning FR
8	Ø	<_DIAM>	REAL	Märkdiameter för gängen
9	P	<_PIT>	REAL	Gängstigning som siffervärde
10		<_PITA>	INT	Värdering av gängstigningen P
				1 = stigning i mm/varv
				2 = stigning i gängor/tum
				3 = stigning i inch/varv
				4 = stigning som MODUL

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
11	DF	<_DAM>	REAL	Belopp/procentsats för varje ytterligare ansättning (degression), se <_AMODE>
12	V1	<_MDEP>	REAL	Minimal ansättning (ink), endast verksam vid degression
13		<_VARI>	INT	Bearbetningstyp ENTAL: reserverat TIOTAL: Urspåning före gängfräsning 0 = ingen urspåning före gängfräsning (verkar endast på slutborrdjup) 1 = urspåning före gängfräsning (verkar endast på slutborrdjup) HUNDRATAL: Höger-/vänstergånga 0 = Höbergånga 1 = Vänstergånga TUSENTAL: Restborrdjup med borrar matning 0 = inget restborrdjup med borrar matning FR 1 = restborrdjup med borrar matning FR TIOTUSENTAL: Förborra med reducerad matning 0 = ingen förborring med minskad matning 1 = Förborra med reducerad matning förborringsdjup = 0,3 F1, när F1 < 0,15 mm/varv förborringsdjup = 0,1 mm/varv, när F1 ≥ 0,15 mm/varv
14		<_CDIR>	INT	Fräsriktning 0 = Medmatning 1 = Motmatning 4 = Motmatning + medmatning (kombination grovbearbetning + finbearbetning)
15	Z2	<_GE>	REAL	Återgångsvärde före gängfräsning (ink)
16	F1	<_FFD>	REAL	Borrmattning (mm/min resp. in/min eller mm/varv)
17	FR	<_FRDP>	REAL	Borrmattning för restborrdjup (mm/min eller mm/varv)
18	F2	<_FFR>	REAL	Matning för gängfräsning (mm/min eller mm/tand)
19	FS	<_FFP2>	REAL	Finbearbetningsmatning för <_CDIR> =4 (mm/min eller mm/tand)
20		<_FFA>	INT	Värdering matningar ENTAL: Borrmattning F1 TIOTAL: Borrmattning för restborrdjup FR HUNDRATAL: Matning för gängfräsning F2 TUSENTAL: Finbearbetningsmatning FS
21		<_PITM>	STRING[15]	String som flagga för inmatning av gängstigning (endast för ytan) ¹⁾
22		<_PTAB>	STRING[20]	String för gängtabell ("", "ISO", "BSW", "BSP", "UNC") (endast för ytan) ¹⁾
23		<_PTABA>	STRING[20]	String för val i gängtabellen (t.ex. "M 10", "M 12", ...) (endast för ytan) ¹⁾

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
24		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena), reserverat
25		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
26		<_AMODE>	INT	Alternativmode
				ENTAL:
				Borrdjup = slutborrdjup Z1 abs/ink
				0 = absolut
				1 = inkrementell
				TOTAL:
				Belopp/procentsats DF för varje ytterligare ansättning (degression)
				0 = Belopp
				1 = procentsats (0.001 till 100 %)

Märk

¹⁾ Parametrarna 21, 22 och 23 används endast vid val av gänga i gängtabellen för inmatningsmasken. En åtkomst på gängtabellerna via cykeldefinition vid cykelkörning är inte möjligt.

20.1.20 CYCLE79 - flerkant**Syntax**

```
CYCLE79(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_NUM>, <_SWL>, <_PA>,
<_PO>, <_STA>, <_RC>, <_AP1>, <_MIDA>, <_MID>, <_FAL>, <_FALD>,
<_FFP1>, <_CDIR>, <_VARI>, <_FS>, <_ZFS>, <_GMODE>, <_DMODE>,
<_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<_RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Flerkantdjup (abs/ink), se <_AMODE>

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
5	N	< _NUM>	INT	Antal kanter (1...n)		
6	SW/L	< _SWL>	REAL	Nyckelvidd eller kantlängd (allt efter < _VARI>) ("SW" vid nyckelvidd, "L" vid kantlängd) Nyckelvidd endast vid jämnt antal kanter och enkant		
7	X0	< _PA>	REAL	Referenspunkt för tapp, 1:a axeln (abs)		
8	Y0	< _PO>	REAL	Referenspunkt för tapp, 2:a axeln (abs)		
9	α0	< _STA>	REAL	Vridvinkel kantmitt mot 1:a axeln (X-axel)		
10	R1/FS1	< _RC>	REAL	Hörnrundning vid < _NUM> > 2 (radie/fas, se < _AMODE>) (ink, mata in utan förtecken) ("R1" vid radie, "FS1" vid fas)		
11	Ø	< _AP1>	REAL	Rådiameter för tappen		
12	DXY	< _MIDA>	REAL	Maximal ansättningsbredd (enhet, se < _AMODE>)		
13	DZ	< _MID>	REAL	Maximal djupansättning		
14	UXY	< _FAL>	REAL	Finbearbetningsmån i planet		
15	UZ	< _FALD>	REAL	Finbearbetningsmån djup		
16	F	< _FFP1>	REAL	Arbetsmatning		
17		< _CDIR>	INT	Fräsriktning	0 =	Medmatning
					1 =	Motmatning
18		< _VARI>	INT	Bearbetningstyp		
				ENTAL:	Bearbetning	
					1 =	grovbearbetning
					2 =	finbearbetning
					3 =	finbearbetning av kant
					5 =	fasning
				TIOTAL:	Nyckelvidd eller kantlängd	
					0 =	Nyckelvidd
					1 =	Kantlängd
19	FS	< _FS>	REAL	Fasbredd (ink)		
20	ZFS	< _ZFS>	REAL	Nermatningsdjup (verktygsspets) vid fasning (abs/ink) se < _AMODE>		
21		< _GMODE>	INT	Geometrimode (värdering av de programmerade geometrivrärderna)		
				ENTAL:	reserverat	
				TIOTAL:	reserverat	
				HUNDRATAL:	Val bearbetning eller endast startpunkt-beräkning	
					1 =	normal bearbetning

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
22		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/18/19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL: --- reserverat
				HUNDRATAL: --- reserverat
				TUSENTAL: --- reserverat
23		<_AMODE>	INT	Alternativmode
				ENTAL:
				Slutdjup (<_DP>)
				0 = absolut
				1 = inkrementell
				TIOTAL:
				Enhet för ansättning i planet (<_MIDA>)
				0 = mm
				1 = % av verktygsdiametern
				HUNDRATAL:
				Nermatningsdjup vid fasning (<_ZFS>)
				0 = absolut
				1 = inkrementell
				TUSENTAL:
				Hörrundning (<_RC>)
				0 = Radie
				1 = Fas

20.1.21 CYCLE81 - borra, centrera

Syntax

```
CYCLE81(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<RTP>	REAL	Återgångsplan (abs)
2	Z0	<RFP>	REAL	Referenspunkt (abs)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1/Ø	<DP>	REAL	Borrdjup (abs) / diameter för centreringen (abs), se <_GMODE>
5	Z1	<DPR>	REAL	Borrdjup (ink)
6	DT	<DTB>	REAL	Fördröjningstid på slutborrdjup, se <_AMODE>
7		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TOTAL: Centrerings relaterad till djupet / diametern
				0 = Kompatibilitet, djup
8		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
9		<_AMODE>	INT	Alternativmode
				ENTAL: Borrdjup Z1 (abs/ink)
				0 = kompatibilitet, från programmering DP/DPR
				1 = inkrementell
				2 = absolut
				TOTAL: Fördröjningstid på slutborrdjup DT i sekunder/varv
				0 = kompatibilitet, av förtecken från DTB (> 0 sekunder eller < 0 varv)
				1 = i sekunder:
				2 = i varv

20.1.22 CYCLE82 - borra, plansänka

Syntax

```
CYCLE82 (<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <_GMODE>,
<_DMODE>, <_AMODE>, <_VARI>, <S_ZA>, <S_FA>, <S_ZD>, <S_FD>)
```


Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<RTP>	REAL	Återgångsplan (abs)
2	Z0	<RFP>	REAL	Referenspunkt (abs)
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<DP>	REAL	Borrdjup (abs), se <_AMODE>
5	Z1	<DPR>	REAL	Borrdjup (ink), se <_AMODE>
6	DT	<DTB>	REAL	Fördröjningstid på slutborrdjup, se <_AMODE>
7		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TIOTAL: Borrdjup relaterat till spets/skaft
				0 = Kompatibilitet, spets
				1 = Skaft
8		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL: reserverat
				HUNDRATAL: reserverat
				TUSENTAL: reserverat
				TIOTUSENTAL: Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
9		<_AMODE>	INT	Alternativmode
				ENTAL:
				Borrdjup Z1 (abs/ink)
				0 = Kompatibilitet, från programmering DP/DPR
				1 = inkrementell
				2 = absolut
				TIOTAL:
				Fördröjningstid DT på slutborrdjup i sekunder/varv
				0 = kompatibilitet, av förtecken från DT (> 0 sekunder eller < 0 varv)
				1 = i sekunder:
				2 = i varv
				HUNDRATAL:
				Förborrdjup ZA abs/ink
				0 = inkrementell
				1 = absolut
				TUSENTAL:
				Värdering förborrmätning
				0 = i % av borrmätning
				1 = F/min
				2 = F/varv
				TIOTUSENTAL:
				Restborrdjup ZD abs/ink
				0 = inkrementell
				1 = absolut
				HUNDRATUSENTAL:
				Värdering restborrmätning
				0 = i % av borrmätning
				1 = F/min
				2 = F/varv
10		<_VARI>	INT	Bearbetningstyp förborra/genomborra
				ENTAL:
				reserverat
				TIOTAL:
				reserverat
				HUNDRATAL:
				reserverat
				TUSENTAL:
				Genomgående borring
				0 = Genomgående borring "nej"
				1 = Genomgående borring "ja"
				TIOTUSENTAL:
				Förborring
				0 = Förborra "nej"
				1 = Förborra "ja"
11	ZA	<S_ZA>	REAL	Förborrdjup inkrementellt relaterat till referenspunkt eller absolut (se <_AMODE> HUNDRATAL)
12	FA	<S_FA>	REAL	Förborrmätning som värde eller i % (i förbindelse med <_AMODE> TUSENTAL)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
13	ZD	<S_ZD>	REAL	Restborrdjup inkrementellt relaterat till slutborrdjup eller absolut (i förbindelse med <_AMODE> TIOTUSENTAL)
14	FD	<S_FD>	REAL	Restborrmatning som värde eller i % (i förbindelse med <_AMODE> HUNDRATUSENTAL)

20.1.23 CYCLE83 - djuphålsborra

Syntax

```
CYCLE83 (<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <FDEP>, <FDPR>, <_DAM>,
<DTB>, <DTS>, <FRF>, <VARI>, <_AXN>, <_MDEP>, <_VRT>, <_DTD>,
<_DIS1>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
1	RP	<RTP>	REAL	Återgångsplan (abs)		
2	Z0	<RFP>	REAL	Referenspunkt (abs)		
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)		
4	Z1	<DP>	REAL	Slutborrdjup (abs), se <_AMODE>		
5	Z1	<DPR>	REAL	Slutborrdjup (ink), se <_AMODE>		
6	D	<FDEP>	REAL	1. Borrdjup (abs), se <_AMODE>		
7	D	<FDPR>	REAL	1. Borrdjup (ink), se <_AMODE>		
8	DF	<_DAM>	REAL	Degressionsbelopp/procentsats för varje ytterligare ansättning, se <_AMODE>		
9	DTB	<DTB>	REAL	Fördröjningstid på borrdjup, se <_AMODE>		
10	DTS	<DTS>	REAL	Fördröjningstid i begynnelsepunkten (endast vid urspåning), se <_AMODE>		
11	FD1	<FRF>	REAL	Procentsats för matningen vid första ansättningen, se <_AMODE>		
12		<VARI>	INT	Bearbetningstyp		
				ENTAL:		Spånbrytning / urspåning
				0 =	Spånbrytning	
				1 =	Urspåning	
13		<_AXN>	INT	Verktygsaxel		
				0 =	3:e geometriaxel	
				1 =	1:a geometriaxel	
				2 =	2:a geometriaxel	
				> 2	3:e geometriaxel	
14	V1	<_MDEP>	REAL	Minimal ansättning (endast vid procentsats för degression)		

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
15	V2	<_VRT>	REAL	Återgångsvärde efter varje bearbetning (endast vid spånbrytning)
				> 0 variabelt återgångsvärde
				0 = standardvärde 1 mm
16	DT	<_DTD>	REAL	Fördröjningstid på slutborrdjup, se <_AMODE>
17	V3	<_DIS1>	REAL	Förstoppavstånd (endast vid urspåning), se <_AMODE>
18		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TIOTAL: Borrdjup relaterat till spets/skaft
				0 = Spets
				1 = Skaft
19		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL: --- reserverat
				HUNDRATAL: --- reserverat
				TUSENTAL: --- reserverat
				TIOTUSENTAL: Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
20		<_AMODE>	INT	Alternativmode
				ENTAL:
				Borrdjup = slutborrdjup Z1 (abs/ink)
				0 = kompatibilitet, från programmering <DP>/<DPR>
				1 = inkrementell
				2 = absolut
				TIOTAL:
				Fördröjningstid på borrdjup DTB i sekunder/varv
				0 = Kompatibilitet, av förtecken från DTB (> 0 sekunder eller < 0 varv)
				1 = i sekunder:
				2 = i varv
				HUNDRATAL:
				Fördröjningstid i begynnelsepunkten till DTS i sekunder/varv
				0 = kompatibilitet av förtecken från DTS (> 0 sekunder eller < 0 varv)
				1 = i sekunder:
				2 = i varv
				TUSENTAL:
				Fördröjningstid på slutborrdjup DTD i sekunder/varv
				0 = Kompatibilitet av förtecken från DTD (> 0 sekunder eller < 0 varv)
				1 = i sekunder:
				2 = i varv
				TIOTUSENTAL:
				1. Borrdjup D (abs/ink)
				0 = kompatibilitet, från programmering <FDEPF>/<DPR>
				1 = inkrementell
				2 = absolut
				HUNDRATUSENTAL:
				Degressionsbelopp/procentsats <_DAM> för varje ytterligare ansättning
				0 = Kompatibilitet, av förtecken från <_DAM> (> 0 degressionsbelopp eller < 0 faktor 0,001 till 1,0)
				1 = Degressionsbelopp
				2 = Procentsats (0,001 till 100 %)
				ENMILJONTAL:
				Förstoppavstånd V3 automatiskt/manuellt
				0 = kompatibilitet av förtecken från <_DIS1> (= 0 automatiskt eller > 0 manuellt)
				1 = automatiskt (beräknas i cykeln)
				2 = manuellt (programmerat värde)
				TIOMILJONTAL:
				Matningsfaktor för första ansättning <FRF> som faktor/procentsats

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
				0 = kompatibilitet, som faktor (0,001 till 1,0, FRF = 0 betyder 100%)
				1 = Procentsats (0,001 till 999,999 %)

20.1.24 CYCLE84 - Gängtappning utan flytande gänghållare

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
1	RP	<RTP>	REAL	Återgångsplan (abs)
2	Z0	<RFP>	REAL	Referenspunkt (abs)
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<DP>	REAL	Borrdjup = slutborrdjup (abs), se <_AMODE>
5	Z1	<DPR>	REAL	Borrdjup = slutborrdjup (ink), se <_AMODE>
6	DT	<DTB>	REAL	Fördröjningstid på borrdjup i sekunder
7	SDE	<SDAC>	INT	Rotationsriktning efter cykelslut
8		<MPIT>	REAL	Gängstorlek endast för "ISO metrisk" (stigning beräknas internt under körtiden)
9	P	<PIT>	REAL	Gängstigning som värde, måttenhet se <_PITA>
10	αS ¹⁾	<POSS>	REAL	Spindelposition för orienterat spindelstopp
11	S	<SST>	REAL	Spindelvarvtal för gängtappning
12	SR	<SST1>	REAL	Spindelvarvtal för återgång
13		<_AXN>	INT	Borraxel
				0 = 3:e geometriaxel
				1 = 1:a geometriaxel
				2 = 2:a geometriaxel
				≥ 3 = 3:e geometriaxel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
14		<_PITA>	INT	Måttenhet för gängstignigen (urvärdering <PIT> och <MPIT>)
				0 = stigning i mm - Utvärdering <MPIT>/<PIT>
				1 = stigning i mm - Utvärdering <PIT>
				2 = stigning i TPI - Utvärdering <PIT> (antal gängor per inch)
				3 = stigning i inch - Utvärdering <PIT>
				4 = MODUL - Utvärdering <PIT>
15		<_TECHNO>	INT	Teknologi ¹⁾
				ENTAL: Precisionsstoppbeteende
				0 = precisionsstoppbeteende som före cykelanrop aktivt
				1 = precisionsstopp G601
				2 = precisionsstopp G602
				3 = precisionsstopp G603
				TIOTAL: Förstyrning
				0 = med/utan förstyrning som före cykelanrop aktiv
				1 = med förstyrning FFWON
				2 = utan förstyrning FFWOF
				HUNDRATAL: Acceleration
				0 = SOFT/BRISK/DRIVE som före cykelanrop aktiv
				1 = med jerkbegränsning SOFT
				2 = utan jerkbegränsning BRISK
				3 = reducerad acceleration DRIVE
				TUSENTAL: MCALL spindeldrift
				0 = vid MCALL åter aktivera spindeldrift
				1 = vid MCALL förbli i lägesreglering
16		<_VARI>	INT	Bearbetningstyp
				ENTAL: 0 = 1 skär
				1 = spånbrytning (djuphålgångtappning)
				2 = urspåning (djuphålgångtappning)
				TUSENTAL: ISO/SIEMENS mode för inmatningsmask inte relevant
				0 = anrop från ISO-kompatibilitet
				1 = anrop från SIEMENS-Kontext
17	D	<_DAM>	REAL	Maximal djupansättning (endast vid urspåning/spånbrytning)
18	V2	<_VRT>	REAL	Återgångsvärde efter varje bearbetning (endast vid spånbrytning), se <_AMODE>

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
19		<_PITM>	STRING[15]	String som flagga för gängstigningsinmatning ²⁾
20		<_PTAB>	STRING[5]	String för gängtabell ("", "ISO", "BSW", "BSP", "UNC") ²⁾
21		<_PTABA>	STRING[20]	String för val i gängtabellen (t. ex. "M 10", "M 12", ...) ²⁾
22		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TIOTAL: reserverat
23		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verk- samma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL: reserverat
				HUNDRATAL: reserverat
				TUSENTAL: Kompatibilitetsmode (endast för inmatnings- mask till återöversättning), när MD 52216 Bit0 = 1 ¹⁾
				0 = teknologiparametrar visas (kompatibili- tet): TECHNO-parametrar verkar
				1 = teknologiparametrar visas inte: Teknolo- gi "som programmerad före cykelanrop" verkar
				TIOTUSENTAL: Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
24		<_AMODE>	INT	Alternativmode		
				ENTAL:	Borrdjup = slutborrdjup Z1 (abs/ink)	
					0 =	kompatibilitet, från programmering <DP>/<DPR>
					1 =	inkrementell
					2 =	absolut
				TIOTAL:	reserverat	
				HUNDRATAL:	reserverat	
				TUSENTAL:	Gångvridningsriktning höger/vänster	
					0 =	kompatibilitet, av förtecken PIT/MPIT
					1 =	höger
					2 =	vänster
				TIOTUSENTAL:	reserverat	
				HUNDRATUSENTAL:	reserverat	
				ENMILJONTAL:	Återgångsvärde efter varje bearbetning V2 manuellt/automatiskt	
					0 =	kompatibilitet, från programmering <_VRT> (> 0 variabelt värde eller ≤ 0 standardvärde 1 mm/0,0394 inch)
1 =	automatiskt (standardvärde 1 mm/ 0,0394 inch)					
2 =	manuellt (som programmerat under V2)					

¹⁾ Fälten Teknologi kan vara gömda beroende på settingdatum SD52216 \$MCS_FUNCTION_MASK_DRILL

²⁾ Parametrarna 19, 20 och 21 används endast vid val av gänga i gängtabellen för inmatningsmasken. En åtkomst på gängtabellerna via cykeldefinition vid cykelkörtid är inte möjligt.

20.1.25 CYCLE85 - brotscha

Syntax

```
CYCLE85(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <FFR>, <RFF>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<RTP>	REAL	Återgångsplan (abs)
2	Z0	<RFP>	REAL	Referenspunkt (abs)
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<DP>	REAL	Borrdjup (abs), se <_AMODE>

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
5	Z1	<DPR>	REAL	Borrdjup (ink), se <_AMODE>
6	DT	<DTB>	REAL	Fördröjningstid på slutborrdjup, se <_AMODE>
7	F	<FFR>	REAL	Matning
8	FR	<RFF>	REAL	Matning vid återgång
9		<_GMODE>	INT	reserverat
10		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
11		<_AMODE>	INT	Alternativmode (borra)
				ENTAL:
				Borrdjup Z1 (abs/ink)
				0 = kompatibilitet, från programmering DP/DPR
				1 = inkrementell
				2 = absolut
				TOTAL:
				Fördröjningstid DT på slutborrdjup i sekunder/varv
				0 = kompatibilitet, av förtecken från DT (> 0 sekunder eller < 0 varv)
				1 = i sekunder:
				2 = i varv

20.1.26 CYCLE86 - ursvarvning

Syntax

```
CYCLE86(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDIR>, <RPA>,
<RPO>, <RPAP>, <POSS>, <_GMODE>, <_DMODE>, <_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<RTP>	REAL	Återgångsplan (abs)
2	Z0	<RFP>	REAL	Referenspunkt (abs)
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<DP>	REAL	Borrdjup (abs), se <_AMODE>

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
5	Z1	<DPR>	REAL	Borrdjup (ink), se <_AMODE>		
6	DT	<DTB>	REAL	Fördröjningstid på slutborrdjup, se <_AMODE>		
7	DIR	<SDIR>	INT	Spindelns rotationsriktning	3 =	M3
					4 =	M4
8	DX	<RPA>	REAL	Lyftningsbelopp i X-riktning		
9	DY	<RPO>	REAL	Lyftningsbelopp i Y-riktning		
10	DZ	<RPAP>	REAL	Lyftningsbelopp i Z-riktning		
11	SPOS	<POSS>	REAL	Spindelposition för lyftningen (för orienterat spindelstopp, i grader)		
12		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)		
				ENTAL:	Lyftningsmode	
				0 =	lyfta, kompatibilitet	
				1 =	ej lyftning	
13		<_DMODE>	INT	Indikeringsmode		
				ENTAL:	Bearbetningsplan G17/G18/G19	
				0 =	kompatibilitet, det före cykelanrop verksamma planet förblir aktivt	
				1 =	G17 (endast aktiv i cykeln)	
				2 =	G18 (endast aktiv i cykeln)	
				3 =	G19 (endast aktiv i cykeln)	
14		<_AMODE>	INT	Alternativmode		
				ENTAL:	Borrdjup Z1 (abs/ink)	
				0 =	kompatibilitet, från programmering <DP>/<DPR>	
				1 =	inkrementell	
				2 =	absolut	
				TIOTAL:	Fördröjningstid på slutborrdjup DT i sekunder/varv	
				0 =	kompatibilitet, av förtecken från DT (> 0 sekunder eller < 0 varv)	
				1 =	i sekunder:	
2 =	i varv					

20.1.27 CYCLE92 - avstick

Syntax

```
CYCLE92 (<_SPD>, <_SPL>, <_DIAG1>, <_DIAG2>, <_RC>, <_SDIS>, <_SV1>,
<_SV2>, <_SDAC>, <_FF1>, <_FF2>, <_SS2>, <_DIAGM>, <_VARI>, <_DN>,
<_DMODE>, <_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	X0	<_SPD>	REAL	Referenspunkt (abs, alltid diameter)
2	Y0	<_SPL>	REAL	Referenspunkt (abs)
3	X1	<_DIAG1>	REAL	Djup för varvtalsreducering, se <_AMODE> (ENTAL)
4	X2	<_DIAG2>	REAL	Slutdjup, se <_AMODE> (TIOTAL)
5	R/FS	<_RC>	REAL	Rundningsradie eller fasbredd, se <_AMODE> (TUSENTAL)
6	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
7	S	<_SV1>	REAL	Konstant spindelvarvtal, se <_AMODE> (TIOTUSENTAL)
	V			Konstant skärhastighet
8	SV	<_SV2>	REAL	Maximalvarvtal vid konstant skärhastighet
9	DIR	<_SDAC>	INT	Spindelns rotationsriktning
				3 = för M3
				4 = för M4
10	F	<_FF1>	REAL	Matning till djup för varvtalsreducering
11	FR	<_FF2>	REAL	Reducerad matning till slutdjup
12	SR	<_SS2>	REAL	Reducerat varvtal till slutdjup
13	XM	<_DIAGM>	REAL	Djup detaljfångare köra ut (abs, alltid diameter)
14		<_VARI>	INT	Bearbetningstyp
				ENTAL:
				Återgång
				0 = Återgång till <_SPD> + <_SDIS>
				1 = ingen återgång till slutet
				TIOTAL:
				Detaljfångare
				0 = nej, inte utföra något M-kommando
				1 = ja, anrop av CUST_TECHCYC(101)- köra ut låda, CUST_TECHCYC(102)- stänga låda
15		<_DN>	INT	D-nummer för 2:a skäret, om inte programmerat ⇒ D+1
20		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
21		<_AMODE>	INT	Alternativmode
				ENTAL:
				Djup för varvtalsreducering (<_DIAG1>)
				0 = absolut, värde för planaxeln i diametern
				1 = inkrementell, värde för planaxeln i radien
				TIOTAL:
				Slutdjup (<_DIAG2>)
				0 = absolut, värde för planaxeln i diametern
				1 = inkrementell, värde för planaxeln i radien
				HUNDRATAL:
				reserverat
				TUSENTAL:
				Radie/fas (<_RC>)
				0 = Radie
				1 = Fas
				TIOTUSENTAL:
				Spindelvarvtal/skärhastighet (<_SV1>)
				0 = konstant spindelvarvtal
				1 = konstant skärhastighet

20.1.28 CYCLE95 - konturavspåna

Syntax

CYCLE95 (<NPP>, <MID>, <FALZ>, <FALX>, <FAL>, <FF1>, <FF2>, <FF3>, <_VARI>, <DT>, <DAM>, <_VRT>, <_GMODE>, <_DMODE>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	CON	<NPP>	STRING [140]	Konturnamn
2	D	<MID>	REAL	Maximal djupansättning vid grovbearbetning, se <_GMODE>
3	UZ	<FALZ>	REAL	Finbearbetningsmån i Z
4	UX	<FALX>	REAL	Finbearbetningsmån i X
5	U	<FAL>	REAL	Finbearbetningsmån konturparallellt (verkar i båda axlarna)
6	F	<FF1>	REAL	Matning för grovbearbetning
7	FY	<FF2>	REAL	Nermatningsmatning baksnitt
8	FS	<FF3>	REAL	Finbearbetningsmatning

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
9		<_VARI>	INT	Bearbetningstyp
				ENTAL och TIO TAL:
				1 = grovbearbetning, längs, yttre
				2 = grovbearbetning, plan, yttre
				3 = grovbearbetning, längs, inre
				4 = grovbearbetning, plan, inre
				5 = finbearbetning, längs, yttre
				6 = finbearbetning, plan, yttre
				7 = finbearbetning, längs, inre
				8 = finbearbetning, plan, inre
				9 = komplettbearbetning, längs, yttre
				10 = komplettbearbetning, plan, yttre
				11 = komplettbearbetning, längs, inre
				12 = komplettbearbetning, plan, inre
				HUNDRATAL:
				0 = med efterdragning vid konturen, utan resthörn
				1 = utan efterdragning vid konturen
				2 = efterdragning endast till den föregående skärningspunkten, det kan uppstå resthörn
10	DT	<DT>	REAL	Fördröjningstid vid matningsstopp
11	DI	<DAM>	REAL	Avstånd för matningsstopp
12	VRT	<_VRT>	REAL	Lyftväg från konturen
				0 = internt används en lyftväg på 1 mm oberoende av måttsystemet inch/metriskt
				> 0 = lyftningsväg
13		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL:
				Värderint av ansättningsdjupet
				0 = ansättningsdjup tas med i beräkningen enligt G-grupp DIAMON/DIAMOF
				1 = ansättningsdjup verkar som radiavärde (oberoende av DIAMON/DIAMOF)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
14		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TUSENTAL:
				0 = kompatibilitetsmode: Konturnamn står i NPP
				1 = konturnamn programmeras i CYCLE62 och överlämnas till _SC_CONT_NAME

20.1.29 CYCLE98 - kopplade gängor

Syntax

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>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	Z0	<_PO1>	REAL	Referenspunkt i Z (abs)
2	X0	<_DM1>	REAL	Referenspunkt i X (abs), i diameter
3	Z1	<_PO2>	REAL	Mellangepunkt 1 i Z (abs/ink), se <_AMODE> (ENTAL)
4	X1	<_DM2>	REAL	Mellangepunkt 1 i X (abs/ink), se <_AMODE> (TIOTAL) eller
	X1α			Gänglutning 1 (-90° till 90°) abs alltid diameter, ink alltid radie
5	Z2	<_PO3>	REAL	Mellangepunkt 2 i Z (abs/ink), se <_AMODE> (HUNDRATAL)
6	X2	<_DM3>	REAL	Mellangepunkt 2 i X (abs/ink), se <_AMODE> (TUSENTAL) eller
	X2α			Gänglutning 2 (-90° till 90°) abs alltid diameter, ink alltid radie
7	Z3	<_PO4>	REAL	Ändpunkt i Z (abs/ink), se <_AMODE> (TIOTUSENTAL)
8	X3	<_DM4>	REAL	Ändpunkt i X (abs/ink), se <_AMODE> (HUNDRATUSENTAL) eller
	X3α			Gänglutning 3 (-90° till 90°) abs alltid diameter, ink alltid radie
9	LW	<APP>	REAL	Gänginkörning (ink, mata in utan förtecken)

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
10	LR	<ROP>	REAL	Gångutgång (ink, mata in utan förtecken)
11	H1	<TDEP>	REAL	Gängdjup (ink, mata in utan förtecken)
12	U	<FAL>	REAL	Finbearbetningsmått i X och Z
13	DP	<_IANG>	REAL	Ansättningslutning som avstånd eller vinkel, se <_AMODE> (MILJONTAL)
	αP			Ansättningslutningen verkar enligt inställningen parameter <_VARI> (HUNDRATAL).
				Definition för <_VARI_HUNDRTER = 0 - kompatibilitetsmode:
				> 0 = flankansättning vid en flank
				0 = ansättning lodrät i gängen
				< 0 = flankansättning med alternerande flanker
				Definition för _VARI_HUNDRTER<>0:
				> 0 = ansättning vid den positiva flanken
				0 = ansättning i mitten
				< 0 = ansättning vid den negativa flanken
14	α0	<NSP>	REAL	Startvinkelförskjutning för det 1:a gängvarvet
15		<NRC>	INT	Antal grovbearbetningsskär, se <_VARI> (TIOTUSENTAL)
16	NN	<NID>	INT	Antal rensningspassager
17	P0	<_PP1>	REAL	Gängstigning 1:a gängavsnittet, se <_PITA>
18	P1	<_PP2>	REAL	Gängstigning 2:a gängavsnittet, se <_PITA>
19	P2	<_PP3>	REAL	Gängstigning 3:e gängavsnittet, se <_PITA>

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
20		<_VARI>	INT	Bearbetning
				ENTAL:
				Teknologi
				1 = yttergånga med linjär ansättning
				2 = yttergånga med linjär ansättning
				3 = yttergånga med degressiv ansättning, spånarea förblir konstant
				4 = Innergånga med degressiv ansättning, spånarea förblir konstant
				TIOTAL:
				reserverat
				HUNDRATAL:
				Ansättningstyp
				0 = kompatibilitetsmode för <_IANG>
				1 = ensidig ansättning
				2 = växlande ansättning
21	N	<_NUMTH>	INT	TUSENTAL:
				reserverat
				TIOTUSENTAL:
				Alternativ djupansättning
				0 = kompatibilitet, föreskrivet antal grovbearbetningsskär (<_NRC>)
				1 = föreskrivet värde för 1:a ansättningen (<_MID>)
				HUNDRATUSENTAL:
				Bearbetningstyp
				0 = kompatibilitet (grovbearbetning och finbearbetning)
				1 = grovbearbetning
				2 = finbearbetning
				3 = grovbearbetning och finbearbetning
				ENMILJONTAL:
				Bearbetningsordningsföljd vid gängor med flera ingångar
				0 = ordningsföljd för gängorna stigande
				1 = ordningsföljd för gängorna omvänd
22		<_VRT>	REAL	Återgångsavstånd (ink)
23	D1	<_MID>	REAL	0 = internt används en lyftväg på 1 mm oberoende av måttssystemet inch/metriskt
				> 0 = lyftningsväg
24	DA	<_GDEP>	REAL	Första ansättning, se <_VARI> (TIOTUSENTAL)
25		<_IFLANK>	REAL	Gängväxlingsdjup (verkar endast vid "flera ingångar")
				0 = inte ta hänsyn till något gängväxlingsdjup
				> 0 = ta hänsyn till gängväxlingsdjup
26		<_IFLANK>	REAL	Ansättningslutning som bredd (endast för ytan)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
26		<_PITA>	INT	Värdering av gängstigningen		
					0 =	kompatibilitetsmode för gängstigning, utvärdering <_PP1> till <_PP3> som tidigare enligt aktivt system metriskt/inch
					1 =	stigning i mm
					2 =	stigning i TPI (gängvarv per inch)
					3 =	stigning i inch
					4 =	MODUL
27		<_PITM1>	STRING[15]	String som flagga för inmatning av gängstigning (endast för ytan)		
28		<_PITM2>	STRING[15]	String som flagga för inmatning av gängstigning (endast för ytan)		
29		<_PITM3>	STRING[15]	String som flagga för inmatning av gängstigning (endast för ytan)		
30		<_DMODE>	INT	Indikeringsmode		
				ENTAL:	Bearbetningsplan G17/G18/G19	
					0 =	kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
					1 =	G17 (endast aktiv i cykeln)
					2 =	G18 (endast aktiv i cykeln)
					3 =	G19 (endast aktiv i cykeln)
				TIOTAL:	---	reserverat
				HUNDRATAL:	---	reserverat
				TUSENTAL:	---	reserverat
				TIOTUSENTAL:	Teknologiskalning inom cykelmaskerna (Sida 814)	
0 =	inmatning: komplett					
1 =	inmatning: enkel					

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
31		<_AMODE>	INT	Alternativmode
				ENTAL:
				1:a Mellanpunkt i Z (Z1)
				0 = absolut
				1 = inkrementell
				TIOTAL:
				1:a Mellanpunkt i X (X1)
				0 = absolut
				1 = inkrementell
				2 = α
				HUNDRATAL:
				2:a Mellanpunkt i Z (Z2)
				0 = absolut
				1 = inkrementell
				TUSENTAL:
				2:a Mellanpunkt i X (X2)
				0 = absolut
				1 = inkrementell
				2 = α
				TIOTUSENTAL:
				Ändpunkt i Z (Z3)
				0 = absolut
				1 = inkrementell
				HUNDRATUSENTAL:
				Ändpunkt i X (X3)
				0 = absolut
				1 = inkrementell
				2 = α
				ENMILJONTAL:
				Val ansättningslutning som vinkel eller bredd
				0 = ansättningsvinkel <_IANG>
				1 = ansättningslutning <_IFLANK>
				TIOMILJONTAL:
				en ingång/flera ingångar
				0 = kompatibilitetsmode (startvinkel <_NSP> utvärderas)
				1 = en ingång (med startvinkelförskjutning <_NSP>)
				2 = med flera ingångar

20.1.30 CYCLE99 - gängsvarvning

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse		
1	Z0	<_SPL>	REAL	Referenspunkt (abs)		
2	X0	<_SPD>	REAL	Referenspunkt (abs, alltid diameter)		
3	Z1	<_FPL>	REAL	Ändpunkt i förbindelse med <_AMODE> (ENTAL)		
4	X1	<_FPD>	REAL	Ändpunkt i förbindelse med <_AMODE> (TIOTAL)		
5	LW/LW2	<_APP>	REAL	Gänginkörning i förbindelse med <_AMODE> (HUNDRATAL) eller Gängingång = gängutgång i förbindelse med <_AMODE> (HUNDRA-TAL)		
6	LR	<_ROP>	REAL	Gängutgång		
7	H1	<_TDEP>	REAL	Gängdjup		
8	U	<_FAL>	REAL	Finbearbetningsmått i X och Z		
9	DP	<_IANG>	REAL	Ansättningslutning som avstånd eller vinkel, i förbindelse med <_AMODE> (TUSENTAL)		
				> 0 =	ansättning vid den positiva flanken	
				< 0 =	ansättning vid den negativa flanken	
				0 =	ansättning i mitten	
10	α0	<_NSP>	REAL	Startvinkelförskjutning (verkar endast vid "engängig")		
11	ND	<_NRC>	INT	Antal grovbearbetningsskär, i förbindelse med <_VARI> (TIOTUSENTAL)		
12	NN	<_NID>	INT	Antal rensningspassager		
13	P	<_PIT>	REAL	Gängstigning som värde, i förbindelse med <_PITA>		

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
14		<_VARI>	INT	Bearbetningstyp
				ENTAL:
				Teknologi
				1 = yttergånga med linjär ansättning
				2 = yttergånga med linjär ansättning
				3 = yttergånga med degressiv ansättning, spånarea förblir konstant
				4 = Innergånga med degressiv ansättning, spånarea förblir konstant
				TIOTAL:
				reserverat
				HUNDRATAL:
				Ansättningstyp
				1 = ensidig ansättning
				2 = växlande ansättning
				TUSENTAL:
				reserverat
		<_VARI>	INT	TIOTUSENTAL:
				Alternativ djupansättning
				0 = föreskrivet antal grovbearbetningsskär (<_NRC>)
				1 = föreskrivet värde för 1:a ansättningen (<_MID>)
				HUNDRATUSENTAL:
				Bearbetningstyp
				1 = grovbearbetning
				2 = finbearbetning
				3 = grovbearbetning och finbearbetning
				ENMILJONTAL:
				Bearbetningsordningsföljd vid gängor med flera ingångar
				0 = ordningsföljd för gängorna stigande
				1 = ordningsföljd för gängorna omvänd
15	N	<_NUMTH>	INT	Antal gängor
16	VR	<_SDIS>	REAL	Återgångsavstånd, ink
17	D1	<_MID>	REAL	Första ansättningsdjup, i förbindelse med <_VARI> (TIOTUSENTAL)
18	DA	<_GDEP>	REAL	Gängväxlingsdjup (verkar endast vid "flera ingångar")
				0 = inte ta hänsyn till något gängväxlingsdjup
				> 0 = ta hänsyn till gängväxlingsdjup
19	G	<_PIT1>	REAL	Ändring av stigningen per varv
				0 = gängstigningen är konstant (G33)
				> 0 = gängstigningen blir större (G34)
				< 0 = gängstigningen blir mindre (G35)
20		<_FDEP>	REAL	Nermatningsdjup (mata in utan förtecken)
21	N1	<_GST>	INT	Startgånga N1 = 1...N, i förbindelse med <_AMODE> (HUNDRATUSENTAL)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
22		< _GUD>	INT	reserverat		
23		< _IFLANK>	REAL	Ansättningslutning som bredd (endast för ytan)		
24		< _PITA>	INT		0 =	stigning i mm - utvärdering MPIT/ PIT
					1 =	stigning i mm - utvärdering PIT
					2 =	stigning i TPI - utvärdering PIT (gångor per inch)
					3 =	stigning i inch - utvärdering PIT
					4 =	MODUL - utvärdering PIT
25		< _PITM>	STRING[15]	String som flagga för inmatning av gängstigning (endast för ytan) ¹⁾		
26		< _PTAB>	STRING[20]	String för gängtabell (endast för ytan) ¹⁾		
27		< _PTABA>	STRING[20]	String för val i gängtabellen (endast för ytan) ¹⁾		
28		< _DMODE>	INT	Indikeringsmode		
				ENTAL:	Bearbetningsplan G17/G18/G19	
					0 =	kompatibilitet, det före cykelan-rop verksamma planet förblir aktivt
					1 =	G17 (endast aktiv i cykeln)
					2 =	G18 (endast aktiv i cykeln)
					3 =	G19 (endast aktiv i cykeln)
				TIOTAL:	Gängtyp	
					0 =	Längsgänga
					1 =	Plangänga
					2 =	Kongänga
				HUNDRATAL:	---	reserverat
				TUSENTAL:	---	reserverat
				TIOTUSENTAL:	Teknologiskalning inom cykelmaskerna (Sida 814)	
					0 =	inmatning: komplett
					1 =	inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
29		<_AMODE>	INT	Alternativmode
				ENTAL:
				Gänglängd i Z
				0 = absolut
				1 = inkrementell
				TIOTAL:
				Gänglängd i X
				0 = absolut, värde för planaxeln i diametern
				1 = inkrementell, värde för planaxeln i radien
				2 = α
				HUNDRATAL:
				Inkörnings-/ingångsvägvärdering <_APP>
				0 = Gängförsprång <_APP>
				1 = Gänginkörning = gängutkörning <_APP> = -<_ROP>
				2 = Gänginkörning föreskriven <_APP> = -<_APP>
30	XS / RS	<_S_XRS>	REAL	TUSENTAL:
				Val ansättningslutning som vinkel eller bredd
				0 = ansättningsvinkel <_IANG>
				1 = ansättningslutning <_IFLANK>
				TIOTUSENTAL:
				en ingång/flera ingångar
				0 = en ingång (med startvinkelförskjutning <_NSP>)
				1 = med flera ingångar
				HUNDRATUSENTAL:
				Startgंगा <_GST>
				0 = komplett-bearbetning
				1 = starta bearbetningen från denna gंगा
				2 = bearbeta endast denna gंगा
				ENMILJONTAL:
				Överhängskompensation vid längsgंगा
				0 = Segmenthöjd kullrig gंगा XS
				1 = Radie kullrig gंगा RS
				Överhängskompensation vid längsgंगा i förbindelse med <_AMODE>: ENMILJONTAL

Märk

¹⁾ Parametrarna <_PITM>, <_PTAB> och <_PTABA> används endast vid val av gंगा i gंगtabellen för inmatningsmasken.

En åtkomst på gंगtabellerna via cykeldefinition vid cykelkörtid är inte möjligt.

20.1.31 CYCLE435 - ställa in koordinatsystem för skärpverktyg

Syntax

```
CYCLE435(<_T>, <_DD>, <S_TA>, <S_DA>, <S_AD>, <S_AL>, <S_PVD>,
<S_PVL>, <S_PD>, <S_PL>, <_AMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1		<_T>	STRING[32]	Verktögsnamn för slipskivan
2		<_DD>	INT	Skärnummer för slipskivan
3		<S_TA>	STRING[32]	Referenspunkt skärpverktyg - namn på skärpverktyget
4		<S_DA>	INT	Skärnummer för skärpverktyget
5		<S_AD>	REAL	Skärpbelopp diameter
6		<S_AL>	REAL	Skärpbelopp plan
7		<S_PVD>	REAL	Profilerförflyttning diameter
8		<S_PVL>	REAL	Profilerförflyttning plan
9		<S_PD>	REAL	Profileringsavmått diameter
10		<S_PL>	REAL	Profileringsavmått plan
11		<_AMODE>	INT	Alternativmode
				ENTAL:
				aktivt verktyg vid cykelslut
				0 = Skärpverktyg aktivt
				1 = Skiva aktiv

20.1.32 CYCLE495 - profilera

Syntax

```
CYCLE495(<_T>, <_DD>, <_SC>, <_F>, <_VARI>, <_D>, <_DX>, <_DZ>,
<S_PA>, <S_N>, <_DMODE>, <_AMODE>, <S_FW>, <S_HW>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1		<_T>	STRING[20]	Verktögsnamn för slipskivan
2		<_DD>	INT	Skärnummer för slipskivan
3		<_SC>	REAL	Lyftningsbelopp för körning runt hinder, inkrementellt
4		<_F>	REAL	Matning profilering

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
5		<_VARI>	INT	Bearbetningstyp
				ENTAL:
				Profileringstyp
				1 = axelparallellt
				2 = konturparallellt
				TIOTAL:
				Bearbetningsriktning
				0 = dragande möjligt med skärlägena 1 till 4
				1 = stötande möjligt med skärlägena 1 till 4
				2 = växlande möjligt med skärlägena 1 till 8
				3 = början → slut möjligt med skärlägena 1 till 8
				4 = slut → början möjligt med skärlägena 1 till 8
6		<_D>	REAL	HUNDRATAL:
				Ansättningsriktning
				1 = Ansättning X- vid G18 resp. Y- vid G19
				2 = Ansättning X+ vid G18 resp. Y+ vid G19
				3 = Ansättning Z- vid G18 och vid G19
7		<_DX>	REAL	4 = Ansättning Z+ vid G18 och vid G19
8		<_DZ>	REAL	Skärpbelopp vid profileringstyp axelparallell
9		<S_PA>	REAL	Skärpbelopp X vid G18 resp. Y vid G19 vid profileringstyp konturparallell
10		<S_N>	REAL	Skärpbelopp Z vid G18 och G19 vid profileringstyp konturparallell
11		<S_PA>	REAL	Profileringsavmätt
12		<S_N>	INT	Antal slag i profileringsprogrammet
13		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
14		<_DMODE>	INT	3 = G19 (endast aktiv i cykeln)

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
12		<_AMODE>	INT	Alternativmode
				ENTAL:
				Urval profilera nytt/fortsätta
				1 = Nytt
				2 = Fortsätta
				TOTAL:
				Urval profileringsavmått
				0 = från råkonturen till den lägsta punkten på konturen
				1 = från råkonturen till den högst punkten på konturen
13		<S_FW>	REAL	Frivinkel för skärpverktyget
14		<S_HW>	REAL	Hållarvinkel för skärpverktyget

20.1.33 CYCLE800 - vridning

Syntax

CYCLE800(<_FR>, <_TC>, <_ST>, <_MODE>, <_X0>, <_Y0>, <_Z0>, <_A>, <_B>, <_C>, <_X1>, <_Y1>, <_Z1>, <_DIR>, <_FR_I>, <_DMODE>)

Parametrar

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
1		<_FR>	INT	Frikörningsmodus:
				0 = ingen frikörning
				1 = frikörning maskinaxel Z
				2 = frikörning maskinaxel Z och där- efter XY
				3 = reserverat
				4 = friköra maximalt i verktygsriktning
				5 = friköra inkrementellt i verktygsriktning
2		<_TC>	STRING[32]	Namn vriddatablock:
				"" (inget namn) när bara 1 vriddatablock finns
				"0" bortval vriddatablock (radering av vridframes)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
3		<_ST>	INT	Status transformationer
				ENTAL:
				0 = ny , vridningsplan raderas och beräknas på nytt med de aktuella parametrarna
				1 = additiv , vridningsplan läggs additivt till aktivt vridningsplan
				TIOTAL:
				Verktygsspets medföljning ja/nej (endast aktiv, om inställd i IBN VRIDA funktionen)
				0 = ingen medföljning av verktygsspetsen.
				1 = medföljning av verktygsspetsen (TRAORI)
				HUNDRATAL:
				Ansätta / uppriktat verktyg (funktionen visas i inmatningsmasken VRIDA verktyg)
				0 = inte ansätta verktyg
				1 = ansätta verktyg (helst radiefrys)
				2 = uppriktat svarvverktyg (om B-axelkinematik för vridteknologi är inställd i IBN Vrida)
				3 = uppriktat fräsvverktyg (om B-axelkinematik för vridteknologi är inställd i IBN Vrida)
				TUSENTAL:
				Intern parameter Vrida i JOG
				TIOTUSENTAL:
				se parameter Riktning <_DIR>
				0 = vrida "ja"
				1 = vrida "nej" riktning "minus" ³⁾
				2 = vrida "nej" riktning "plus" ³⁾
				HUNDRATUSENTAL:
				se parameter Riktning <_DIR>
				0 = kompatibilitet
				1 = Riktningsval "minus" optimerat (endast för användargränssnitt) ⁴⁾
				2 = Riktningsval "plus" optimerat (endast för användargränssnitt) ⁴⁾

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
4		<_MODE> ⁵⁾	INT	Vridningsmodus: Utvärdering av vridvinklarna och ordningsföljden för vridningarna (bitcoderat!)
				Bit: 7 6
				0 0: Vridvinkel axelvis -> se parameter <_A>, <_B>, <_C>
				0 1: Rymdvinkel -> se parameter <_A>, <_B> ¹⁾
				1 0: Projektionsvinkel -> se parameter <_A>, <_B>, <_C> ¹⁾
				1 1: Vridningsmodus roterande axlar direkt -> se parameter <_A>, <_B> ¹⁾
				Bit: 5 4 3 2 1 0 (vid rymdvinklar utan betydelse!)
				x x x x 0 1 1.vridning _A runt X
				x x x x 1 0 1.vridning _A runt Y
				x x x x 1 1 1.vridning _A runt Z
				x x 0 1 x x 2.vridning _B runt X
				x x 1 0 x x 2.vridning _B runt Y
				x x 1 1 x x 2.vridning _B runt Z
5	X0	<_X0>	REAL	Referenspunkt X före vridning
6	Y0	<_Y0>	REAL	Referenspunkt Y före vridning
7	Z0	<_Z0>	REAL	Referenspunkt Z före vridning
8	X(A)	<_A>	REAL	1:a vridning enligt inställning i parameter <_MODE>
9	Y(B)	<_B>	REAL	2:a vridning enligt inställning i parameter <_MODE>
10	Z(C)	<_C>	REAL	3:e vridning enligt inställning i parameter <_MODE>
11	X1	<_X1>	REAL	Referenspunkt X efter vridningen
12	Y1	<_Y1>	REAL	Referenspunkt Y efter vridningen
13	Z1	<_Z1>	REAL	Referenspunkt Z efter vridningen
14	- eller +	<_DIR>	INT	Utlösa förflyttningsrörelse för de roterande axlarna (default = -1!)
				-1 = positionera på det mindre värdet för de roterande axlarna 1 eller 2 ²⁾
				+1 = positionera på det större värdet för de roterande axlarna 1 eller 2 ²⁾
				0 = vridning nej (endast beräkna vridframe) ^{1) 3)}
15	FR	<_FR_I>	REAL	Värde (ink) frikörning i verktygsriktning inkrementell

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
16		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL:
				Framställning ärdet vid uppriktning av verktyget
				0 = Värde
				1 = Pil

Märk

Är följande överföringsparametrar programmerade indirekt (som parameter), återöversätts inte inmatningsmasken: <_FR>, <_ST>, <_TC>, <_MODE>, <_DIR>

¹⁾ Val möjligt, om fuktion inställd i IBN VRIDA.

²⁾ Val möjligt, om riktningsreferens är inställd på de roterande axlarna 1 eller 2 i IBN VRIDA.

Ingen urvalsruta vid riktningsreferens nej

³⁾ Val vrida "nej" kan vara gömt SD 55221 Bit 0

Vrida "nej" riktning "minus" motsvarar <_DIR> = 0 och _ST TIOTUSENTAL = 1

Vrida "nej" riktning "plus" motsvarar <_DIR> = 0 och _ST TIOTUSENTAL = 2

⁴⁾ Riktningsvalet för den roterande axeln 1 eller 2 görs också när den roterande axeln befinner sig med riktningsreferensen i polläge (positionsvärde lika med noll).

⁵⁾ Exempel kodning: Vridning axelvis, ordningsföljd för vridning ZYX

Binär: 00011011; decimal: 27

Axelbeteckningen XYZ motsvarar geometriaxlarna i NC-kanalen. Vridningarna runt axlarna XYZ får utföras separat. T. ex. ordningsföljden för vridningen runt ZXZ är inte tillåtet i ett anrop från CYCLE800

20.1.34 CYCLE801 - gitter eller ram**Syntax**

```
CYCLE801 (<_SPCA>, <_SPCO>, <_STA>, <_DIS1>, <_DIS2>, <_NUM1>,
<_NUM2>, <_VARI>, <_UMODE>, <_ANG1>, <_ANG2>, <_HIDE>, <_NSP>,
<_DMODE>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse		
1	X0	<_SPCA>	REAL	Referenspunkt för positionsmönster (gitter/ram) i den 1:a axeln (abs)		
2	Y0	<_SPCO>	REAL	Referenspunkt för positionsmönster (gitter/ram) i den 2:a axeln (abs)		
3	$\alpha 0$	<_STA>	REAL	Grund-rotationsvinkel (vinkel till 1:a axeln)	< 0 =	Medurs rotation
					> 0 =	Moturs rotation
4	L1	<_DIS1>	REAL	Avstånd för spalterna (positionsavstånd 1:a axeln, mata in utan förtecknen)		
5	L2	<_DIS2>	REAL	Avstånd för raderna (positionsavstånd 2:a axeln, mata in utan förtecken)		
6	N1	<_NUM1>	INT	Antal spalter		
7	N2	<_NUM2>	INT	Antal rader		
8		<_VARI>	INT	Bearbetningstyp		
				ENTAL:	Positionsmönster	
					0 =	Gitter
					1 =	Ram
				TIOTAL:	reserverat	
				HUNDRATAL:	reserverat	
9		<_UMODE>	INT	reserverat		
10	αX	<_ANG1>	REAL	Skjuvvinkel till den 1:a axeln (snedställning av raderna relaterade till den 1:a axeln)		
					< 0 =	mätning medurs (0 till -90 grader)
					> 0 =	mätning moturs (0 till 90 grader)
11	αY	<_ANG2>	REAL	Skjuvvinkel till den 2:a axeln (snedställning av spalterna relaterade till den 2:a axeln)		
					< 0 =	mätning medurs (0 till -90 grader)
					> 0 =	mätning moturs (0 till 90 grader)
12		<_HIDE>	STRING [200]	Gömda positioner - max. 198 tecken - Uppgift över det fortlöpande positionsnumret, t.ex. "1,3" (positioner 1 och 3 utförs inte)		
13		<_NSP>	INT	reserverat		
14		<_DMODE>	INT	Indikeringsmode		
				ENTAL:	Bearbetningsplan G17/G18/G19	
					0 =	kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
					1 =	G17 (endast aktiv i cykeln)
					2 =	G18 (endast aktiv i cykeln)
					3 =	G19 (endast aktiv i cykeln)

20.1.35 CYCLE802 - valfria positioner

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1		<_XA>	INT	Alternativ för alla X-positioner (9-siffrigt decimalvärde) Antal platser: 876543210 (Platsen motsvarar borrhpositionen Xn)
				Platsvärde:
				1 = absolut (1:a programmerade position alltid absolut!) 2 = inkrementell
2		<_YA>	INT	Alternativ för alla Y-positioner (9-siffrigt decimalvärde) Antal platser: 876543210 (Platsen motsvarar borrhpositionen Yn)
				Platsvärde:
				1 = absolut (1:a programmerade position alltid absolut!) 2 = inkrementell
3	X0	<_X0>	REAL	1:a position X
4	Y0	<_Y0>	REAL	1:a position Y
5	X1	<_X1>	REAL	2:a position X
6	Y1	<_Y1>	REAL	2:a position Y
7	X2	<_X2>	REAL	3:e position X
8	Y2	<_Y2>	REAL	3:e position Y
9	X3	<_X3>	REAL	4. position X
10	Y3	<_Y3>	REAL	4. position Y
11	X4	<_X4>	REAL	5. position X
12	Y4	<_Y4>	REAL	5. position Y
13	X5	<_X5>	REAL	6. position X
14	Y5	<_Y5>	REAL	6. position Y
15	X6	<_X6>	REAL	7. position X
16	Y6	<_Y6>	REAL	7. position Y
17	X7	<_X7>	REAL	8. position X
18	Y7	<_Y7>	REAL	8. position Y
19	X8	<_X8>	REAL	9. position X
20	Y8	<_Y8>	REAL	9. position Y

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
21		<_VARI>	INT	Bearbetning
				HUNDRATAL:
				(endast för anrop från jobshop) (först endast 0 och 2 utvärderade)
				0 =
				inte låsa någon spindel
				1 =
				låsa spindel endast vid lodrät ner- matning med G00 eller G01
				2 =
				låsa spindel under hela bearbet- ningen
				TUSENTAL:
				reserverat
				TIOTUSENTAL:
				Positionsmönster med/utan roterande axel – axelkombination (med <_VARI> HUNDRATUSENTAL)
				0 =
				XY (endast XY utan roterande axel, kompatibilitet)
22		<_UMODE>	INT	1 =
				X,Y eller Z och roterande axel: XA, YB, ZC (1 roterande axel med geo-axel, runt vilken den roterande axel vri- der sig)
				2 =
				XY och roterande axel: XYA, XYB, XYC (1 roterande axel med 1:a och 2:a geo-axel, utan TRACYL)
				HUNDRATUSENTAL:
				roterande axel
				0 =
				utan roterande axel (endast XY, kompatibilitet)
				1 =
				A-axel (roterande axel runt X)
				2 =
				B-axel (roterande axel runt Y)
				3 =
				C-axel (roterande axel runt Z)
				TIOMILJONTAL + ENMIL- JONTAL:
				Positionsmönster med roterande axel – offset (vid flera roterande axlar runt samma axel; när index för stort, då 1:a axeln)
				00 =
				1:a A, B eller C-axel eller vid kom- patibilitet
				01 =
				2:a A, B eller C-axel
				...
				19 =
				20. A, B eller C-axel
22		<_UMODE>	INT	Urval för den spindel som ska låsas: (endast för anrop från jobshop) (anrop användarcykel CUST_TechCYC)
				3 =
				Låsa/lossa huvudspindel
				23 =
				Låsa/lossa motspindel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
23		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
24		<S_ABA>	INT	Alternativ för alla AB-positioner (9-siffrigt decimalvärde)
				Antal platser: 876543210 (Platsen motsvarar positionen ABn)
				Platsvärde:
				1 = absolut (1:a programmerade position alltid absolut!)
				2 = inkrementell
25	A0	<S_AB0>	REAL	1:a Roterande axelposition vid positionsmönster med roterande axel (i förbindelse med <_VARI>)
26	A1	<S_AB1>	REAL	2:a Roterande axelposition vid positionsmönster med roterande axel
27	A2	<S_AB2>	REAL	3:e Roterande axelposition vid positionsmönster med roterande axel
28	A3	<S_AB3>	REAL	4:e Roterande axelposition vid positionsmönster med roterande axel
29	A4	<S_AB4>	REAL	5:e Roterande axelposition vid positionsmönster med roterande axel
30	A5	<S_AB5>	REAL	6:e Roterande axelposition vid positionsmönster med roterande axel
31	A6	<S_AB6>	REAL	7:e Roterande axelposition vid positionsmönster med roterande axel
32	A7	<S_AB7>	REAL	8:e Roterande axelposition vid positionsmönster med roterande axel
33	A8	<S_AB8>	REAL	9:e Roterande axelposition vid positionsmönster med roterande axel

Märk

Ej nödvändiga positioner för parametrarna X1/Y1/A1 till X8/Y8/A8 kan utelämnas.

Alternativvärdena för <_XA>, <_YA> och <S_ABA> ska dock anges fullständigt för alla 9 positionerna.

Vid positionsmönster XA, YB eller ZC (en geometriaxel och roterande axel) ska axeln i bearbetningsplanet, som inte förflyttas över positionsmönstret (vid G17 och XA är det Y) positioneras före cykelanrop.

20.1.36 CYCLE830 - djuphålsborra 2**Syntax**

```
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>)
```

Parameter

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
1	RP	<RTP>	REAL	Återgångsplan (abs)
2	Z0	<RFP>	REAL	Referenspunkt (abs)
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, utan förtecken)
4	Z1	<_DP>	REAL	Slutborrdjup abs/ink (se <_AMODE> ENTAL)
5	D	<FDEP>	REAL	1:a borrdjupet för spånbrytning/urspåning absolut eller inkrementellt relaterat till referenspunkten vid förborring/utan förborring eller relaterad till pilotborrningsdjup (se <_AMODE> TIOTUSENTAL)
6	DF	<_DAM>	REAL	Belopp/procentsats för varje ytterligare ansättning degressionsbelopp eller procentsats, (se <_AMODE> HUNDRATUSENTAL)
7	DTB	<DTB>	REAL	Fördröjningstid på varje borrdjup (se <_AMODE> TIOTAL)
8	DTS	<DTS>	REAL	Fördröjningstid vid urspåning i begynnelsepunkten (se <_AMODE> HUNDRATAL)
9	FD1	<FRF>	REAL	Procentsats för matningen vid första ansättningen, se <_AMODE> TIO-MILJONTAL)
10		<VARI>	INT	Bearbetning ENTAL: Spånbrytning/urspåning 0 = i ett skär 1 = Spånbrytning 2 = Urspåning 3 = Spånbrytning och urspåning TIOTAL: Återgång vid urspåning 0 = på pilotborrdjup 1 = på säkerhetsavstånd HUNDRATAL: mjuk inskärning 0 = nej 1 = ja TUSENTAL: Genomgående borring 0 = nej 1 = ja TIOTUSENTAL: Förborring/pilotborring 0 = utan förborring 1 = med förborring 2 = med pilothål HUNDRATUSENTAL: Återgång 0 = på pilotborrdjup 1 = till återgångsplanet
11	V1	<_MDEP>	REAL	Minimal ansättning inkrementell (endast vid procentsats för degression)
12	V2	<_VRT>	REAL	Återgångsvärde efter varje bearbetning inkrementellt (endast vid spån- brytning) 0 = standardvärde 1 mm > 0 = variabelt återgångsvärde
13	DT	<_DTD>	REAL	Fördröjningstid på slutborrdjup, (se <_AMODE> TUSENTAL)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
14	V3	<_DIS1>	REAL	Förhållningsavstånd inkrementellt endast vid urspånning (se <_AMODE> ENMILJONTAL)
15	FP	<S_FP>	REAL	Matning för inkörning i pilothålet som värde eller i % (i förbindelse med <S_AMODE2> HUNDRATAL)
16		<S_SDAC2>	INT	Spindelns rotationsriktning vid uppsökning
				3 = M3
				4 = M4
				5 = M5 (default)
17	SP	<S_SV2>	REAL	Uppsökning med
	V4			konstant spindelvarvtal, se <S_AMODE2> (TIOMILJONTAL)
				konstant skärhastighet
				Spindelvarvtal i % av borrarvarvtalet
18	F	<S_FB>	REAL	Borrmatning (se <S_AMODE2> ENTAL)
19		<_SDAC>	REAL	Spindelns rotationsriktning vid borrar
				3 = M3
				4 = M4
20	S	<_SV1>	REAL	Borra med
	V5			konstant spindelvarvtal, se <S_AMODE2> (ENMILJONTAL)
				konstant skärhastighet
21	SPOS	<S_SPOS>	REAL	Spindelposition, endast om uppsökning med M5
22	ZA	<S_ZA>	REAL	Förborrdjup inkrementellt relaterat till referenspunkt eller absolut (se <S_AMODE3> ENTAL)
23	FA	<S_FA>	REAL	Förborrmatning som värde eller i % (i förbindelse med <S_AMODE2> TIOTAL)
24	ZP	<S_ZP>	REAL	Pilothål inkrementellt relaterat till referenspunkt, absolut eller faktor för håldiametern (se <S_AMODE3> TIOTAL)
25	FS	<S_FS>	REAL	Matning inskärning som värde eller i % (i förbindelse med <S_AMODE2> TUSENTAL)
26	ZS1	<S_ZS1>	REAL	Djup för varje inskärning med konstant matning (ink)
27	ZS2	<S_ZS2>	REAL	Djup för varje inskärning för matningsökning (ink)
28	N	<S_N>	INT	Antal spånbrytningsslag före varje urspånning
29	ZD	<S_ZD>	REAL	Restborrdjup inkrementellt relaterat till slutborrdjup eller absolut (i förbindelse med <S_AMODE3> HUNDRATAL)
30	FD	<S_FD>	REAL	Restborrmatning som värde eller i % (i förbindelse med <S_AMODE2> TIOTUSENTAL)
31	FR	<S_FR>	REAL	Matning återgång (i förbindelse med <S_AMODE2> (HUNDRATUSENTAL)
32		<S_SDAC3>	INT	Spindelns rotationsriktning vid återgång
				3 = M3
				4 = M4
				5 = M5
33	SR	<S_SV3>	REAL	Återgång med
	V6			konstant spindelvarvtal, se <S_AMODE2> (HUNDRAMILJONTAL)
				konstant skärhastighet
				Spindelvarvtal i % av borrarvarvtalet

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
34	Skärvätska till	<S_CON>	STRING[10]	Skärvätska till, M-kommando eller underprogramanrop
35	Skärvätska från.	<S_COFF>	STRING[10]	Skärvätska från, M-kommando eller underprogramanrop
36		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TIOTAL: Borrdjup relaterat till spets/skaft
				0 = Spets
				1 = Skaft
37		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL: reserverat
				HUNDRATAL: reserverat
				TUSENTAL: reserverat
				TIOTUSENTAL: Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
38		<_AMODE>	INT	Alternativmode 1
				ENTAL:
				Borrdjup = slutborrdjup Z1 abs/ink
				0 = inkrementell
				1 = absolut
				TIOTAL:
				Fördröjningstid på varje borrdjup DTB i sekunder/varv
				0 = i sekunder:
				1 = i varv
				HUNDRATAL:
				Fördröjningstid för urspänning DTS i sekunder/varv
				0 = i sekunder:
				1 = i varv
				TUSENTAL:
				Fördröjningstid på slutborrdjup DT i sekunder/varv
				0 = i sekunder:
				1 = i varv
				TIOTUSENTAL:
				1:a borrdjupet D (abs/ink)
				0 = inkrementell
				1 = absolut
				HUNDRATUSENTAL:
				Belopp/procentsats DF för varje ytterligare ansättning (degression)
				0 = Belopp
				1 = Procentsats (0,001 till 100 %)
				ENMILJONTAL:
				Förstoppavstånd V3 automatiskt/manuellt
				0 = automatiskt (beräknas i cykeln)
				1 = manuellt (programmerat värde)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
39		<S_AMODE2 >	INT	Alternativmode 2
				ENTAL:
				ENTAL: Borrmätning F
				0 = F/min
				1 = F/varv
				TIOTAL:
				Värdering förborrmätning FA
				0 = i % av borrmätning
				1 = F/min
				2 = F/varv
				HUNDRATAL:
				Värdering matning för inkörning i pilothål FP
				0 = i % av borrmätning
				1 = F/min
				2 = F/varv
				TUSENTAL:
				Värdering matning inskränning FS
				0 = i % av borrmätning
				1 = F/min
				2 = F/varv
				TIOTUSENTAL:
				Värdering matning genomborrning FD
				0 = i % av borrmätning
				1 = F/min
				2 = F/varv
				HUNDRATUSENTAL:
				Matning återgång FR
				0 = F/min
				1 = Snabbmatning
				ENMILJONTAL:
				Borra - spindelvarvtal/skärhastighet (S/V5)
				0 = konstant spindelvarvtal
				1 = konstant skärhastighet
				TIOMILJONTAL:
				Uppsökning med spindelvarvtal/skärhastighet (SP/V4)
				0 = konstant spindelvarvtal
				1 = konstant skärhastighet
				2 = Spindelvarvtal i % av borrvavrtalet
				HUNDRAMILJONTAL:
				Återgång - spindelvarvtal/skärhastighet (SR/V6)
				0 = konstant spindelvarvtal
				1 = konstant skärhastighet
				2 = Spindelvarvtal i % av borrvavrtalet

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
40		<S_AMODE3 >	INT	Alternativmode 3
				ENTAL:
				Förborrdjup ZA abs/ink
				0 = inkrementell
				1 = absolut
				TIOTAL:
				Djup för pilothålet ZP
				0 = inkrementell
				1 = absolut
				2 = Faktor för håldiametern
41	ZPV	<S_ZPV>	REAL	HUNDRATAL:
				Restborrdjup ZD ink/abs
				0 = inkrementell
				1 = absolut
				Förhållningsavstånd inkrementellt på pilotborrningsdjup

20.1.37 CYCLE832 - High Speed Settings

Syntax

CYCLE832 (<S_TOL>, <S_TOLM>, <S_OTOL>)

Märk

CYCLE832 avlastar inte maskintillverkaren från nödvändiga optimeringsuppgifter vid idrifttagningen av maskinen. Detta gäller optimeringen av de axlar som deltar i bearbetningen och inställningarna på NCU (förstyrning, jerkbegränsning, osv.).

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse																																	
1	Tole-rans	<S_TOL>	REAL	Konturtolerans Konturtoleransen motsvarar axeltoleransen för geometriaxlarna.																																	
2		<S_TOLM>	INT	<table><tr><td colspan="3">Bearbetningstyp (teknologi)</td></tr><tr><td rowspan="5">ENTAL:</td><td colspan="2"></td></tr><tr><td>0 =</td><td>Bortval</td></tr><tr><td>1 =</td><td>finbearbetning (finish)</td></tr><tr><td>2 =</td><td>förfinbearbetning (semifinish)</td></tr><tr><td>3 =</td><td>grovbearbetning (rough)</td></tr><tr><td rowspan="3">TIOTAL:</td><td colspan="2"></td></tr><tr><td>0 =</td><td>kompatibilitet¹⁾ resp. ingen orienteringstolerans</td></tr><tr><td>1 =</td><td>Orienteringstolerans i parametern <S_OTOL></td></tr><tr><td>HUNDRA ... HUNDRATUSEN</td><td colspan="2">upptagen på grund av kompatibilitet</td></tr><tr><td rowspan="4">ENMILJONTAL:</td><td colspan="2"></td></tr><tr><td>0 =</td><td>Kompatibilitet. Den bästa formbyggnadsfunktionen som står till förfogande används automatiskt: - Option Top Surface inte aktiv: ⇒ Advanced Surface - Option Top Surface aktiv: ⇒ Top Surface med jämning </td></tr><tr><td>1 =</td><td>Top Surface utan jämning</td></tr><tr><td>2 =</td><td>Top Surface med jämning</td></tr></table>	Bearbetningstyp (teknologi)			ENTAL:			0 =	Bortval	1 =	finbearbetning (finish)	2 =	förfinbearbetning (semifinish)	3 =	grovbearbetning (rough)	TIOTAL:			0 =	kompatibilitet ¹⁾ resp. ingen orienteringstolerans	1 =	Orienteringstolerans i parametern <S_OTOL>	HUNDRA ... HUNDRATUSEN	upptagen på grund av kompatibilitet		ENMILJONTAL:			0 =	Kompatibilitet. Den bästa formbyggnadsfunktionen som står till förfogande används automatiskt: - Option Top Surface inte aktiv: ⇒ Advanced Surface - Option Top Surface aktiv: ⇒ Top Surface med jämning	1 =	Top Surface utan jämning	2 =	Top Surface med jämning
Bearbetningstyp (teknologi)																																					
ENTAL:																																					
	0 =	Bortval																																			
	1 =	finbearbetning (finish)																																			
	2 =	förfinbearbetning (semifinish)																																			
	3 =	grovbearbetning (rough)																																			
TIOTAL:																																					
	0 =	kompatibilitet ¹⁾ resp. ingen orienteringstolerans																																			
	1 =	Orienteringstolerans i parametern <S_OTOL>																																			
HUNDRA ... HUNDRATUSEN	upptagen på grund av kompatibilitet																																				
ENMILJONTAL:																																					
	0 =	Kompatibilitet. Den bästa formbyggnadsfunktionen som står till förfogande används automatiskt: - Option Top Surface inte aktiv: ⇒ Advanced Surface - Option Top Surface aktiv: ⇒ Top Surface med jämning																																			
	1 =	Top Surface utan jämning																																			
	2 =	Top Surface med jämning																																			
3	ORI-Tolerans	<S_OTOL>	REAL	Orienteringstolerans resp. versionskännetecken CYCLE832 Toleransparameter för orienteringen av verktyget. Behövs vid genomarbetningen av ett höghastighetsbearbetningsprogram på maskiner med dynamisk orienteringstransformation (t. ex. 5-axel-bearbetningen). Parametern <S_OTOL> måste programmeras. Detta gäller också för användningar på 3-axel-maskiner vid program utan orientering av verktyget (<S_OTOL> = 1).																																	

¹⁾ Orienteringstolerans härledd från konturtoleransen multiplicerad med faktorn från cykel-settingdata SD55441 till SD55443.

Litteratur:

Idrifttagningshandbok Bassoftware och manöversoftware; SINUMERIK Operate (IM9), Kapitel "Konfigurera funktionen High Speed Setting (CYCLE832)"

Klartextinmatning

För att förbättra cykelanropets läsbarhet, kan parametern <S_TOLM> "Bearbetningstyp" också matas in som klartext. Klartexter är språkberoende. Följande inmatningar är tillåtna:

_OFF	för	0	Bortval
_FINISH	för	1	Finbearbetning
_SEMIFIN	för	2	förfinbearbetning
_ROUGH	för	3	Grovbearbetning
_ORI_FINISH	för	11	Finbearbetning med inmatning av en orienteringstolerans
_ORI_SEMIFIN	för	12	Förfinbearbetning med inmatning av en orienteringstolerans
_ORI_ROUGH	för	13	Grovbearbetning med inmatning av en orienteringstolerans
_TOP_SURFACE_SMOOTH_OFF	för	1000000	Top Surface utan jämnning
_TOP_SURFACE_SMOOTH_ON	för	2000000	Top Surface med jämnning

För klartextinmatningen vid Top Surface kombineras klartexterna som i följande exempel:

```
CYCLE832 (0.1, _TOP_SURFACE_SMOOTH_OFF+_ORI_FINISH, 1)
```

Märk

Klartexterna stöder sig på funktionsnamnen i G-gruppen 59 (Dynamikmode för baninterpolering). Med dessa klartexter åtskiljs 3-axel-maskiner och maskiner med flerfaldig orienteringstransformation (TRAORI) tydligt i användningen.

Bortval av CYCLE832

Vid bortval av CYCLE832 måste parametern <S_TOL> överföras med noll.

Exempel: CYCLE832 (0,0,1)

Syntaxen CYCLE832 () är också tillåten för bortval av CYCLE832.

Exempel

Exempel 1: CYCLE832 på 3-axel-maskin utan orienteringstransformation

a) Cykelanrop med klartextinmatning

Programkod	Kommentar
G710	; Måttsystem är metriskt.
CYCLE832 (0.004, _FINISH, 1)	; Anrop CYCLE832 med: Konturtolerans = 0,004 mm, bearbetningstyp: finbearbetning
...	; Genomarbetning av ett höghastighets-bearbe- tnings-program

b) Cykelanrop utan klartextinmatning

Programkod	Kommentar
G710	; S. O.
CYCLE832(0.004,1,1)	; S. O.
...	; S. O.

Exempel 2: CYCLE832 på 5-axel-maskin med orienteringstransformation

a) Cykelanrop och bortval med klartextinmatning

Programkod	Kommentar
G710	; Måttsystem är metriskt.
TRAORI	; Aktivera orienteringstransformation.
CYCLE832(0.3,_ORI_ROUGH,0.8)	; Anrop CYCLE832 med: Konturtolerans = 0,3 mm, bearbetningstyp: Grov- bearbetning med inmatning av en orienteringsto- lerans, orienteringstolerans = 0,8 grader
...	; Genomarbetning av ett höghastighets-bearbe- tnings-program
CYCLE832(0,_OFF,1)	; Konturtolerans = 0, bearbetningstyp: bortval CYCLE832, orienteringstolerans = 0 grader

b) Cykelanrop och bortval utan klartextinmatning

Programkod	Kommentar
G710	; S. O.
TRAORI	; S. O.
CYCLE832(0.3,13,0.8)	; S. O.
...	; S. O.
CYCLE832(0,0,1)	; S. O.

20.1.38 CYCLE840 - gängtappning med flytande gänghållare

Syntax

CYCLE840(<RTP>, <RFP>, <SDIS>, <DP>, <DPR>, <DTB>, <SDR>, <SDAC>, <ENC>, <MPIT>, <PIT>, <_AXN>, <_PITA>, <_TECHNO>, <_PITM>, <_PTAB>, <_PTABA>, <_GMODE>, <_DMODE>, <_AMODE>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse			
1	RP	<RTP>	REAL	Återgångsplan (abs)			
2	Z0	<RFP>	REAL	Referenspunkt (abs)			
3	SC	<SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)			
4	Z1	<DP>	REAL	Borrdjup (abs), se <_AMODE>			
5	Z1	<DPR>	REAL	Borrdjup (ink), se <_AMODE>			
6	DT	<DTB>	REAL	Fördröjningstid på borrdjup/på säkerhetsavstånd efter återgång i sekunder, se <ENC>			
7		<SDR>	INT	Rotationsriktning för återgången			
8	SDE	<SDAC>	INT	Rotationsriktning efter cykelslut			
9		<ENC>	INT	Gängtappning med spindelgivare (G33)/gängtappning utan spindelgivare (G63)			
					0 =	med spindelgivare	- Stigning ur <MPIT>/<PIT> - utan DT
					20 =	med spindelgivare	- Stigning ur <MPIT>/<PIT> - med DT efter återgång till säkerhetsavstånd
					11 =	utan spindelgivare	- Stigning ur <MPIT>/<PIT> - med DT på borrdjup
					1 =	utan spindelgivare	- Stigning ur programmerad matning - med DT på borrdjup (matning = varvtal · stigning)
10		<MPIT>	REAL	Gängstorlek endast för "ISO metrisk" (stigning beräknas internt under körtiden) Värdeområde: 3 till 48 (för M3 till M48), alternativt till <PIT>			
11		<PIT>	REAL	Gängstigning som värde, måttenhet se <_PITA> Värdeområde: > 0, alternativt till MPIT			
12		<_AXN>	INT	Borraxel	0 =	3:e geometriaxel	
					1 =	1:a geometriaxel	
					2 =	2:a geometriaxel	
					≥ 3 =	3:e geometriaxel	

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
13		<_PITA>	INT	Måttenhet för gängstignigen (urvärdering <PIT> och <MPIT>)
				0 = stigning i mm - Utvärdering <MPIT>/<PIT>
				1 = stigning i mm - Utvärdering <PIT>
				2 = stigning i TPI - Utvärdering <PIT> (antal gånger per inch)
				3 = stigning i inch - Utvärdering <PIT>
				4 = MODUL - Utvärdering <PIT>
14		<_TECHNO>	INT	Teknologi ¹⁾
				ENTAL:
				Precisionsstoppbeteende
				0 = precisionsstoppbeteende som före cykelanrop aktivt
				1 = precisionsstopp G601
				2 = precisionsstopp G602
				3 = precisionsstopp G603
				TOTAL:
				Förstyrning
				0 = med/utan förstyrning som före cykelanrop aktiv
				1 = med förstyrning FFWON
				2 = utan förstyrning FFWOF
15		<_PITM>	STRING[15]	String som flagga för gängstigningsinmatning ²⁾
16		<_PTAB>	STRING[5]	String för gängtabell ("", "ISO", "BSW", "BSP", "UNC") ²⁾
17		<_PTABA>	STRING[20]	String för val i gängtabellen (t.ex. "M 10", "M 12", ...) ²⁾
18		<_GMODE>	INT	reserverat

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
19		<_DMODE>	INT	Indikeringsmode		
				ENTAL:	Bearbetningsplan G17/G18/G19	
					0 =	kompatibilitet, det före cykelanrop verk-samma planet förblir aktivt
					1 =	G17 (endast aktiv i cykeln)
					2 =	G18 (endast aktiv i cykeln)
					3 =	G19 (endast aktiv i cykeln)
				TIOTAL:	reserverat	
				HUNDRATAL:	reserverat	
				TUSENTAL:	Kompatibilitetsmode (endast för inmatnings-mask till återöversättning), när MD 52216 Bit0 = 1 ¹⁾	
					0 =	teknologiparametrar visas (kompatibili-tet): TECHNO-parametrar verkar
					1 =	teknologiparametrar visas inte: Teknolo-gi "som programmerad före cykelanrop" verkar
				TIOTUSENTAL:	Teknologiskalning inom cykelmaskerna (Sida 814)	
					0 =	inmatning: komplett
1 =	inmatning: enkel					
20		<_AMODE>	INT	Alternativmode		
				ENTAL:	Borrdjup Z1 (abs/ink)	
					0 =	kompatibilitet, från programmering <DP>/<DPR>
					1 =	inkrementell
					2 =	absolut

¹⁾ Fälten Teknologi kan vara gömda beroende på settingdatum SD52216 MCS_FUNCTION_MASK_DRILL

²⁾ Parametrarna 15, 16 och 17 används endast vid val av gänga i gängtabellen för inmatningsmasken. En åtkomst på gäng-tabellerna via cykeldefinition vid cykelkörtid är inte möjligt!

20.1.39 CYCLE899 - fräsa öppet spår

Syntax

```
CYCLE899(<_RTP>, <_RFP>, <_SDIS>, <_DP>, <_LENG>, <_WID>, <_PA>,
<_PO>, <_STA>, <_MID>, <_MIDA>, <_FAL>, <_FALD>, <_FFP1>, <_CDIR>,
<_VARI>, <_GMODE>, <_DMODE>, <_AMODE>, <_UMODE>, <_FS>, <_ZFS>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	RP	<_RTP>	REAL	Återgångsplan (abs)
2	Z0	<_RFP>	REAL	Referenspunkt för verktygsaxeln (abs)
3	SC	<_SDIS>	REAL	Säkerhetsavstånd (additivt till referenspunkten, skall matas in utan förtecken)
4	Z1	<_DP>	REAL	Spårdjup (abs/ink), se <_AMODE>
5	L	<_LENG>	REAL	Längd för spåret (ink)
6	W	<_WID>	REAL	Bredd för spåret (ink)
7	X0	<_PA>	REAL	Referenspunkt/startposition 1:a axeln (abs)
8	Y0	<_PO>	REAL	Referenspunkt/startposition 2:a axeln (abs)
9	$\alpha 0$	<_STA>	REAL	Vridvinkel till 1:a axeln
10	DZ	<_MID>	REAL	Maximalt ansättningsdjup (ink), endast för virvelfräsning
11	DXY	<_MIDA>	REAL	Maximal ansättning i planet, se <_AMODE>
12	UXY	<_FAL>	REAL	Finbearbetningsmån i planet
13	UZ	<_FALD>	REAL	Finbearbetningsmån djup
14	F	<_FFP1>	REAL	Matning
15		<_CDIR>	INT	Fräsriktning
				ENTAL:
				0 = Medmatning
				1 = Motmatning
				4 = Växelvis
16		<_VARI>	INT	Bearbetning
				ENTAL:
				1 = grovbearbetning
				2 = finbearbetning
				3 = finbearbetning av botten
				4 = finbearbetning av kant
				5 = förfinbearbetning
				6 = fasning
				TIOTAL: reserverat
				HUNDRATAL: reserverat
				TUSENTAL:
				1 = Virvelfräsning
				2 = Stickfräsning

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
17		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL: reserverat
				TIOTAL: reserverat
				HUNDRATAL: Val bearbetning/enda startpunktberäkning
				1 = normal bearbetning
				TUSENTAL: Måtsättning via mitt / kant
				0 = måtsättning via mitt
				1 = måtsättning via kant "vänster" ("-" riktning för den 1:a axeln)
				2 = måtsättning via kant "höger" ("+" riktning för den 1:a axeln)
18		<_DMODE>	INT	Indikeringsmode
				ENTAL: Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL: --- reserverat
				HUNDRATAL: --- reserverat
				TUSENTAL: --- reserverat
				TIOTUSENTAL: Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel
19		<_AMODE>	INT	Alternativmode
				ENTAL: Spår djup Z1
				0 = absolut
				1 = inkrementell
				TIOTAL: Enhet för ansättning i planet (<_MIDA>) DXY
				0 = mm
				1 = % av verktygsdiameter
				HUNDRATAL: Nermatningsdjup vid fasning ZFS
				0 = absolut
				1 = inkrementell
20		<_UMODE>	INT	reserverat
21	FS	<_FS>	REAL	Fasbredd (ink)
22	ZFS	<_ZFS>	REAL	Nermatningsdjup (verktygsspets) vid fasning (abs/ink) se <_AMODE>

20.1.40 CYCLE930 - Instick

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse		
1	X0	<_SPD>	REAL	Referenspunkt i planaxeln (alltid diameter)		
2	Z0	<_SPL>	REAL	Referenspunkt i längsaxeln		
3	B1	<_WIDG>	REAL	Insticksbredd nere		
4	B2	<_WIDG2>	REAL	Insticksbredd uppe (endast för ytan)		
5	T1	<_DIAG>	REAL	Insticksbredd i referenspunkten, vid abs och längsbearbetning = diameter, annars ink		
6	T2	<_DIAG2>	REAL	Insticksdjup mittemot referenspunkten (endast för ytan), vid abs och längsbearbetning = diameter, annars ink		
7	α0	<_STA>	REAL	Vinkel för lutningen ($-180 \leq <_STA> \leq 180$)		
8	α1	<_ANG1>	REAL	Flankvinkel 1 ($0 \leq <_ANG1> < 90$) vid den av referenspunkten bestämda sidan av insticket		
9	α2	<_ANG2>	REAL	Flankvinkel 2 ($0 \leq <_ANG2> < 90$) mittemot referenspunkten		
10	R1/FS1	<_RCO1>	REAL	Rundningsradie eller fasbredd 1, på utsidan vid referenspunkten		
11	R2/FS2	<_RCI1>	REAL	Rundningsradie eller fasbredd 2, på insidan vid referenspunkten		
12	R3/FS3	<_RCI2>	REAL	Rundningsradie eller fasbredd 3, på insidan mittemot referenspunkten		
13	R4/FS4	<_RCO2>	REAL	Rundningsradie eller fasbredd 4, på utsidan mittemot referenspunkten		
14	U	<_FAL>	REAL	Finbearbetningsmån i X och Z, se <_VARI> (TIOTUSENTAL) (mata in utan förtecken)		
15	D	<_IDEP1>	REAL	Maximal djupansättning vid nermatning (mata in utan förtecken)		
					0 =	1:a skäret direkt på helt djup
					> 0 =	1:a skäret <_IDEP1>, 2:a skäret $2 \cdot <_IDEP1>$ osv.
16	SC	<_SDIS>	REAL	Säkerhetsavstånd (mata in utan förtecken)		

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
17		<_VARI>	INT	Bearbetningstyp
				ENTAL:
				reserverat
				TIOTAL:
				Teknologisk bearbetning
				1 = grovbearbetning
				2 = finbearbetning
				3 = grovbearbetning och finbearbetning
				HUNDRATAL:
				Läge längs/plan yttre/inre +Z/+Z resp. +X/-X
				1 = längs/yttre +Z
				2 = plan/inre -X
				3 = längs/inre +Z
				4 = plan/inre +X
				5 = längs/yttre -Z
				6 = plan/yttre -X
				7 = längs/inre -Z
				8 = plan/yttre +X
				TUSENTAL:
				Referenspunktens läge
				0 = Referenspunkt ovan
				1 = Referenspunkt nere
				TIOTUSENTAL:
				Definition verka som finbearbetningsmån
				0 = finbearbetningsmån U konturparallell
				1 = finbearbetningsmånerna UX och UZ åtskilda
18		<_DN>	INT	D-nummer för 2:a skäret till verktyget
				> 0 = D-nummer för verktygskompensering för det 2:a skäret till instickaren
				0 = inget 2:a skär programmerat
19	N	<_NUM>	INT	Antal instick (0 = 1 instick)
20	DP	<_DBH>	REAL	Avstånd för insticken (endast nödvändigt när <_NUM> > 1)
21	F	<_FF1>	REAL	Matning
22		<_NR>	INT	Kännetecken för insticksformen motsvarar de vertikala funktionstangenterna för val av formen
				0 = 90°-flanker utan faser/rundningar
				1 = sneda flanker med faser/rundningar (utan α0)
				2 = som 1, men på kon (med α0)
23	UX	<_FALX>	REAL	Finbearbetningsmån i X axeln, se <_VARI> (TIOTUSENTAL) (mata in utan förtecken)
24	UZ	<_FALZ>	REAL	Finbearbetningsmån i Z axeln, se <_VARI> (TIOTUSENTAL) (mata in utan förtecken)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
25		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
26		<_AMODE>	INT	Alternativmode
				ENTAL:
				Måttsättning djup (endast för ytan)
				0 = i referenspunkten
				1 = mittemot referenspunkten
				TIOTAL:
				Djup
				0 = absolut
				1 = inkrementell
				HUNDRATAL:
				Måttsättning bredd (endast för ytan)
				0 = på ytterdiametern (uppe)
				1 = på innerdiametern (nere)
				TUSENTAL:
				Radie/fas 1 (<_RCO1>)
				0 = Radie
				1 = Fas
				TIOTUSENTAL:
				Radie/fas 2 (<_RCI1>)
				0 = Radie
				1 = Fas
				HUNDRATUSENTAL:
				Radie/fas 3 (<_RCI2>)
				0 = Radie
				1 = Fas
				ENMILJONTAL:
				Radie/fas 4 (<_RCO2>)
				0 = Radie
				1 = Fas

20.1.41 CYCLE940 - fristick former

Med cykeln CYCLE940 kan olika fristick programmeras. Dessa skiljer sig till en del betydande i parameterinställningen.

De extra spalterna i tabellen visar vilken parameter som behövs vid vilken fristicksform. De motsvarar de vertikala funktionstangenterna i cykelmasken:

- E: Fristick form E
- F: Fristick form F
- A-D: Fristick gänga DIN (formerna A-D)
- T: Fristick gänga (fri definition av formen)

Syntax

CYCLE940 (<_SPD>, <_SPL>, <_FORM>, <_LAGE>, <_SDIS>, <_FFP>, <_VARI>,
<_EPD>, <_EPL>, <_R1>, <_R2>, <_STA>, <_VRT>, <_MID>, <_FAL>,
<_FALX>, <_FALZ>, <_PITI>, <_PTAB>, <_PTABA>, <_DMODE>, <_AMODE>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Progr. vid form				Betydelse
				E	F	A-D	T	
1	X0	<_SPD>	REAL	x	x	x	x	Referenspunkt i planaxeln (alltid diameter)
2	Z0	<_SPL>	REAL	x	x	x	x	Referenspunkt längsaxel (abs)
3	FORM	<_FORM>	CHAR	x	x	x	x	Form för fristicket (stora bokstäver, t.ex. "T") Val ur vilken tabellen fristicksvärdena ska tas
								A = yttre, referens DIN76, A = normal
								B = yttre, referens DIN76, B = kort
								C = inre, referens DIN76, C = normal
								D = inre, referens DIN76, D = kort
								E = referens DIN509
								F = referens DIN509
								T = friform
4	LAGE	<_LAGE>	INT	x	x	x	x	Läge för fristicket (parallell Z)
								0 = yttre +Z: ____
								1 = yttre -Z: ____/
								2 = inre +Z: /-----
								3 = inre -Z: -----\
5	SC	<_SDIS>		x	x	x	x	Säkerhetsavstånd (ink)
6	F	<_FFP>		x	x	x	x	Arbetsmatning (mm/varv)
7		<_VARI>	INT	-	-	x	x	Bearbetningstyp
								ENTAL: Bearbetning
								1 = grovbearbetning
								2 = finbearbetning
								3 = grovbearbetning + finbearbetning
								TIOTAL: Bearbetningsstrategi
								0 = konturparallell
								1 = längs
								Fristick form E och F bearbetas alltid i ett svep som finbearbetning.
8	X1	<_EPD>		x	x	-	-	Avmått X (abs/ink), se <_AMODE>
								- - - x Fristicksdjup (abs/ink), se <_AMODE>
								- x - - Avmått Z
								- - - x Fristicksbredd (abs/ink), se <_AMODE>

Nr	Parameter mask	Parameter intern	Datatyp	Progr. vid form				Betydelse			
10	R1	<_R1>		-	-	-	x	Rundningsradie vid avfasningar			
11	R2	<_R2>		-	-	-	x	Rundningsradie i hörnet			
12	α	<_STA>		-	-	x	x	Nermatningsvinkel			
13	VX	<_VRT>		x	x	-	-	Planrörelse X (abs/ink), se <_AMODE>			
				-	-	x	x	Planrörelse X vid finbearbetning, (abs/ink), se <_AMODE>			
14	D	<_MID>		-	-	x	x	Djupansättning			
15	U	<_FAL>		-	-	x	x	Finbearbetningsmån konturparallell, se <_AMODE>			
16	UX	<_FALX>		-	-	x	x	Finbearbetningsmån X			
17	UZ	<_FALZ>		-	-	x	x	Finbearbetningsmån Z			
18	P	<_PITI>	INT	-	-	x	-	Val av stigning, form A-D, motsvarar 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	-	-	Val radie/djup, form 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]					String för gängtabell ("", "ISO", "BSW", "BSP", "UNC") (endast för ytan)			
20		<_PTABA>	STRING [20]					String för val i gängtabellen (t.ex. "M 10", "M 12", ...) (endast för ytan)			
21		<_DMODE>	INT					Indikeringsmode			
				x	x	x	x	ENTAL:	Bearbetningsplan G17/G18/G19		
									0 =	kompatibilitet, det före cykelanrop verksamma planet förblir aktivt	
									1 =	G17 (endast aktiv i cykeln)	
									2 =	G18 (endast aktiv i cykeln)	
					3 =	G19 (endast aktiv i cykeln)					

Nr	Parameter mask	Parameter intern	Datatyp	Progr. vid form	Betydelse
22		<_AMODE>	INT		Alternativmode
				x x - x	ENTAL: Parameter <_EPD> avmått X eller fristicksdjup
					0 = absolut (alltid diameter)
					1 = inkrementell
				x x - x	TIOTAL: Parameter <_EPL> avmått Z eller fristicksbredd
					0 = absolut
					1 = inkrementell
				x x x x	HUNDRATAL: Parameter <_VRT> planrörelse X
					0 = absolut (alltid diameter)
					1 = inkrementell
				- - x x	TUSENTAL: Finbearbetningsmån
					0 = Finbearbetningsmån konturparallell (<_FAL>)
					1 = Finbearbetningsmån åtskild (<_FALX>/<_FALZ>)

20.1.42 CYCLE951- avspåna

Syntax

CYCLE951(<_SPD>, <_SPL>, <_EPD>, <_EPL>, <_ZPD>, <_ZPL>, <_LAGE>, <_MID>, <_FALX>, <_FALZ>, <_VARI>, <_RF1>, <_RF2>, <_RF3>, <_SDIS>, <_FF1>, <_NR>, <_DMODE>, <_AMODE>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	X0	<_SPD>	REAL	Referenspunkt (abs, alltid diameter)
2	Z0	<_SPL>	REAL	Referenspunkt (abs)
3	X1	<_EPD>	REAL	Slutpunkt
4	Z1	<_EPL>	REAL	Slutpunkt
5	XM α1 α2	<_ZPD>	REAL	Mellanpunkt, se <_DMODE> (TIOTAL)
6	ZM α1 α2	<_ZPL>	REAL	Mellanpunkt, se <_DMODE> (TIOTAL)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse			
7	Läge	<_LAGE>	INT	Läge för avspåningshörn	0 =	yttre/bak	
					1 =	yttre/fram	
					2 =	inre/bak	
					3 =	inre/fram	
8	D	<_MID>	REAL	Maximal djupansättning vid nermatning			
9	UX	<_FALX>	REAL	Finbearbetningsmån i X			
10	UZ	<_FALZ>	REAL	Finbearbetningsmån i Z			
11		<_VARI>	INT	Bearbetningstyp			
				ENTAL:	Avspåningsriktning (längs eller plan) i koordinatsystemet		
					1 =	längs	
					2 =	plan	
				TIOTAL:			
					1 =	grovbearbetning till finbearbetningsmån	
					2 =	finbearbetning	
				HUNDRATAL:		reserverat	
				TUSENTAL:		reserverat	
TIOTUSENTAL:		reserverat					
12	R1/FS1	<_RF1>	REAL	Rundningsradie eller fasbredd 1, se <_AMODE> (TIOTUSENTAL)			
13	R2/FS2	<_RF2>	REAL	Rundningsradie eller fasbredd 2, se <_AMODE> (HUNDRATUSENTAL)			
14	R3/FS3	<_RF3>	REAL	Rundningsradie eller fasbredd 3, se <_AMODE> (ENMILJONTAL)			
15	SC	<_SDIS>	REAL	Säkerhetsavstånd			
16	F	<_FF1>	REAL	Matning för grovbearbetning/finbearbetning			
17		<_NR>	INT	Kännetecken för avspåningstypen (motsvarar de vertikala funktions-tangenterna för val av formen):			
					0 =	avspåning 1, 90 grader hörn utan faser/rundningar	
					1 =	avspåning 2, 90 grader hörn med faser/rundningar	
					2 =	avspåning 3, valfritt hörn med faser/rundningar	

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
18		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL:
				Inmatningsform <_ZPD>/<_ZPL>
				0 = Xm/Zm
				1 = Xm/ α 1
				2 = Xm/ α 2
				3 = α 1/Zm
				4 = α 2/Zm
				5 = α 1/ α 2
21		<_AMODE>	INT	Alternativmode
				ENTAL:
				Mellangepunkt i X
				0 = absolut, värde för planaxeln i diametern
				1 = inkrementell, värde för planaxeln i radien
				TIOTAL:
				Mellangepunkt i Z
				0 = absolut
				1 = inkrementell
				HUNDRATAL:
				Ändpunkt i X
				0 = absolut, värde för planaxeln i diametern
				1 = inkrementell, värde för planaxeln i radien
				TUSENTAL:
				Ändpunkt i Z
				0 = absolut
				1 = inkrementell
				TIOTUSENTAL:
				Radie/fas 1
				0 = Radie
				1 = Fas
				HUNDRATUSENTAL:
				Radie/fas 2
				0 = Radie
				1 = Fas
				ENMILJONTAL:
				Radie/fas 3
				0 = Radie
				1 = Fas

20.1.43 CYCLE952 - kontursticka

Syntax

```
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>)
```

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1	PRG	<_PRG>	STRING[100]	Namn för avspänningsprogrammet
2	CON	<_CON>	STRING[100]	Programmets namn från vilket den aktualiserade råämneskonturen läses (vid restbearbetning)
3	CONR	<_CONR>	STRING[100]	Programmets namn i vilket den aktualiserade råämneskonturen skrivs (se <_AMODE> TIOTUSENTAL)

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
4		<_VARI>	INT	Bearbetningstyp
				ENTAL:
				Avspåningstyp
				1 = längs
				2 = plan
				3 = konturparallell
				TIOTAL:
				Teknologisk bearbetning (se <_GMODE> HUNDRATAL)
				1 = grovbearbetning
				2 = finbearbetning
				3 = reserverat
				4 = grovbearbetning tvåkanalig
				5 = finbearbetning tvåkanalig
				HUNDRATAL:
				Bearbetningsriktning
				1 = Bearbetningsriktning X -
				2 = Bearbetningsriktning X +
				3 = Bearbetningsriktning Z -
				4 = Bearbetningsriktning Z +
				TUSENTAL:
				Ansättningsriktning
				1 = yttre X -
				2 = inre X +
				3 = framsida Z -
				4 = baksida Z +
				TIOTUSENTAL:
				Definition, hur finbearbetningsmån verkar
				0 = finbearbetningsmånerna UX och UZ åtskilda
				1 = finbearbetningsmån U konturparallell
				HUNDRATUSENTAL:
				Efterdragning
				0 = kompatibilitet, efterdragning automatisk
				1 = med efterdragning vid konturen
				2 = utan efterdragning
				3 = efterdragning automatisk
				ENMILJONTAL:
				Baksnitt
				0 = plats utvärderas inte vid stickning, - rest och sticksvarva, - rest
				1 = Bearbeta baksnitt
				2 = Inte bearbeta baksnitt
				TIOMILJONTAL:
				bakom/framför rotationscentrum
				0 = bearbetning framför rotationscentrum
				1 = reserverat

Nr	Parameter mask	Parameter intern	Datotyp	Betydelse
5	F	<_F>	REAL	Matning för grovbearbetning/finbearbetning
	FZ			Matning abskissa sticksvärning
6	FR	<_FR>	REAL	Matning för nermatning i baksnitt grovbearbetning
	FX			Matning ordinata sticksvärning
7	RP	<_RP>	REAL	Återgångsplan vid innerbearbetning (abs., alltid diameter)
8	D	<_D>	REAL	Ansättning grovbearbetning (se <_AMODE> ENTAL)
9	DX	<_DX>	REAL	Ansättning X (se <_AMODE> ENTAL)
10	DZ	<_DZ>	REAL	Ansättning Z (se <_AMODE> ENTAL)
11	UX	<_UX>	REAL	Finbearbetningsmån X (se <_VARI> TIOTUSENTAL)
12	UZ	<_UZ>	REAL	Finbearbetningsmån Z (se <_VARI> TIOTUSENTAL)
13	U	<_U>	REAL	Finbearbetningsmån konturparallell (se <_VARI> TIOTUSENTAL)
14	U1	<_U1>	REAL	Extra finbearbetningsmån vid finbearbetning (se <_AMODE> TUSENTAL)
15	BL	<_BL>	INT	Råämnesdefinition
				1 = cylinder med avmätt
				2 = avmätt på färdigdetaljkontur
				3 = råämneskontur är angiven
16	XD	<_XD>	REAL	Råämnesdefinition X (se <_AMODE> HUNDRATUSENTAL)
17	ZD	<_ZD>	REAL	Råämnesdefinition Z (se <_AMODE> ENMILJONTAL)
18	XA	<_XA>	REAL	Gräns 1 X (abs., alltid diameter)
19	ZA	<_ZA>	REAL	Gräns 1 Z (abs.)
20	XB	<_XB>	REAL	Gräns 2 X (se <_AMODE> TIOMILJONTAL)
21	ZB	<_ZB>	REAL	Gräns 2 Z (se <_AMODE> HUNDRAMILJONTAL)
22	XDA	<_XDA>	REAL	Insticksgräns 1 för den 1:a instickspositionen på framsidan (abs., alltid diameter)
23	XDB	<_XDB>	REAL	Insticksgräns 2 för den 1:a instickspositionen på framsidan (abs., alltid diameter)
24	N	<_N>	INT	Antal instick
25	DP	<_DP>	REAL	Avstånd för insticken
				Längs-instick: Parallellt till Z-axeln Plan-instick: Parallellt till X-axeln
26	DI	<_DI>	REAL	Avstånd för matningsstopp
				0 = inget avbrott > 0 = med avbrott
27	SC	<_SC>	REAL	Säkerhetsavstånd för körning runt hinder, inkrementalt
28	D2	<_DN>	INT	D-nummer för 2:a skäret, om inte programmerat ⇒ D+1

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
29		<_GMODE>	INT	Geometrimode (värdering av de programmerade geometrivärdena)
				ENTAL:
				re-serve-rat
				TIOTAL:
				re-serve-rat
				HUNDRATAL:
				Val bearbetning/enda startpunktberäkning
				0 = normal bearbetning (ingen kompatibilitetsmode nödvändig)
				1 = normal bearbetning
				2 = beräkna startposition - ingen bearbetning (endast för anrop från ShopMill/ShopTurn)
				TUSENTAL:
				Begränsning
				0 = nej
				1 = ja
				TIOTUSENTAL:
				Mata in gräns 1 X
				0 = nej
				1 = ja
				HUNDRATUSENTAL:
				Mata in gräns 2 X
				0 = nej
				1 = ja
				ENMILJONTAL:
				Mata in gräns 1 Z
				0 = nej
				1 = ja
				TIOMILJONTAL:
				Mata in gräns 2 Z
				0 = nej
				1 = ja

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
30		<_DMODE>	INT	Indikeringsmode
				ENTAL:
				Bearbetningsplan G17/G18/G19
				0 = kompatibilitet, det före cykelanrop verksamma planet förblir aktivt
				1 = G17 (endast aktiv i cykeln)
				2 = G18 (endast aktiv i cykeln)
				3 = G19 (endast aktiv i cykeln)
				TIOTAL:
				Teknologimodus
				1 = Konturavspåning
				2 = Kontursticka
				3 = Stickvarva
				HUNDRATAL:
				Bearbeta restmaterial
				0 = nej
				1 = ja
				TUSENTAL:

				reserverat
				TIOTUSENTAL:
				Teknologiskalning inom cykelmaskerna (Sida 814)
				0 = inmatning: komplett
				1 = inmatning: enkel

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
31		<_AMODE>	INT	Alternativmode
				ENTAL:
				Val ansättning
				0 = ansättning DX och DZ vid avspåningstyp konturparallell
				1 = ansättning D
				TIOTAL:
				Ansättningsstrategi
				0 = variabelt skärdjup (90 ... 100 %)
				1 = konstant snittdjup
				HUNDRATAL:
				Snittdelning
				0 = likformig
				1 = uppriktning vid kanter
				TUSENTAL:
				Val konturavmätt U1, dubbel finbearbetning
				0 = nej
				1 = ja
				TIOTUSENTAL:
				Aktualisera val råämne
				0 = nej
				1 = ja
				HUNDRATUSENTAL:
				Val avmätt på råämne XD
				0 = absolut, värde för planaxeln i diametern
				1 = inkrementell, värde för planaxeln i radien
				ENMILJONTAL:
				Val avmätt på råämne ZD
				0 = absolut
				1 = inkrementell
				TIOMILJONTAL:
				Val gräns 2 XB
				0 = absolut, värde för planaxeln i diametern
				1 = inkrementell, värde för planaxeln i radien
				HUNDRAMILJONTAL:
				Val gräns 2 ZB
				0 = absolut
				1 = inkrementell
				ENMILJARDTAL
				0 = styrkanal
				1 = följekanal
32		<_PK>	INT	Nummer för partnerkanalen när det finns fler än 2 kanaler på maskinen
33	DCH	<_DCH>	REAL	Kanalförskjutning
34	FS	<_FS>	REAL	Matning finbearbetning bei Komplettbearbetning

20.1.44 CYCLE4071 - längsslipning med ansättning i vändpunkten

Syntax

```
CYCLE4071(<S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>, <S_H>, <S_A1>, <S_A2>)
```

Parameter

Nr	Parameter	Datotyp	Betydelse
1	<S_A>	REAL	Anställningsdjup i början
2	<S_B>	REAL	Anställningsdjup i slutet
3	<S_W>	REAL	Slipbredd
4	<S_U>	REAL	Renbränningstid
5	<S_I>	REAL	Matning för ansättning
6	<S_K>	REAL	Matning för tväransättning
7	<S_H>	INT	Antal upprepningar
8	<S_A1>	AXIS	Ansättningsaxel (optional) resp. 1:a geometriaxeln
9	<S_A2>	AXIS	Pendelaxel (optional) resp. 2:a geometriaxeln

Funktion

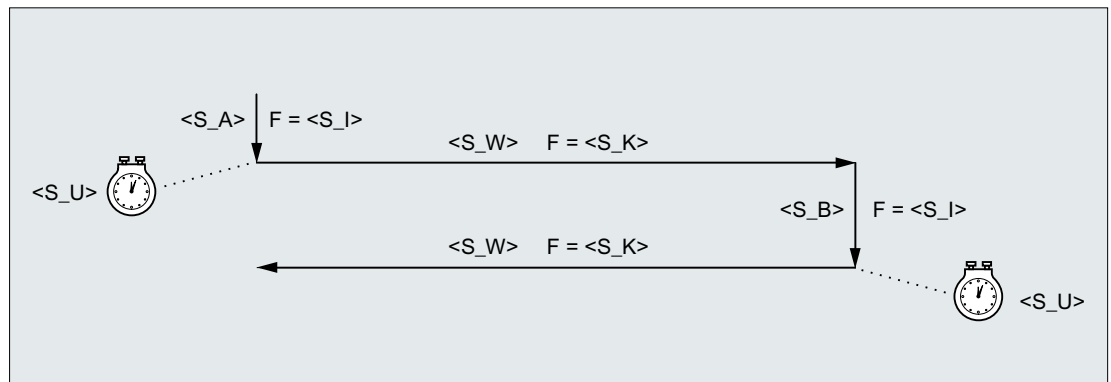
Cykeln tjänar till genomarbetning av <S_H> ansättningar. Därvid kan ansättningsdjupet i början och i slutet vara olika. Mellan ansättningarna görs en tangentiell rörelse.

Förlopp

1. Start av cykeln vid den aktuella positionen för pendelaxeln.
2. Förflyttning av ansättningsaxeln till anställningsdjupet vid början P1 <S_A> med matningen för ansättning P5 <S_I>.
3. Renbränna med renbränningstid P4 <S_U>.
4. Förflyttning av pendelaxeln med slipbredd P3 <S_W> som förflyttningsväg och matningen för tväransättning P6 <S_K>.
5. Förflyttning av ansättningsaxeln till anställningsdjupet vid slutet P2 <S_B> med matningen för ansättning P5 <S_I>.
6. Renbränna med renbränningstid P4 <S_U>.
7. Förflyttning av pendelaxeln med slipbredd P4 <S_A> som förflyttningsväg på begynnelsepunkten och matningen för tväransättning P6 <S_K>.

Förloppet kan inte stoppas med enkelblock.

Förloppet upprepas motsvarande det programmerade antalet upprepningar P7 (<S_H>).



Exempel

Utföra två pendelrörelser med följande cykelparametrar:

- Anställningsdjup i början: 0,02 mm
- Anställningsdjup i slutet: 0,01 mm
- Slag: 100 mm
- Renbränningstid: 1 s
- Matning ansättning: 1 mm/min
- Tvärmattning: 1000 mm/min
- Upprepningar: 2
- Pendel- och ansättningsaxel: Standardgeometriaxlar

Programkod

```
N10 T1 D1
N20 CYCLE4071 (0.02,0.01,100,1,1,1000,2)
N30 M30
```

20.1.45 CYCLE4072 - längsslipning mit ansättning i vändpunkten

Syntax

```
CYCLE4072 (<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_I>, <S_K>,
<S_H>, <S_A1>, <S_A2>)
```

Parameter

Nr	Parameter	Datatyp	Betydelse
1	<S_GAUGE>	STRING	Avbrottsvillkor för ansättning: 1. Nummer för en snabb ingång 2. Logiskt uttryck
2	<S_A>	REAL	Anställningsdjup i början
3	<S_B>	REAL	Anställningsdjup i slutet
4	<S_W>	REAL	Slipbredd
5	<S_U>	REAL	Renbränningstid
6	<S_I>	REAL	Matning för ansättning
7	<S_K>	REAL	Matning för tväransättning
8	<S_H>	INT	Antal upprepningar
9	<S_A1>	AXIS	Ansättningsaxel (optional) resp. 1:a geometriaxeln
10	<S_A2>	AXIS	Pendelaxel (optional) resp. 2:a geometriaxeln

Funktion

Cykeln tjänar till genomarbetning av <S_H> ansättningar under hänsynstagande till en extern avbrottsignal. Ansättningsdjupet kan vara olika i början och i slutet. Mellan ansättningarna görs en tangentiell rörelse. Djupansättningen avbryts, när avbrottsvillkoret är uppfyllt. Efter avbrottet av djupansättningen genomförs alltid ett komplett slag.

Förlopp

1. Start av cykeln vid den aktuella positionen för pendelaxeln.
2. Förflyttning av ansättningsaxeln till anställningsdjupet vid början P2 <S_A> med matningen för ansättning P6 <S_I>.
3. Renbränna med renbränningstid P5 <S_U>.
4. Förflyttning av pendelaxeln med slipbredd P4 <S_W> som förflyttningsväg och matningen för tväransättning P7 <S_K>.
5. Förflyttning av ansättningsaxeln till anställningsdjupet vid slutet P3 <S_B> med matningen för ansättning P6 <S_I>.
6. Renbränna med renbränningstid P5 <S_U>.
7. Förflyttning av pendelaxeln med slipbredd P4 <S_W> som förflyttningsväg på begynnelsepunkten och matningen för tväransättning P7 <S_K>.
8. Utan avbrott: Det ovan beskrivna förloppet upprepas så ofta tills det programmerade antalet upprepningar P7 (<S_H>) har uppnåtts.
Med avbrott: Bearbetningen avslutad när nästa startpunkt uppnåtts.

Förloppet kan inte stoppas med enkelblock.

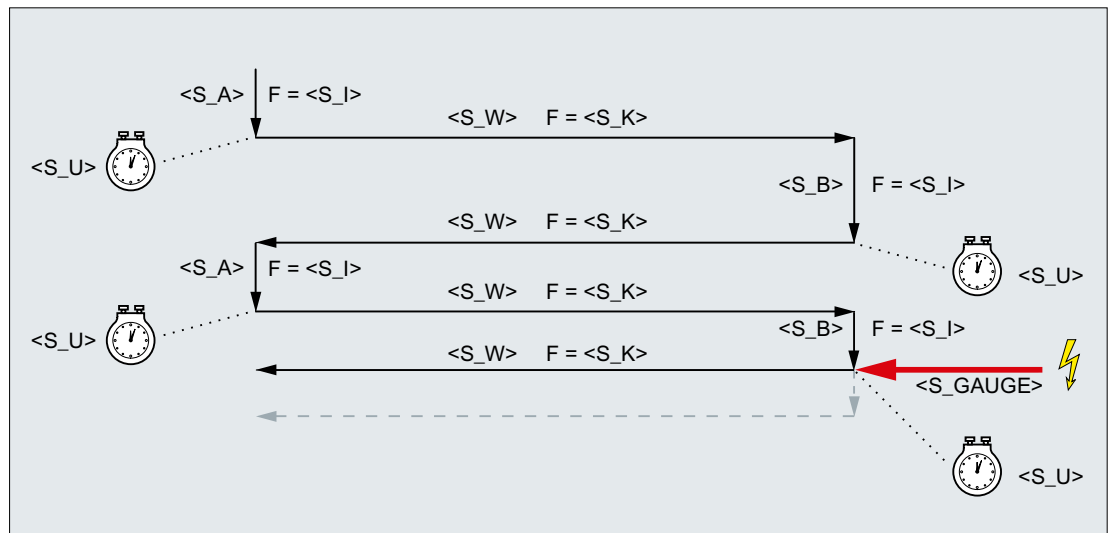


Bild 20-1 Avbrott av ansättningen i slutet

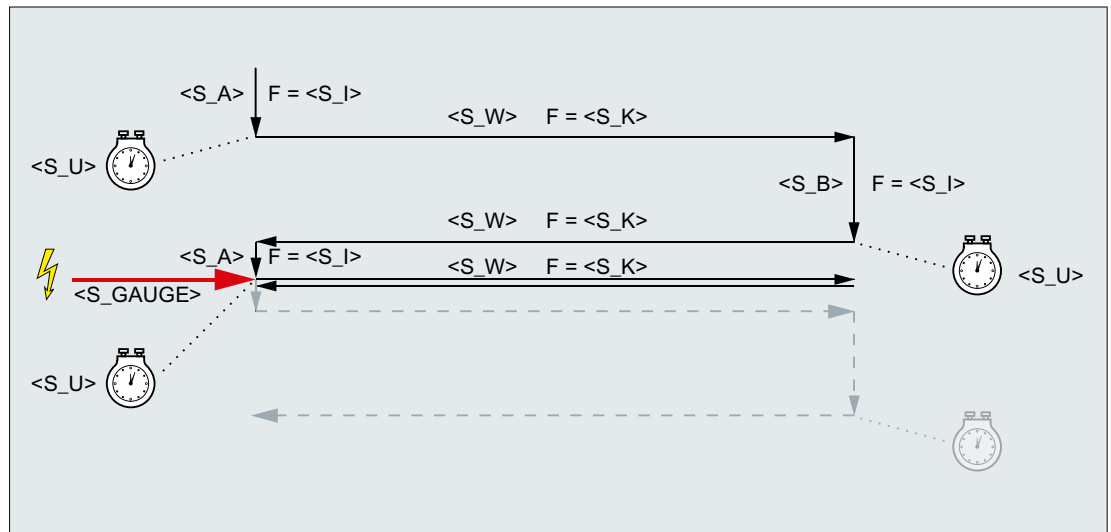


Bild 20-2 Avbrott av ansättningen i början

Resurser

Som resurser använder cykeln en blockövergripande synkronaktionsvariabel. Synkronaktionen fastställs dynamiskt från det fria området i synkronaktionsförbandet (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR - 1199 ...). Som synkronaktionsvariabel används SYG_IS[1].

Exempel

Exempel 1: Oscillering med två slag

Cykelparameter:

- Anställningsdjup i början: 0,02 mm
- Anställningsdjup i slutet: 0,01 mm
- Slag: 100 mm
- Renbränningstid: 1 s
- Matning ansättning: 1 mm/min
- Tvärmätning: 1000 mm/min
- Upprepningar: 2
- Pendel- och ansättningsaxel: Standardgeometriaxlar

Avbrottssignal: snabb ingång 1 (\$A_IN[1])

Programkod

```
N10 T1 D1
N20 CYCLE4072 ("1",0.02,0.01,100,1,1,1000,2)
N30 M30
```

Exempel 2: Oscillering med två slag

Cykelparameter:

- Anställningsdjup i början: 0,02 mm
- Anställningsdjup i slutet: 0,01 mm
- Slag: 100 mm
- Renbränningstid: 1 s
- Matning ansättning: 1 mm/min
- Tvärmätning: 1000 mm/min
- Upprepningar: 2
- Pendel- och ansättningsaxel: Standardgeometriaxlar

Avbrottssignal: Variable \$A_DBR[20] < 0,01

Programkod

```
N10 T1 D1
N20 CYCLE4072 ("($A_DBR[20]<0.01)",0.02,0.01,100,1,1,1000,2)
N30 M30
```

20.1.46 CYCLE4073 - längsslipning med kontinuerlig ansättning

Syntax

CYCLE4073 (<S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>, <S_A1>, <S_A2>)

Parameter

Nr	Parameter	Datatyp	Betydelse
1	<S_A>	REAL	Anställningsdjup i början
2	<S_B>	REAL	Anställningsdjup i slutet
3	<S_W>	REAL	Slipbredd
4	<S_U>	REAL	Renbränningstid
5	<S_K>	REAL	Matning för tväransättning
6	<S_H>	INT	Antal upprepningar
7	<S_A1>	AXIS	Ansättningsaxel (optional) resp. 1:a geometriaxeln
8	<S_A2>	AXIS	Pendelaxel (optional) resp. 2:a geometriaxeln

Funktion

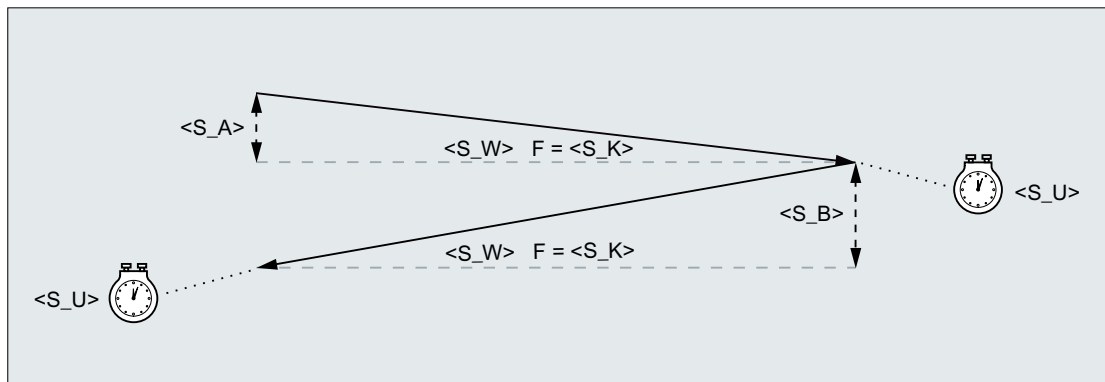
Cykeln tjänar till genomarbetning av <S_H> ansättningar. Därvid kan ansättningen vara olika från början till slutet eller från slutet till början.

Förlopp

1. Start av cykeln vid den aktuella positionen för pendelaxeln med ansättningsdjup 0
2. Förflyttning av pendelaxeln med slipbredd P3 <S_W> som förflyttningsväg och matningen för tväransättning P5 <S_K> med kontinuerlig tillväxt av ansättningsdjupet till ansättningsdjupet vid början P1 <S_A> .
3. Renbränna med renbränningstid P4 <S_U>.
4. Förflyttning av pendelaxeln med slipbredd P3 <S_W> som förflyttningsväg på begynnelsepunkten och matningen för tväransättning P5 <S_K> med kontinuerlig tillväxt av ansättningsdjupet till ansättningsdjupet vid slutet P2 <S_B> .
5. Renbränna med renbränningstid P4 <S_U>.

Förloppet kan inte stoppas med enkelblock.

Förloppet upprepas motsvarande det programmerade antalet upprepningar P7 (<S_H>).



Exempel

Oscillering med två slag

Cykelparameter:

- Anställningsdjup i början: 0,02 mm
- Anställningsdjup i slutet: 0,01 mm
- Slag: 100 mm
- Renbränningstid: 1 s
- Tvärrätning: 1000 mm/min
- Upprepningar: 2
- Pendel- och ansättningsaxel: Standardgeometriaxlar

Programkod

```
N10 T1 D1
N20 CYCLE4073(0.02,0.01,100,1,1000,2)
N30 M30
```

20.1.47 CYCLE4074 - Längssllipning med kontinuerlig ansättning och avbrottssignal

Syntax

```
CYCLE4074(<S_GAUGE>, <S_A>, <S_B>, <S_W>, <S_U>, <S_K>, <S_H>,
<S_A1>, <S_A2>)
```

Parameter

Nr	Parameter	Datatyp	Betydelse
1	<S_GAUGE>	STRING	Avbrottsvillkor för ansättning: 1. Nummer för en snabb ingång 2. Logiskt uttryck
2	<S_A>	REAL	Anställningsdjup i början
3	<S_B>	REAL	Anställningsdjup i slutet
4	<S_W>	REAL	Slipbredd
5	<S_U>	REAL	Renbränningstid
6	<S_K>	REAL	Matning för tväransättning
7	<S_H>	INT	Antal upprepningar
8	<S_A1>	AXIS	Ansättningsaxel (optional) resp. 1:a geometriaxeln
9	<S_A2>	AXIS	Pendelaxel (optional) resp. 2:a geometriaxeln

Funktion

Cykeln tjänar till genomarbetning av <S_H> ansättningar under hänsynstagande till t. ex. en extern avbrottssignal. Ansättningsdjupet kan vara olika i början och i slutet. Djupansättningen avbryts, när avbrottsvillkoret är uppfyllt. Efter avbrottet av djupansättningen genomförs alltid ett komplett slag.

Förlopp

1. Start av cykeln vid den aktuella positionen för pendelaxeln med ansättningsdjup 0
2. Förflyttning av pendelaxeln med slipbredd P4 <S_W> som förflyttningsväg och matningen för tväransättning P6 <S_K> med kontinuerlig tillväxt av ansättningsdjupet till ansättningsdjupet vid början P2 <S_A> .
3. Renbränna med renbränningstid P5 <S_U>.
4. Förflyttning av pendelaxeln med slipbredd P4 <S_W> som förflyttningsväg på begynnelsepunkten och matningen för tväransättning P6 <S_K> med kontinuerlig tillväxt av ansättningsdjupet till ansättningsdjupet vid slutet P3 <S_B> .
5. Renbränna med renbränningstid P5 <S_U>.
6. Utan avbrott: Det ovan beskrivna förloppet upprepas så ofta tills det programmerade antalet upprepningar P7 (<S_H>) har uppnåtts.
Med avbrott: Djupansättningen avbryts. Bearbetningen avslutad när nästa startpunkt uppnåtts.

Förloppet kan inte stoppas med enkelblock.

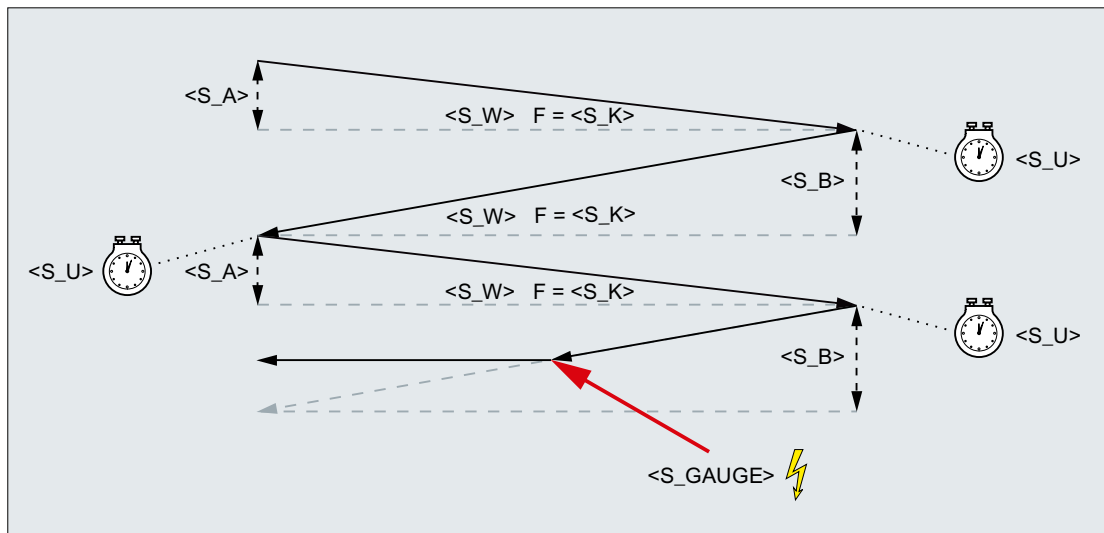


Bild 20-3 Avbrott av ansättningen från slutet till början

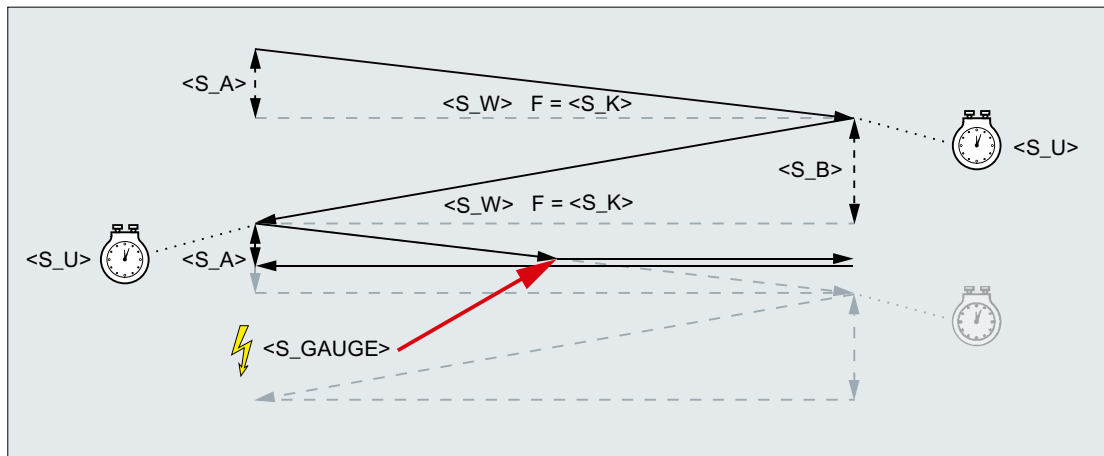


Bild 20-4 Avbrott av ansättningen från början till slutet

Resurser

Som resurser använder cykeln en blockövergripande synkronaktionsvariabel. Synkronaktionen fastställs dynamiskt från det fria området i synkronaktionsförbandet (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR - 1199 ...). Som synkronaktionsvariabel används SYG_IS[1].

Exempel

Exempel 1: Oscillering med två slag

Cykelparameter:

- Anställningsdjup i början: 0,02 mm
- Anställningsdjup i slutet: 0,01 mm

- Slag: 100 mm
- Renbränningstid: 1 s
- Tvärmätning: 1000 mm/min
- Upprepningar: 2
- Pendel- och ansättningsaxel: Standardgeometriaxlar

Avbrottssignal: snabb ingång 1 (\$A_IN[1])

Programkod

```
N10 T1 D1
N20 CYCLE4074 ("1", 0.02, 0.01, 100, 1, 1000, 2)
N30 M30
```

Exempel 2: Oscillering med två slag

Cykelparameter:

- Anställningsdjup i början: 0,02 mm
- Anställningsdjup i slutet: 0,01 mm
- Slag: 100 mm
- Renbränningstid: 1 s
- Tvärmätning: 1000 mm/min
- Upprepningar: 2
- Pendel- och ansättningsaxel: Standardgeometriaxlar

Avbrottssignal: Variable \$A_DBR[20] < 0,01

Programkod

```
N10 T1 D1
N20 CYCLE4074 ("($A_DBR[20]<0.01)", 0.02, 0.01, 100, 1, 1000, 2)
N30 M30
```

20.1.48 CYCLE4075 - flatslipning med ansättning i vändpunkten

Syntax

```
CYCLE4075(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>,
<S_A2>)
```

Parameter

Nr	Parameter	Datotyp	Betydelse
1	<S_I>	REAL	Anställningsdjup i början
2	<S_J>	REAL	Anställningsdjup i slutet

Nr	Parameter	Datatyp	Betydelse
3	<S_K>	REAL	Hela ansättningsdjupet
4	<S_A>	REAL	Slipbredd
5	<S_R>	REAL	Matning för ansättning
6	<S_F>	REAL	Matning för tväransättning
7	<S_P>	REAL	Renbränningstid
8	<S_A1>	AXIS	Ansättningsaxel (optional)
9	<S_A2>	AXIS	Pendelaxel (optional)

Funktion

Cykeln tjänar till bearbetning med ett totalt ansättningsdjup P3 <S_K> i ansättningssteg. Anställningsdjupen i början P1 <S_I> och i slutet P2 <S_J> kan vara olika. Mellan ansättningsarna görs en tangentiell rörelse.

Vägguppgifterna P1 till P4 kan vara negativa eller positiva.

Uppgiften för ansättningsaxeln P8 <S_A1> och/eller pendelaxeln P9 <S_A2> är tillval. Blir en resp. båda parametrarna inte angivna, så använder cykeln de första båda geometriaxlarna i kanalen.

Är summan av ansättningsdjup i början P1 <S_I> och i slutet P2 <S_J> lika med 0 resp. det totala ansättningsdjupet P3 <S_K> lika med 0 görs endast ett renbränningsslag.

Förlopp

1. Start av cykeln vid den aktuella positionen för pendelaxeln.
2. Förflyttning av ansättningsaxeln till anställningsdjupet vid början P1 <S_I> med matningen för ansättning P5 <S_R>.
3. Renbränna med renbränningstid P7 <S_P>.
4. Förflyttning av pendelaxeln med slipbredd P4 <S_A> som förflyttningsväg och matningen för tväransättning P6 <S_F>.
5. Förflyttning av ansättningsaxeln till anställningsdjupet vid slutet P2 <S_J> med matningen för ansättning P5 <S_R>.
6. Renbränna med renbränningstid P7 <S_P>.
7. Förflyttning av pendelaxeln med slipbredd P4 <S_A> som förflyttningsväg på begynnelsepunkten och matningen för tväransättning P6 <S_F>.

Förloppet kan inte stoppas med enkelblock.

Förloppet upprepas så ofta tills det totala ansättningsdjupet P3 (<S_K>) har uppnåtts. Det sista slaget delas sedan upp olikformigt.

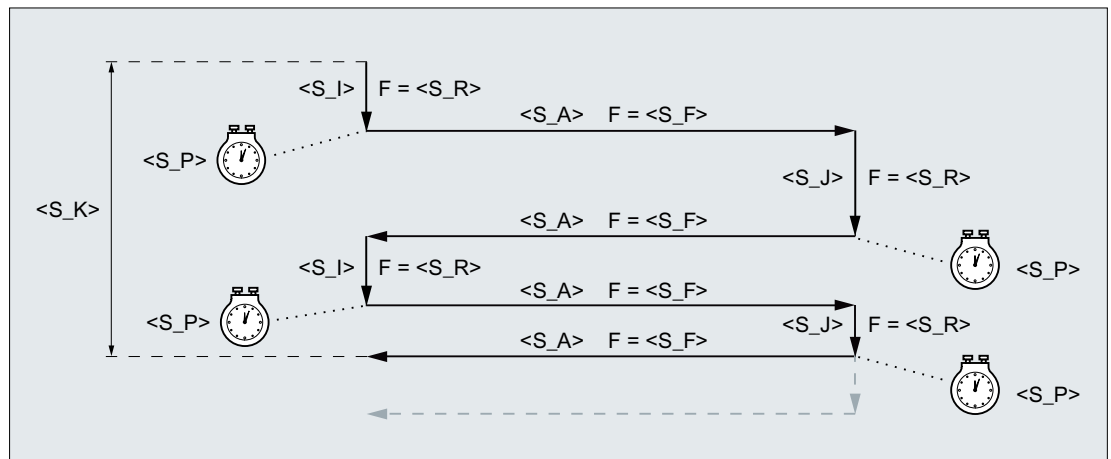


Bild 20-5 Hela anställningsdjupet uppnått vid ansättningen i andra vändpunkten

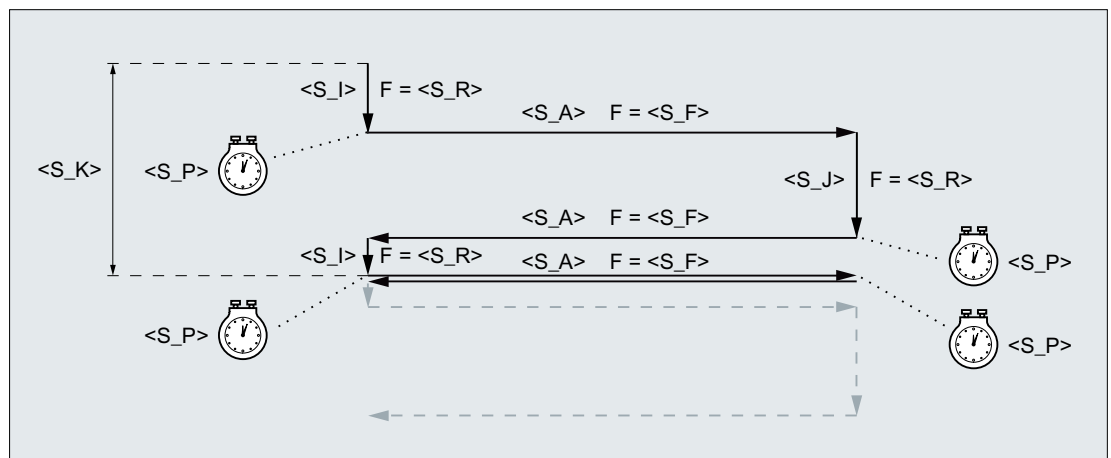


Bild 20-6 Hela anställningsdjupet uppnått vid ansättningen i första vändpunkten

Exempel

Oscillering med:

- 0,02 mm anställningsdjup i början
- 0,01 mm anställningsdjup i slutet
- Hela anställningsdjupet 1 mm
- Slag 100 mm
- Matning ansättning 1 mm/min
- Tvärmatning 1000 mm/min
- Renbränningstid 1 sekund
- Standardgeometriaxlar

Programkod

N10 T1 D1

Programkod

```
N20 CYCLE4075(0.02,0.01,1,100,1,1000,1)
N30 M30
```

20.1.49 CYCLE4077 - flatslipning med ansättning i vändpunkten och avbrottssignal

Syntax

```
CYCLE4077(<S_GAUGE>, <S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>,
<S_P>, <S_A1>, <S_A2>)
```

Parameter

Nr	Parameter	Datotyp	Betydelse
1	<S_GAUGE>	STRING	Avbrottsvillkor för ansättning: • Nummer för en snabb ingång • Logiskt uttryck
2	<S_I>	REAL	Anställningsdjup i början
3	<S_J>	REAL	Anställningsdjup i slutet
4	<S_K>	REAL	Hela ansättningsdjupet
5	<S_A>	REAL	Slipbredd
6	<S_R>	REAL	Matning för ansättning
7	<S_F>	REAL	Matning för tväransättning
8	<S_P>	REAL	Renbränningstid
9	<S_A1>	AXIS	Ansättningsaxel (optional)
10	<S_A2>	AXIS	Pendelaxel (optional)

Funktion

Cykeln tjänar till bearbetning med ett totalt ansättningsdjup P4 <S_K> i ansättningssteg. Ansättningsdjupen i början P2 <S_I> och i slutet P3 <S_J> kan vara olika. Mellan ansättningarna görs en tangentiell rörelse. Djupansättningen avbryts, när avbrottssignalen för den snabba ingången lika med 1 eller avbrottsvillkoret är uppfyllt. Efter avbrottet genomförs ett komplett slag.

Väguppgifterna P2 till P5 kan vara negativa eller positiva.

Uppgiften för ansättningsaxeln P9 <S_A1> och/eller pendelaxeln P10 <S_A2> är tillval. Blir en resp. båda parametrarna inte angivna, så använder cykeln de första båda geometriaxlarna i kanalen.

Är summan av ansättningsdjup i början P2 <S_I> och i slutet P3 <S_J> lika med 0 resp. det totala ansättningsdjupet P4 <S_K> lika med 0 görs endast ett renbränningslag.

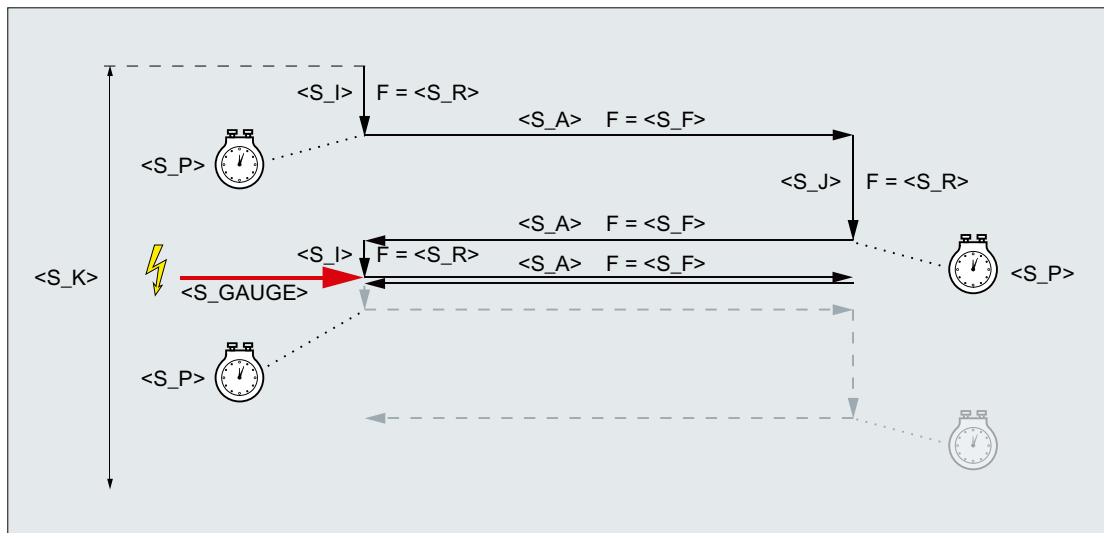


Bild 20-8 Avbrott av ansättningen i början

Resurser

Som resurser använder cykeln en blockövergripande synkronaktionsvariabel. Synkronaktionen fastställs dynamiskt från det fria området i synkronaktionsförbandet (CUS.DIR - 1 ..., CMA.DIR - 1000 ..., CST.DIR - 1199 ...). Som synkronaktionsvariabel används SYG_IS[1].

Exempel

Exempel 1

Oscillering med:

- 0,02 mm anställningsdjup i början
- 0,01 mm anställningsdjup i slutet
- Hela ansättningsdjupet 1 mm
- Slag 100 mm
- Matning ansättning 1 mm/min
- Tvärmattning 1000 mm/min
- Renbränningstid 1 sekund
- Standardgeometriaxlar

Avbrottssignal: snabb ingång 1 (\$A_IN[1])

Programkod

```
N10 T1 D1
N20 CYCLE4077("1",0.02,0.01,1,100,1,1000,1)
N30 M30
```

Exempel 2

Oscillering med:

- 0,02 mm anställningsdjup i början
- 0,01 mm anställningsdjup i slutet
- Hela ansättningsdjupet 1 mm
- Slag 100 mm
- Matning ansättning 1 mm/min
- Tvärmätning 1000 mm/min
- Renbränningstid 1 sekund
- Standardgeometriaxlar

Avbrottssignal: Dualport-RAM-variabel 20 mindre än 0,01 (\$A_DBR[20] < 0,01)

Programkod
N10 T1 D1
N20 CYCLE4077("(\$A_DBR[20]<0.01)",0.02,0.01,1,100,1,1000,1)
N30 M30

20.1.50 CYCLE4078 - flatsllipning med kontinuerlig ansättning**Syntax**

CYCLE4078 (<S_I>, <S_J>, <S_K>, <S_A>, <S_F>, <S_P>, <S_A1>, <S_A2>)

Parameter

Nr	Parameter	Datatyp	Betydelse
1	<S_I>	REAL	Ansättningsdjup från början till slut
2	<S_J>	REAL	Ansättningsdjup från slut till början
3	<S_K>	REAL	Hela ansättningsdjupet
4	<S_A>	REAL	Slipbredd
5	<S_F>	REAL	Matning
6	<S_P>	REAL	Renbränningstid
7	<S_A1>	AXIS	Ansättningsaxel (optional)
8	<S_A2>	AXIS	Pendelaxel (optional)

Funktion

Cykeln tjänar till bearbetning med ett totalt ansättningsdjup P3 <S_K> med hjälp av kontinuerlig ansättning. Ansättningsdjupen från början till slut P1 <S_I> och från slut till början P2 <S_J> kan vara olika.

Väguppgifterna P1 till P4 kan vara negativa eller positiva.

Uppgiften för ansättningsaxeln P8 <S_A1> och/eller pendelaxeln P9 <S_A2> är tillval. Blir en resp. båda parametrarna inte angivna, så använder cykeln de första båda geometriaxlarna i kanalen.

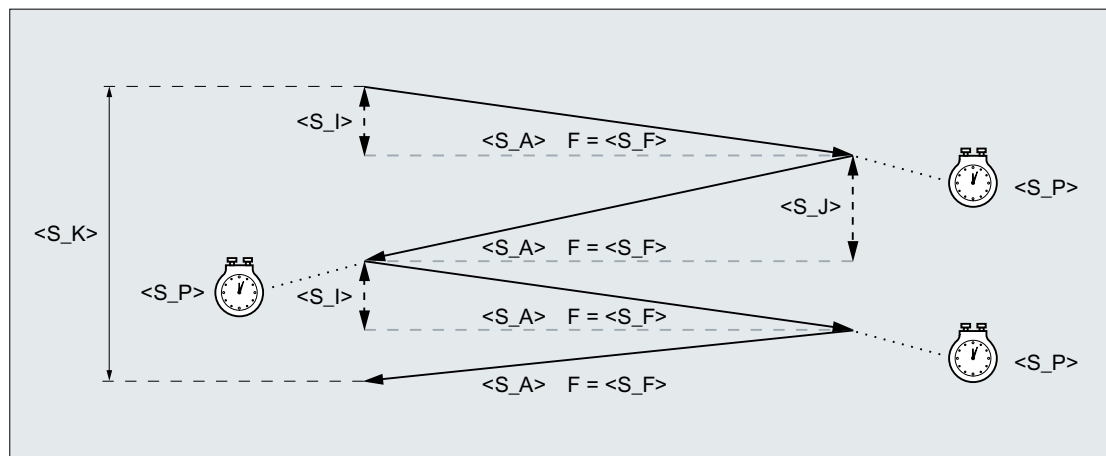
Är summan av ansättningsdjupen P1 <S_I> och P2 <S_J> lika med 0 resp. det totala ansättningsdjupet P3 <S_K> lika med 0 görs endast ett renbränningsslag.

Förlopp

1. Start av cykeln vid den aktuella positionen för pendelaxeln med ansättningsdjup 0
2. Förflyttning av pendelaxeln med slipbredd P4 <S_A> som förflyttningsväg och matningen P5 <S_F> med kontinuerlig tillväxt av ansättningsdjupet till ansättningsdjupet i början P1 <S_I>.
3. Renbränna med renbränningstid P7 <S_P>.
4. Förflyttning av pendelaxeln med slipbredd P4 <S_A> som förflyttningsväg på begynnelsepunkten och matningen P5 <S_F> med kontinuerlig tillväxt av ansättningsdjupet till ansättningsdjupet vid slutet P2 <S_J>.
5. Renbränna med renbränningstid P7 <S_P>.
6. Förflyttning av pendelaxeln med slipbredd P4 <S_A> som förflyttningsväg på begynnelsepunkten och matning P5 <S_F>.

Förloppet kan inte stoppas med enkelblock.

Förloppet upprepas så ofta tills det totala ansättningsdjupet P3 (<S_K>) har uppnåtts. Det sista slaget delas sedan upp olikformigt.



Exempel

Oscillering med:

- 20 mm anställningsdjup i början
- 10 mm anställningsdjup i slutet
- Hela ansättningsdjupet 100 mm
- Slag 100 mm

- Matning 1000 mm/min
- Renbränningstid 1 sekund
- Standardgeometriaxlar

Programkod
N10 T1 D1
N20 CYCLE4078(20,10,100,100,1000,1)
N30 M30

20.1.51 CYCLE4079 - flatslipning med intermittent ansättning

Syntax

CYCLE4079(<S_I>, <S_J>, <S_K>, <S_A>, <S_R>, <S_F>, <S_P>, <S_A1>, <S_A2>)

Parameter

Nr	Parameter	Datotyp	Betydelse
1	<S_I>	REAL	Anställningsdjup i början
2	<S_J>	REAL	Anställningsdjup i slutet
3	<S_K>	REAL	Hela ansättningsdjupet
4	<S_A>	REAL	Slipbredd
5	<S_R>	REAL	Matning för ansättning
6	<S_F>	REAL	Matning för tväransättning
7	<S_P>	REAL	Renbränningstid
8	<S_A1>	AXIS	Ansättningsaxel (optional)
9	<S_A2>	AXIS	Pendelaxel (optional)

Funktion

Cykeln tjänar till bearbetning med ett totalt ansättningsdjup P3 <S_K> i ansättningssteg. Anställningsdjupen i början P1 <S_I> och i slutet P2 <S_J> kan vara olika. Mellan ansättningarna görs en tangentiell rörelse.

Väguppgifterna P1 till P4 kan vara negativa eller positiva.

Uppgiften för ansättningsaxeln P8 <S_A1> och/eller pendelaxeln P9 <S_A2> är tillval. Blir en resp. båda parametrarna inte angivna, så använder cykeln de första båda geometriaxlarna i kanalen.

Är summan av ansättningsdjup i början P1 <S_I> och i slutet P2 <S_J> lika med 0 resp. det totala ansättningsdjupet P3 <S_K> lika med 0 görs endast ett renbränningsslag.

Förlopp

1. Start av cykeln vid den aktuella positionen för pendelaxeln.
2. Förflyttning av ansättningsaxeln till anställningsdjupet vid början P1 <S_I> med matningen för ansättning P5 <S_R>.
3. Renbränna med renbränningstid P7 <S_P>.
4. Förflyttning av pendelaxeln med slipbredd P4 <S_A> som förflyttningsväg och matningen för tväransättning P6 <S_F>.
5. Förflyttning av ansättningsaxeln till anställningsdjupet vid slutet P2 <S_J> med matningen för ansättning P5 <S_R>.
6. Renbränna med renbränningstid P7 <S_P>.
7. Förflyttning av pendelaxeln med slipbredd P4 <S_A> som förflyttningsväg på begynnelsepunkten och matningen för tväransättning P6 <S_F>.

Förloppet kan inte stoppas med enkelblock.

Förloppet upprepas så ofta tills det totala ansättningsdjupet P3 (<S_K>) har uppnåtts. Det sista slaget delas sedan upp olikformigt.

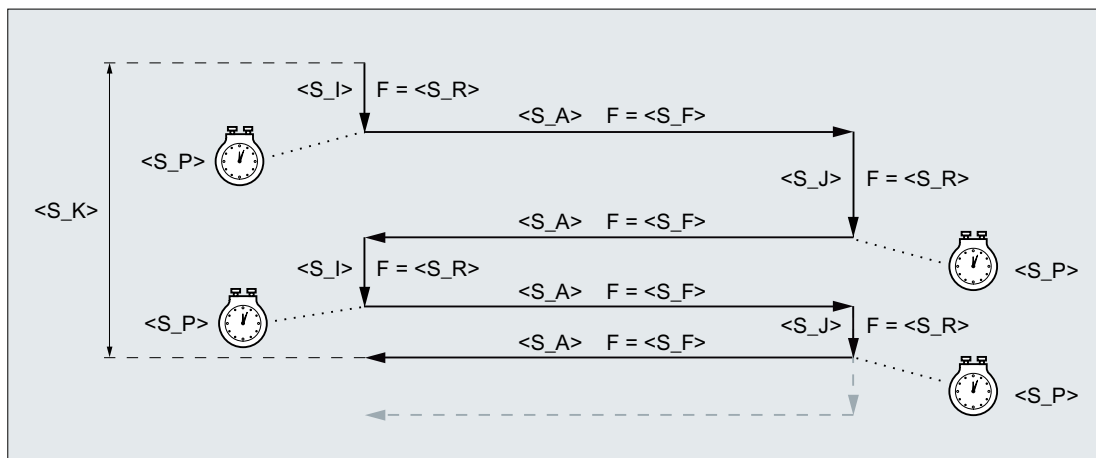


Bild 20-9 Hela ansättningsdjupet uppnått vid ansättningen i andra vändpunkten

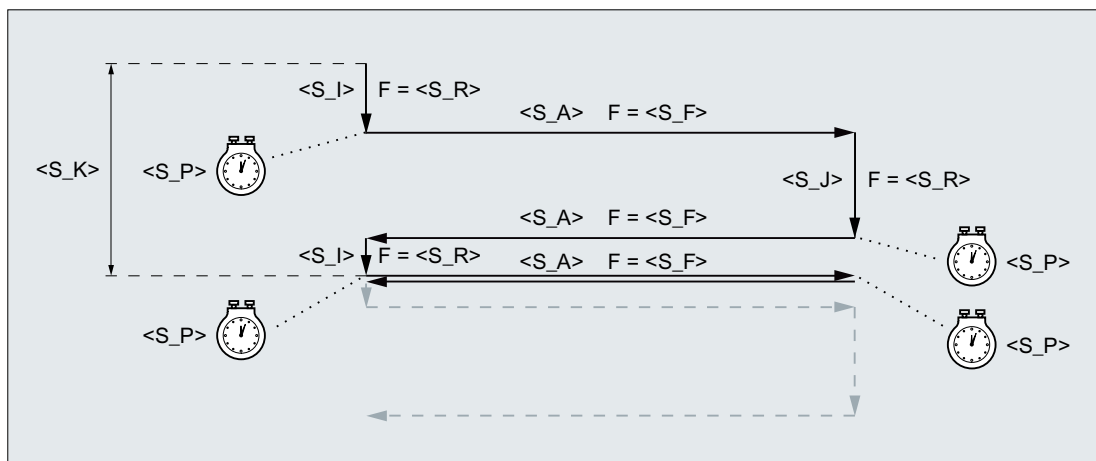


Bild 20-10 Hela ansättningsdjupet uppnått vid ansättningen i första vändpunkten

Exempel

Oscillering med:

- 0,02 mm anställningsdjup i början
- 0,01 mm anställningsdjup i slutet
- Hela ansättningsdjupet 1 mm
- Slag 100 mm
- Matning ansättning 1 mm/min
- Tvärmätning 1000 mm/min
- Renbränningstid 1 sekund
- Standardgeometriaxlar

Programkod
N10 T1 D1
N20 CYCLE4079(0.02,0.01,1,100,1,1000,1)
N30 M30

20.1.52 GROUP_BEGIN - början programblock

Syntax

GROUP_BEGIN(<_LEVEL>, <_NAME>, <_SP>, <_MODE>, <S_ICON>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1		<_LEVEL>	INT	Plan
				0 = Huvudnivå
				1 = 1:a underplanet
2		<_NAME>	STRING[128]	Blocknamn
3		<_SP>	INT	Spindel
				0 = ingen spindel
				1 = Huvudspindel
				2 = Motspindel
4		<_MODE>	INT	Mode
				Bit 0 = 1 GROUP_ADDEND existerar
				Bit 1 = 1 ShopTurn: Automatisk återgång (fara till verktygsväxlingspunkt)
				Bit 12 reserverat
				Bit 13 reserverat
5		<S_ICON>	STRING[32]	Namn för ikonerna (endast för användargränssnittet)

20.1.53 GROUP_END - slut programblock

Syntax

GROUP_END(<_LEVEL>, <_SP>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1		<_LEVEL>	INT	Plan
				0 = Huvudnivå
				1 = 1:a underplanet
2		<_SP>	INT	Spindel
				0 = ingen spindel
				1 = Huvudspindel
				2 = Motspindel

20.1.54 GROUP_ADDEND - slut inkörningstillägg

Syntax

GROUP_ADDEND(<_LEVEL>, <_SP>)

Parametrar

Nr	Parameter mask	Parameter intern	Datatyp	Betydelse
1		<_LEVEL>	INT	Plan
				0 = Huvudnivå
				1 = 1:a underplanet
2		<_SP>	INT	Spindel
				0 = ingen spindel
				1 = Huvudspindel
				2 = Motspindel

20.1.55 Randvillkor

20.1.55.1 Teknologiskalning inom cykelmaskerna

Är teknologiskalningen aktiv, kan vid olika cykelmasker den förenklade inmatningen väljas, i vilken endast de viktigaste cykelparametrarna visas.

Vid följande cykelmasker kan t.ex. den förenklade inmatningen väljas:

Teknologi	Cykelmask
Borra	Djuphålsborra
	Gängtappning
Fräsa	Fyrkantficka
	Konturfräsning: Ficka
Svarva	Gängsvarvning: Längs
	Kontursvarva: Avspåning
	Kontursvarva: Sticka
	Kontursvarva: Sticksvarva

Vid de ifrågavarande cykelmaskerna står på operatörsgränssnitt valmöjligheterna "Inmatning: enkel" och "Inmatning: **komplett**" till förfogande.

Ej visade cykelparametrar

De vid den förenklade inmatningen ej visade cykelparametrarna förbeläggs antingen med fasta, teknologiskt meningsfulla men ej förändringsbara värden. Eller cykelparametrarna tillordnas parametrerbara värden via kanalspecifika cykler-settingdata. Se under avsnittet "Idrifttagning" > "Kanalspecifika cykler-settingdata"

Omkoppling "Inmatning: komplett" > "Inmatning: enkel"

Blir en cykelmask ifylld i under inställningen "Inmatning komplett" och sedan omkopplad till "Inmatning enkel", används vid genereringen av cykelanropet för de då inte längre visade parametrarna, default- eller settingdatavärden.

Idrifttagning

Kanalspecifika konfigurationsmaskindata

Med maskindatum aktiveras möjligheten till teknologiskalning inom cykelmaskerna:

MD52210 \$MCS_FUNCTION_MASK_DISP, Bit 9 = 1 (visa val "Inmatning enkel")

Kanalspecifika cykler-settingdata

Är den förenklade inmatningen inom cykelmaskerna aktiv, kan värdena för vissa cykelparametrar föreskrivas via följande settingdata:

Nummer	Beteckning	Betydelse
SD55300	\$SCS_EASY_SAFETY_CLEARANCE	Säkerhetsavstånd
SD55301	\$SCS_EASY_DWELL_TIME	Fördröjningstid
SD55305	\$SCS_EASY_DRILL_DEEP_FD1	Djuphålsborra: Procentsats: 1:a matning
SD55306	\$SCS_EASY_DRILL_DEEP_DF	Djuphålsborra: Procentsats: Ansättning
SD55307	\$SCS_EASY_DRILL_DEEP_V1	Djuphålsborra: Minimal djupansättning
SD55308	\$SCS_EASY_DRILL_DEEP_V2	Djuphålsborra: Återgångsvärde
SD55309	\$SCS_EASY_THREAD_RETURN_DIST	Gängsvarvning: Återgångsavstånd

20.2 Mätcykler

Mätcykler är speciella av Siemens iordningställda underprogram för lösning av vissa mätuppgifter. Som i allmänhet vanligt för cykler, kan också mätcyklerna anpassas via parametrar till det konkreta problemet.

Mätcykler står för mätningar inom följande områden och teknologier till förfogande:

- Verktygmätningar svarva / fräsa
- Verktygmätningar svarva / fräsa

Litteratur

Den utförliga beskrivningen av mätcyklerna finns i:

Programmeringshandbok Mätcykler

Tabeller

21.1 Anvisningar

Märk Cykler

Listan över användningar innehåller alla cykler, som kan förekomma i NC-programmet (G-kod), dvs. kan programmeras i programeditorn via masker resp. måste programmeras vid slipning utan programstöd. Ingen hänsyn togs till cykler, som av kompatibilitetsskäl fortfarande finns i styrningen, men inte längre kan redigeras via programeditorn av SINUMERIK Operate ("kompatibilitetscykler").

Anvisningar A ... C

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
:	O	NC-huvudblocknummer, hoppmärkes-slut, länkningsoperator		+		PGAsl
*	O	Operator för multiplikation		+		PGAsl
+	O	Operator för addition		+		PGAsl
-	O	Operator för subtraktion		+		PGAsl
<	O	Jämförelseoperator, mindre än		+		PGAsl
<<	O	Länkningsoperator för strings		+		PGAsl
<=	O	Jämförelseoperator, mindre än lika med		+		PGAsl
=	O	Tilldelningsoperator		+		PGAsl
>=	O	Jämförelseoperator, större än lika med		+		PGAsl
/	O	Operator för division		+		PGAsl
/0		Block annulleras (1:a annulleringsnivån) ^o		+		PGsl
...		...				
...		...				
/7		Block annulleras (8:e annulleringsnivån)				
A	A	Axelnamn	m/s	+		PGAsl
A2	A	Verktygsorientering: RPY- eller Eulervinkel	s	+		PGAsl
A3	A	Verktygsorientering: 1:a komponent till rikt-ningsvektorn	s	+		PGAsl
A4	A	Verktygsorientering: 1:a komponent till ytnor-malvektorn i blockbörjan	s	+		PGAsl
A5	A	Verktygsorientering: 1:a komponent till ytnor-malvektorn i blockslutet	s	+		PGAsl
A6	A	Verktygsorientering: 1:a komponent till rikt-ningsvektorn för konens vridaxel	s	+		PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
A7	A	Verktygsorientering: 1:a vektorkomponent för mellanorienteringen på konens mantelyta	s	+		PGAsl
ABS	F	Absolutvärde (belopp)		+	+	PGAsl
AC	K	Absolut måttuppgift för koordinater/positioner	s	+		PGsl
ACC	K	Påverkan av den aktuella axiala accelerationen	m	+	+	PGsl
ACCLIMA	K	Påverkan av den aktuella maximala axiala accelerationen	m	+	+	PGAsl
ACN	K	Absolut måttuppgift för roterande axlar, uppsöka position i negativ riktning	s	+		PGsl
ACOS	F	Arcus-cosinus (trigon. funktion)		+	+	PGAsl
ACP	K	Absolut måttuppgift för roterande axlar, uppsöka position i positiv riktning	s	+		PGsl
ACTBLOCNO	P	Utmatning av det aktuella blocknumret för ett larmblock, även när "Aktuell blockindikering undertryckt" (DISPLOF) är aktiv!		+		PGAsl
ADDFRAME	F	Inräkning och ev. aktivering av en uppmätt frame		+	-	PGAsl, FB1sl (K2)
ADIS	A	Översläpningsavstånd för banfunktionerna G1, G2, G3, ...	m	+		PGsl
ADISPOS	A	Översläpningsavstånd för snabbgång G0	m	+		PGsl
ADISPOSA	P	Storlek på toleransfönstret för IPOBRKA	m	+	+	PGAsl
ALF	A	Snabblyftningsvinkel	m	+		PGAsl
AMIRROR	G	Programmerbar spegling	s	+		PGsl
AND	K	Logiskt OCH		+		PGAsl
ANG	A	Konturtågs-vinkel	s	+		PGsl
AP	A	Polarvinkel	m/s	+		PGsl
APR	K	Läsa / visa åtkomstskydd		+		PGAsl
APRB	K	Läsa åtkomstskydd, BTSS		+		PGAsl
APRP	K	Läsa åtkomstskydd, detaljprogram		+		PGAsl
APW	K	Skriva åtkomstskydd		+		PGAsl
APWB	K	Skriva åtkomstskydd, BTSS		+		PGAsl
APWP	K	Skriva åtkomstskydd, detaljprogram		+		PGAsl
APX	K	Definition av åtkomstskyddet för utförandet av det angivna språkelementet		+		PGAsl
AR	A	Öppningsvinkel	m/s	+		PGsl
AROT	G	Programmerbar vridning	s	+		PGsl
AROTS	G	Programmerbara framevridningar med rymd-vinklar	s	+		PGsl
AS	K	Makro-definition		+		PGAsl
ASCALE	G	Programmerbar skalning	s	+		PGsl
ASIN	F	Räknefunktion, arcussinus		+	+	PGAsl
ASPLINE	G	Akima-spline	m	+		PGAsl

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
ATAN2	F	Arcus-tangens2		+	+	PGAsl
ATOL	A	Axelspecifik tolerans för kompressor-funktioner, orienteringsjämning och översläpningstyper	m	+		PGAsl
ATRANS	G	Additiv programmerbar nollpunktsförskjutning	s	+		PGsl
AUXFUDEL	P	Radera hjälpfunktion kanalspecifikt från den globala listan		+	-	FB1sl (H2)
AUXFUDELG	P	Radera alla hjälpfunktioner i en hjälpfunktionsgrupp kanalspecifikt från den globala listan		+	-	FB1sl (H2)
AUXFUMSEQ	P	Fastställa ordningsföljden för utmatning av M-hjälpfunktioner		+	-	FB1sl (H2)
AUXFUSYNC	P	Från den globala listan över hjälpfunktioner generera ett fullständigt detaljprogramblock för den kanalspecifika SERUPRO-Ende-ASUP som string		+	-	FB1sl (H2)
AX	K	Variabel axelbeteckning	m/s	+		PGAsl
AXCTSWE	P	Vrida axelcontainer		+	-	PGAsl
AXCTSWEC	P	Återkalla frigivning av axelcontainer-vridning		+	+	PGAsl
AXCTSWED	P	Vrida axelcontainer (kommandovariant för idrifttagningen!)		+	-	PGAsl
AXIS	K	Axelbeteckning, axeladress		+		PGAsl
AXNAME	F	Konverterar ingångsstring till axelbeteckning		+	-	PGAsl
AXSTRING	F	Konverterar stringen spindelnummer		+	-	PGAsl
AXTOCHAN	P	Begära axel för en bestämd kanal. Är möjligt från NC-program och synkronaktion.		+	+	PGAsl
AXTOSPI	F	Konverterar axelbeteckning till ett spindelindex		+	-	PGAsl
B	A	Axelnamn	m/s	+		PGAsl
B2	A	Verktysorientering: RPY- eller Eulervinkel	s	+		PGAsl
B3	A	Verktysorientering: komponent till riktningsvektorn	s	+		PGAsl
B4	A	Verktysorientering: 2:a komponent till ytnormalvektorn i blockbörjan	s	+		PGAsl
B5	A	Verktysorientering: 2:a komponent till ytnormalvektorn i blockslutet	s	+		PGAsl
B6	A	Verktysorientering: 2:a komponent till riktningsvektorn för konens vridaxel	s	+		PGAsl
B7	A	Verktysorientering: 2:a vektorkomponent för mellanorienteringen på konens mantelyta	s	+		PGAsl
B_AND	O	Bitvis OCH		+		PGAsl
B_OR	O	Bitvis ELLER		+		PGAsl
B_NOT	O	Bitvis negering		+		PGAsl
B_XOR	O	Bitvis exklusivt-ELLER		+		PGAsl
BAUTO	G	Definiering av det första spline-avsnittet genom de efterföljande 3 punkterna	m	+		PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
BLOCK	K	Definierar tillsammans med nyckelordet TO den programdel som ska genomarbetas i en indirekt underprogramkörning		+		PGAsI
BLSYNC	K	Bearbetningen av interruptrutinen ska börja först med nästa blockbyte		+		PGAsI
BNAT ⁶⁾	G	Naturlig övergång till första spline-blocket	m	+		PGAsI
BOOL	K	Datotyp: Sanningsvärden TRUE/FALSE resp. 1/0		+		PGAsI
BOUND	F	Kontrollerar om värde ligger inom det definierade värdeområdet. Likhet returnerar kontrollvärde.		+	+	PGAsI
BRISK ⁶⁾	G	Hoppformig banacceleration	m	+		PGAsI
BRISKA	P	Koppla till språngartad banacceleration för de programmerade axlarna		+	-	PGAsI
BSPLINE	G	B-spline	m	+		PGAsI
BTAN	G	Tangentiell övergång till första spline-blocket	m	+		PGAsI
C	A	Axelnamn	m/s	+		PGAsI
C2	A	Verktygsorientering: RPY- eller Eulervinkel	s	+		PGAsI
C3	A	Verktygsorientering: 3:e komponent till rikt-ningsvektorn	s	+		PGAsI
C4	A	Verktygsorientering: 3:e komponent till ytnormalvektorn i blockbörjan	s	+		PGAsI
C5	A	Verktygsorientering: 3:e komponent till ytnormalvektorn i blockslutet	s	+		PGAsI
C6	A	Verktygsorientering: 3:e komponent till rikt-ningsvektorn för konens vridaxel	s	+		PGAsI
C7	A	Verktygsorientering: 3:e vektorkomponent för mellanorienteringen på konens mantelyta	s	+		PGAsI
CAC	K	Absolut uppsökning av en position		+		PGAsI
CACN	K	I tabell lagrat värde uppsöks absolut i negativ riktning		+		PGAsI
CACP	K	I tabell lagrat värde uppsöks absolut i positiv riktning		+		PGAsI
CALCDAT	F	Beräknar radie och medelpunkt för en cirkel ur 3 eller 4 punkter		+	-	PGAsI
CALCPOSI	F	Kontroll av kränkning av skyddsområden, arbetsfältsbegränsning och softwarelimits		+	-	PGAsI
CALL	K	Indirekt underprogramanrop		+		PGAsI
CALLPATH	P	Programmerbar sökväg vid underprogramanrop		+	-	PGAsI
CANCEL	P	Avbryta modal synkronaktion		+	-	FBSYsI
CASE	K	Villkorlig programförgrening		+		PGAsI
CDC	K	Direkt uppsökning av en position		+		PGAsI
CDOF ⁶⁾	G	Koppla från kollisionsovervakning	m	+		PGsI
CDOF2	G	Koppla från kollisionsovervakning, vid 3D-periferifräsning	m	+		PGsI

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
CDON	G	Koppla till kollisionsovervakning	m	+		PGsl
CFC ⁶⁾	G	Konstant matning vid konturen	m	+		PGsl
CFIN	G	Konstant matning endast vid innerböjning, inte vid ytterböjning	m	+		PGsl
CFINE	F	Tilldelning av fin-förflyttningen till en FRAME-variabel		+	-	PGAsl
CFTCP	G	Konstant matning i verktygsskär-referenspunkten, medelpunktsbanan	m	+		PGsl
CHAN	K	Specifisering av giltighetsområdet för data		+		PGAsl
CHANDATA	P	Ställa in kanalnummer för åtkomst av kanal-data		+	-	PGAsl
CHAR	K	Datatyp: ASCII-tecken		+		PGAsl
CHF	A	Fas; värde = längd på fasen	s	+		PGsl
CHKDM	F	Kontroll av entydigheten inom ett magasin		+	-	FBWsl
CHKDNO	F	Entydighetskontroll av D-numren		+	-	PGAsl
CHR	A	Fas; värde = längd på fasen i rörelseriktningen		+		PGsl
CIC	K	Inkrementell uppsökning av en position		+		PGAsl
CIP	G	Cirkelinterpolering via mellanpunkt	m	+		PGsl
CLEARM	P	Återställa en/ flera märken för kanalkoordinering		+	+	PGAsl
CLRINT	P	Välja Interrupt		+	-	PGAsl
CMIRROR	F	Spegling i en koordinataxel		+	-	PGAsl
COARSEA	K	Rörelseslut när "Grovt precisionsstopp" uppnås	m	+		PGAsl
COLLPAIR	F	Kontroll av tillhörighet till ett kollisionsspar		+		PGAsl
COMPCAD	G	Koppl till kompressor-funktion COMPCAD	m	+		PGAsl
COMPCURV	G	Koppla till kompressor-funktion COMPCURV	m	+		PGAsl
COMPLETE		Styranvisning för ut- och inläsning av data		+		PGAsl
COMPOF ⁶⁾	G	Koppla från NC-block-kompression	m	+		PGAsl
COMPON	G	Koppla till kompressor-funktion COMPON	m	+		PGAsl
COMPSURF	G	Koppla till kompressor-funktion COMSURF	m	+		PGAsl
CONTDCON	P	Koppla till konturavkodning i tabellform		+	-	PGAsl
CONTPRON	P	Koppla till referensförberedelse		+	-	PGAsl
CORROF	P	Alla aktiva överlagrade rörelser väljs bort.		+	-	PGsl
CORRTAFO	F	Modifiera offsetvektorer eller riktningssvektorer till orienteringsaxlarna i den kinematiska modellen för maskinen		+	-	PGAsl
COS	F	Cosinus (trigon. funktion)		+	+	PGAsl
COUPDEF	P	Definition ELG-förband / synkronspindel-förband		+	-	PGAsl
COUPDEL	P	Radera ELG-förband		+	-	PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
COUPOF	P	Koppla från ELG-förband / synkronspindelpar		+	-	PGAsl
COUPOFS	P	Koppla från ELG-förband / synkronspindelpar med stopp för följdspindel		+	-	PGAsl
COUPON	P	Koppla till ELG-förband / synkronspindelpar		+	-	PGAsl
COUPONC	P	Koppla till ELG-förband / synkronspindelpar överta med föregående programmering		+	-	PGAsl
COUPRES	P	Återställa ELG-förband		+	-	PGAsl
CP ⁶⁾	G	Banrörelse	m	+		PGAsl
CPBC	K	Generisk koppling: Blockbyteskriterium		+	+	FB3sl (M3)
CPDEF	K	Generisk koppling: Lägga till en kopplingsmodul		+	+	FB3sl (M3)
CPDEL	K	Generisk koppling: Radering av en kopplingsmodul		+	+	FB3sl (M3)
CPFMOF	K	Generisk koppling: Beteende hos följdaxeln vid fullständig fränkoppling		+	+	FB3sl (M3)
CPFMON	K	Generisk koppling: Beteende hos följdaxeln vid tillkoppling		+	+	FB3sl (M3)
CPFMSON	K	Generisk koppling: Synkroniseringsmodus		+	+	FB3sl (M3)
CPFPOS	K	Generisk koppling: Synkronposition för följdaxeln		+	+	FB3sl (M3)
CPFRS	K	Generisk koppling: Koordinatreferenssystem		+	+	FB3sl (M3)
CPLA	K	Generisk koppling: Definition av en styraxel		+	-	FB3sl (M3)
CPLCTID	K	Generisk koppling: Nummer för kurvtabellen		+	+	FB3sl (M3)
CPLDEF	K	Generisk koppling: Definition av en styraxel och tillägg av en kopplingsmodul		+	+	FB3sl (M3)
CPLDEL	K	Generisk koppling: Radering av en styraxel till en kopplingsmodul		+	+	FB3sl (M3)
CPLDEN	K	Generisk koppling: Nämnare för kopplingsfaktorn		+	+	FB3sl (M3)
CPLINSC	K	Generisk koppling: Skalfaktor för en styraxels ingångsvärde		+	+	FB3sl (M3)
CPLINTR	K	Generisk koppling: Förflyttningsvärde för en styraxels ingångsvärde		+	+	FB3sl (M3)
CPLNUM	K	Generisk koppling: Täljare för kopplingsfaktorn		+	+	FB3sl (M3)
CPLOF	K	Generisk koppling: Fränkoppling av en styraxel till en kopplingsmodul		+	+	FB3sl (M3)
CPLON	K	Generisk koppling: Tillkoppling av en styraxel till en kopplingsmodul		+	+	FB3sl (M3)
CPLOUTSC	K	Generisk koppling: Skalfaktor för en kopplings utgångsvärde		+	+	FB3sl (M3)
CPLOUTTR	K	Generisk koppling: Förflyttningsvärde för en kopplings utgångsvärde		+	+	FB3sl (M3)
CPLPOS	K	Generisk koppling: Synkronposition för styraxeln		+	+	FB3sl (M3)
CPLSETVAL	K	Generisk koppling: Kopplingsreferens		+	+	FB3sl (M3)

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
CPMALARM	K	Generisk koppling: Undertryckning av speciella kopplingsrelaterade larmmeddelanden		+	+	FB3sl (M3)
CPMBRAKE	K	Generisk koppling: Beteende hos följdaxeln vid vissa stoppsignaler och -kommandon		+	-	FB3sl (M3)
CPMPRT	K	Generisk koppling: Kopplingsbeteende vid detaljprogramstart under blocksökning via programtest		+	+	FB3sl (M3)
CPMRESET	K	Generisk koppling: Kopplingsbeteende vid RESET		+	+	FB3sl (M3)
CPMSTART	K	Generisk koppling: Kopplingsbeteende vid detaljprogramstart		+	+	FB3sl (M3)
CPMVDI	K	Generisk koppling: Beteende hos följdaxeln vid vissa NC/PLC-gränssnittssignaler		+	+	FB3sl (M3)
CPOF	K	Generisk koppling: Frånkoppling av en kopplingsmodul		+	+	FB3sl (M3)
CPON	K	Generisk koppling: Tillkoppling av en kopplingsmodul		+	+	FB3sl (M3)
CPRECOF ⁶⁾	G	Koppla från programmerbar konturprecision	m	+		PGAsl
CPRECON	G	Koppla till programmerbar konturprecision	m	+		PGAsl
CPRES	K	Generisk koppling: Aktiverar de projekterade data för synkron-spindelkopplingen		+	-	FB3sl (M3)
CPROT	P	Koppla till/från kanalspecifikt skyddsområde		+	-	PGAsl
CPROTDEF	P	Definition av ett kanalspecifikt skyddsområde		+	-	PGAsl
CPSETTYPE	K	Generisk koppling: Kopplingstyp		+	+	FB3sl (M3)
CPSYNCOF	K	Generisk koppling: Tröskelvärde för positions-synkrongång "Grov"		+	+	FB3sl (M3)
CPSYNCOF2	K	Generisk koppling: Tröskelvärde för positions-synkrongång "Grov" 2		+	+	FB3sl (M3)
CPSYNCOV	K	Generisk koppling: Tröskelvärde för hastighets-synkrongång "Grov"		+	+	FB3sl (M3)
CPSYNFIP	K	Generisk koppling: Tröskelvärde för positions-synkrongång "Fin"		+	+	FB3sl (M3)
CPSYNFIP2	K	Generisk koppling: Tröskelvärde för positions-synkrongång "Fin" 2		+	+	FB3sl (M3)
CPSYNFIV	K	Generisk koppling: Tröskelvärde för hastighets-synkrongång "Fin"		+	+	FB3sl (M3)
CR	A	Cirkelradie	s	+		PGsl
CROT	F	Vridning av det aktuella koordinatsystemet		+	-	PGAsl
CROTS	F	Programmerbara framevridningar med rymdvinklar (vridning i de angivna axlarna)	s	+	-	PGsl
CRPL	F	Frame-vridning i ett valfritt plan		+	-	FB1sl (K2)
CSCALE	F	Skalfaktor för flera axlar		+	-	PGAsl
CSPLINE	F	Kubisk spline	m	+		PGAsl
CT	G	Cirkel med tangential övergång	m	+		PGsl
CTAB	F	Fastställ följdaxelposition med hjälp av styraxelpositionen ur kurvtabellen		+	+	PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
CTABDEF	P	Koppla till tabelldefinition		+	-	PGAsl
CTABDEL	P	Radera kurvtabell		+	-	PGAsl
CTABEND	P	Koppla från tabelldefinition		+	-	PGAsl
CTABEXISTS	F	Kontrollerar kurvtabellen med nummer n		+	+	PGAsl
CTABFNO	F	Antal ännu möjliga kurvtabeller i minnet		+	+	PGAsl
CTABFPOL	F	Antal ännu möjliga polynom i minnet		+	+	PGAsl
CTABFSEG	F	Antal ännu möjliga kurvsegment i minnet		+	+	PGAsl
CTABID	F	Levererar tabellnummer för den n:e kurvtabel- len		+	+	PGAsl
CTABINV	F	Fastställ styraxelposition med hjälp av följda- xelpositionen ur kurvtabellen		+	+	PGAsl
CTABISLOCK	F	Ger tillbaka spärrtillståndet för kurvtabellen med nummer n		+	+	PGAsl
CTABLOCK	P	Radera och skriva över, spärra		+	+	PGAsl
CTABMEMTYP	F	Ger tillbaka minnet i vilket kurvtabellen med nummer n ligger.		+	+	PGAsl
CTABMPOL	F	Antal maximalt möjliga polynom i minnet		+	+	PGAsl
CTABMSEG	F	Antal maximalt möjliga kurvsegment i minnet		+	+	PGAsl
CTABNO	F	Antal definierade kurvtabeller i SRAM eller DRAM		+	+	FB3sl (M3)
CTABNOMEM	F	Antal definierade kurvtabeller i SRAM eller DRAM		+	+	PGAsl
CTABPERIOD	F	Ger tillbaka tabellperiodiciteten för kurvtabel- len med nummer n		+	+	PGAsl
CTABPOL	F	Antal redan använda polynom i minnet		+	+	PGAsl
CTABPOLID	F	Antal använda kurvpolytom från kurvtabellen med nummer n		+	+	PGAsl
CTABSEG	F	Antal redan använda kurvsegment i minnet		+	+	PGAsl
CTABSEGID	F	Antal använda kurvsegment från kurvtabellen med nummer n		+	+	PGAsl
CTABSEV	F	Levererar slutvärdet för följdaxeln till ett seg- ment i kurvtabellen		+	+	PGAsl
CTABSSV	F	Levererar startvärdet för följdaxeln till ett seg- ment i kurvtabellen		+	+	PGAsl
CTABTEP	F	Levererar värdet för styraxeln vid kurvtabe- llens slut		+	+	PGAsl
CTABTEV	F	Levererar värdet för följdaxeln vid kurvtabe- llens slut		+	+	PGAsl
CTABTMAX	F	Levererar maximalvärdet för följdaxeln i kurv- tabellen		+	+	PGAsl
CTABTMIN	F	Levererar minimalvärdet för följdaxeln i kurv- tabellen		+	+	PGAsl
CTABTSP	F	Levererar värdet för styraxeln vid kurvtabe- llens början		+	+	PGAsl
CTABTSV	F	Levererar värdet för följdaxeln vid kurvtabe- llens början		+	+	PGAsl

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
CTABUNLOCK	P	Upphåva rader- och överskrivningsspärren		+	+	PGAsl
CTOL	A	Konturtolerans för kompressor-funktioner, orienteringsjämnning och översläpningstyper	m	+		PGAsl
CTTRANS	F	Nollpunktsförflyttning för flera axlar		+	-	PGAsl
CUT2D ⁶⁾	G	2D-WRK	m	+		PGsl
CUT2DD	G	Till ett differensverktyg relaterad 2½-D-WRK	m	+		PGsl
CUT2DF	G	2D-WRK, relativt till den aktuella framen (lutande plan)	m	+		PGsl
CUT2DFD	G	Till ett differensverktyg relaterad 2½ D-WRK, relativt till den aktuella framen (lutande plan)	m	+		PGsl
CUT3DC	G	3D-WRK för periferifräsningen	m	+		PGAsl
CUT3DCC	G	3D-WRK för periferifräsningen under hänsynstagande till en begränsningsyta med 3D-radiekompensering: Kontur på bearbetningsytan	m	+		PGAsl
CUT3DCCD	G	3D-WRK för periferifräsningen under hänsynstagande till en begränsningsyta med differensverktyg på verktygets medelpunktsbana: Ansättning mot begränsningsyta	m	+		PGAsl
CUT3DCD	G	Till ett differensverktyg relaterad 3D-WRK för periferifräsningen	m	+		PGAsl
CUT3DF	G	3D-WRK för planfräsning med orienteringsändring	m	+		PGAsl
CUT3DFD	G	Till ett differensverktyg relaterad 3D-WRK för planfräsning med orienteringsändring	m	+		PGAsl
CUT3DFF	G	3D-WRK för planfräsning med konstant orientering. Verktygsorienteringen är fastlagd genom G17 - G19 och eventuellt av en frameviden riktning.	m	+		PGAsl
CUT3DFS	G	3D-WRK för planfräsning med konstant orientering. Verktygsorienteringen är fastlagd genom G17 - G19 och påverkas inte av frames.	m	+		PGAsl
CUTCONOF ⁶⁾	G	Koppla från konstant radiekompensering	m	+		PGsl
CUTCONON	G	Koppla till konstant radiekompensering	m	+		PGsl
CUTMOD	A	Koppla till modifikation av kompenseringsdata för vridbara verktyg (i förbindelse med orienterbar verktygsbärare)	m	+		PGAsl
CUTMODK	A	Koppla till modifikation av kompenseringsdata för vridbara verktyg (i förbindelse med orienteringstransformationer, som definierades med kinematiska följder)	m	+		PGAsl
CYCLE60	C (T)	Gravyrcykel		+		PGAsl
CYCLE61	C (T)	Planfräsning		+		PGAsl
CYCLE62	C (T)	Konturanrop		+		PGAsl
CYCLE63	C (T)	Fräsa konturficka		+		PGAsl
CYCLE64	C (T)	Förborra konturficka		+		PGAsl
CYCLE70	C (T)	Gängfräsning		+		PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
CYCLE72	C (T)	Banfräsning		+		PGAsI
CYCLE76	C (T)	Fräsa fyrkanttapp		+		PGAsI
CYCLE77	C (T)	Fräsa cirkeltapp		+		PGAsI
CYCLE78	C (T)	Fräsa hålgänga		+		PGAsI
CYCLE79	C (T)	Flerkant		+		PGAsI
CYCLE81	C (T)	Borra, centrera		+		PGAsI
CYCLE82	C (T)	Borra, plansänka		+		PGAsI
CYCLE83	C (T)	Djuphålsborra		+		PGAsI
CYCLE84	C (T)	Gängtappning utan flytande gänghållare		+		PGAsI
CYCLE85	C (T)	Brotschning		+		PGAsI
CYCLE86	C (T)	Ursvarvning		+		PGAsI
CYCLE92	C (T)	Avstickning		+		PGAsI
CYCLE95	C (T)	Konturavspåning		+		PGAsI
CYCLE98	C (T)	Kopplade gängor		+		PGAsI
CYCLE99	C (T)	Gängsvarvning		+		PGAsI
CYCLE150	C (M)	Visa / protokollera mätresultat		+		BNMsl
CYCLE435	C (T)	Beräkna skärpverktygsposition		+		PGAsI
CYCLE495	C (T)	Profilera		+		PGAsI
CYCLE750	C (A)	Intern arvetscykel för CYCLE751 ... CYCLE759 (innehåller MMC-kommandot den det egentliga funktionsanropet)		-		FB3sl (T4)
CYCLE751	C (A)	Öppna / utföra / stänga optimeringssession		M		FB3sl (T4)
CYCLE752	C (A)	Lägga till axel till en optimeringssession		M		FB3sl(T4)
CYCLE753	C (A)	Välja optimeringsmode		M		FB3sl (T4)
CYCLE754	C (A)	Lägga till / ta bort datablock		M		FB3sl (T4)
CYCLE755	C (A)	Spara / återskapa datablock		M		FB3sl (T4)
CYCLE756	C (A)	Aktivera optimeringsresultat		M		FB3sl (T4)
CYCLE757	C (A)	Spara optimeringsdata		M		FB3sl (T4)
CYCLE758	C (A)	Ändra parametervärde		M		FB3sl (T4)
CYCLE759	C (A)	Läsa parametervärde		M		FB3sl (T4)
CYCLE800	C (T)	Vridning		+		PGAsI
CYCLE801	C (T)	Gitter eller ram		+		PGAsI
CYCLE802	C (T)	Valfria positioner		+		PGAsI
CYCLE830	C (T)	Djuphålsborra 2		+		PGAsI
CYCLE832	C (T)	High Speed Settings		+		PGAsI
CYCLE840	C (T)	Gängtappning med flytande gänghållare		+		PGAsI
CYCLE899	C (T)	Fräsa öppet spår		+		PGAsI
CYCLE930	C (T)	Instick		+		PGAsI
CYCLE940	C (T)	Fristick former		+		PGAsI
CYCLE951	C (T)	Avspåning		+		PGAsI
CYCLE952	C (T)	Kontursticka		+		PGAsI

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
CYCLE961	C (M)	Bestämma läget för ett arbetsstyckshörn (inre eller yttre) och använda som nollpunktsförflyttning		+		BNMsl
CYCLE971	C (M)	Kalibrera verktygsmätprobe, mäta verktygslängd och/eller verktygsradie (endast för teknologin fräsa)		+		BNMsl
CYCLE973	C (M)	Kalibrera arbetsstycksmätprobe kalibreras vid en yta på arbetsstycket eller i ett spår (endast för teknologin svarva).		+		BNMsl
CYCLE974	C (M)	Bestämma arbetsstycksnollpunkt i den valda mätaxeln, bestämma verktygskompensering med 1-punkts mätning (endast för teknologin svarva).		+		BNMsl
CYCLE976	C (M)	Kalibrera arbetsstycksmätprobe i en kalibrering resp. vid en kalibreringskula fullständigt i arbetsplanet eller vid en kant för en bestämd axel och riktning.		+		BNMsl
CYCLE977	C (M)	Bestämma medelpunkt i planet samt bredden resp. diametern		+		BNMsl
CYCLE978	C (M)	Mäta position för en kant i arbetsstyckskoordinatsystemet		+		BNMsl
CYCLE979	C (M)	Bestämma medelpunkt i planet, mäta radie för cirkelsegmenten		+		BNMsl
CYCLE982	C (M)	Kalibrera verktygsmätprobe, mäta svarv-, borr- och fräsverktyg (endast för teknologin svarva)		+		BNMsl
CYCLE994	C (M)	Bestämma arbetsstycksnollpunkt i den valda mätaxeln med 2-punkts mätning (endast för teknologin svarva)		+		BNMsl
CYCLE995	C (M)	Mäta vinklingen för spindeln på en verktygsmaskin		+		BNMsl
CYCLE996	C (M)	Bestämma transformationsrelevanta data för kinematiska transformationer med roterande axlar		+		BNMsl
CYCLE997	C (M)	Bestämma medelpunkt och diameter för en kula, mäta medelpunkter för tre fördelade kullor		+		BNMsl
CYCLE998	C (M)	Bestämma vinkelläget för en yta (plan) relaterat till arbetsplanet, bestämma vinkeln från kanter i arbetsstyckskoordinatsystemet		+		BNMsl
CYCLE4071	C (T)	Längsslipning med ansättning i vändpunkten		+		PGAsl
CYCLE4072	C (T)	Längsslipning med ansättning i vändpunkten och avbrottssignal		+		PGAsl
CYCLE4073	C (T)	Längsslipning med kontinuerlig ansättning		+		PGAsl
CYCLE4074	C (T)	Längsslipning med kontinuerlig ansättning och avbrottssignal		+		PGAsl
CYCLE4075	C (T)	Planslipning med ansättning i vändpunkten		+		PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
CYCLE4077	C (T)	Planslipning med ansättning i vändpunkten och avbrottssignal		+		PGAsl
CYCLE4078	C (T)	Planslipning med kontinuerlig ansättning		+		PGAsl
CYCLE4079	C (T)	Planslipning med intermittent ansättning		+		PGAsl

Anvisningar D ... F

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
D	A	Verktygskompenserings-nummer		+		PGsl
D0	A	Vid D0 är kompenseringsarna för verktyget ej verksamma		+		PGsl
DAC	K	Absolut blockvis axelspecifik diometer-programmering	s	+		PGsl
DC	K	Absolut måttuppgift för roterande axlar, uppsöka position direkt	s	+		PGsl
DCI	K	Tillordna dataklass I (= individuell) (endast SINUMERIK 828D!)		+		PGAsl
DCM	K	Tillordna dataklass M (= tillverkare) (endast SINUMERIK 828D!)		+		PGAsl
DCU	K	Tillordna dataklass U (= användare) (endast SINUMERIK 828D!)		+		PGAsl
DEF	K	Variabeldefinition		+		PGAsl
DEFAULT	K	Gren i CASE-förgreningen		+		PGAsl
DEFINE	K	Nyckelord för makrodefinitioner		+		PGAsl
DELAYFSTOF	P	Definiera slutet på ett Stopp-Delay-område	m	+	-	PGAsl
DELAYFSTON	P	Definiera början på ett Stopp-Delay-område	m	+	-	PGAsl
DELDL	F	Radera additiva kompenserings		+	-	PGAsl
DELDTG	P	Restvägsradering		-	+	FBSYsl
DELETE	P	Radera den angivna filen. Filnamnet kan anges med sökväg och fil-identifikation.		+	-	PGAsl
DELMLOWNER	F	Radera verktygets ägarmagasinplats		+	-	FBWsl
DELMRES	F	Radera magasinplats-reservering		+	-	FBWsl
DELMT	P	Radera Multitool		+	-	FBWsl
DELOBJ	F	Radering av element till kinematiska följder, skyddsområden, skyddsområdeselement, kollisionsspar och transformationsdata		+		PGAsl
DELT	P	Radera verktyg		+	-	FBWsl
DELTC	P	Radera datablock för verktygsbärare		+	-	FBWsl
DELTOOLENV	F	Radera datablock för beskrivning av verktygsomgivning		+	-	PGAsl
DIACYCOFA	K	Axelspecifik modal diameter-programmering: FRÅN i cykler	m	+		FB1sl (P1)

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
¹⁾²⁾³⁾⁴⁾⁵⁾ Förklaringar se teckenförklaring (Sida 852).						
DIAM90	G	Diameterprogrammering för G90, radieprogrammering för G91	m	+		PGAsl
DIAM90A	K	Axelspecifik modal diameterprogrammering för G90 och AC, radieprogrammering för G91 och IC	m	+		PGsl
DIAMCHAN	K	Övertagande av alla axlar från MD axelfunktionerna i kanaltilståndet för diameterprogrammeringen		+		PGsl
DIAMCHANA	K	Övertagande kanal-tilstånd för diameterprogrammering		+		PGsl
DIAMCYCOF	G	Kanalspecifik diameter-programmering: FRÅN i cykler	m	+		FB1sl (P1)
DIAMOF ⁶⁾	G	Diameter-programmering FRÅN Grundläge se maskintillverkare	m	+		PGsl
DIAMOFA	K	Axelspecifik modal diameter-programmering: FRÅN Grundläge se maskintillverkare	m	+		PGsl
DIAMON	G	Diameter-programmering TILL	m	+		PGsl
DIAMONA	K	Axelspecifik modal diameter-programmering: TILL Frikoppling se maskintillverkare	m	+		PGsl
DIC	K	Relativ blockvis axelspecifik diameter-programmering	s	+		PGsl
DILF	A	Återgångsväg (längd)	m	+		PGsl
DISABLE	P	Interrupt FRÅN		+	-	PGAsl
DISC	A	Överhöjning övergångscirkel verktygsradiekompenseringen	m	+		PGsl
DISCL	A	Avståndet för den snabba ansättningsrörelsens slutpunkt från bearbetningsplanet		+		PGsl
DISPLOF	PA	Undertrycka aktuell blockindikering		+		PGAsl
DISPLON	PA	Upphäva undertryckning av den aktuella blockindikeringen		+		PGAsl
DISPR	A	Repos-bandifferens	s	+		PGAsl
DISR	A	Repos-avstånd	s	+		PGAsl
DISRP	A	Avstånd för återgångsplanet från bearbetningsplanet vid mjuk fram- och bortkörning		+		PGsl
DITE	A	Gångutgångsväg	m	+		PGsl
DITS	A	Gångingångsväg	m	+		PGsl
DIV	K	Integer-division		+		PGAsl
DL	A	Välja ortsberoende additiv verktygskompensering (DL, summa-rigningskompensering)	m	+		PGAsl
DO	A	Nyckelord för synkronaktion, utlöser action vid uppfyllt villkor		-	+	FBSYsl
DRFOF	P	Frånkoppling av handrattsförflyttningar (DRF)	m	+	-	PGsl
DRIVE	G	Hastighetsberoende banacceleration	m	+		PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
¹⁾²⁾³⁾⁴⁾⁵⁾ Förklaringar se teckenförklaring (Sida 852).						
DRIVEA	P	Koppla till bruten accelerationskurva för de programmerade axlarna		+	-	PGAsI
DYNFINISH	G	Dynamik för mycket fin bearbetning	m	+		PGAsI
DYNNORM ⁶⁾	G	Normal dynamik	m	+		PGAsI
DYNPOS	G	Dynamik för positioneringsdrift, gängtappning	m	+		PGAsI
DYNROUGH	G	Dynamik för grovbearbetning	m	+		PGAsI
DYNSEMIFIN	G	Dynamik för finbearbetning	m	+		PGAsI
DZERO	P	Kännetecknar alla D-nummer i TO-enheten som ogiltiga		+	-	PGAsI
EAUTO	G	Fastläggande av det sista spline-avsnittet genom de sista 3 punkterna	m	+		PGAsI
EGDEF	P	Definition av en elektronisk växel		+	-	PGAsI
EGDEL	P	Radera kopplings-definition för följd-axeln		+	-	PGAsI
EGOFC	P	Koppla från elektronisk växel kontinuerligt		+	-	PGAsI
EGOFS	P	Koppla från elektronisk växel selektivt		+	-	PGAsI
EGON	P	Koppla till elektronisk växel		+	-	PGAsI
EGONSYN	P	Koppla till elektronisk växel		+	-	PGAsI
EGONSYNE	P	Koppla till elektroniska växel, med föreskrift om uppsökningsmode		+	-	PGAsI
ELSE	K	Programförgrening, om IF-villkor inte uppfyllt		+		PGAsI
ENABLE	P	Interrupt TILL		+	-	PGAsI
ENAT ⁶⁾	G	Naturlig kurvövergång till nästa förflytningsblock	m	+		PGAsI
ENDFOR	K	Slutrad för FOR-räkneslinga		+		PGAsI
ENDIF	K	Slutrad för IF-förgrening		+		PGAsI
ENDLABEL	K	Slutmärke för upprepning av detaljprogram via REPEAT		+		PGAsI, FB1sI (K1)
ENDLOOP	K	Slutrad för den ändlösa programslingan LOOP		+		PGAsI
ENDPROC	K	Slutrad i ett program med begynnelserad PROC		+		
ENDWHILE	K	Slutrad i WHILE-slingan		+		PGAsI
ESRR	P	Parametrera drivningsautarkt ESR-återdragande i drivningen		+		PGAsI
ESRS	P	Parametrera drivningsautarkt ESR-stoppan- de i drivningen		+		PGAsI
ETAN	G	Tangentiell kurvövergång till nästa förflytt- ningsblock vid spline-början	m	+		PGAsI
EVERY	K	Utföra synkronaktion vid övergång av villkoret från FALSE till TRUE		-	+	FBSYsI
EX	K	Nyckelvärde för värdetilldelning i exponentiellt skrivsätt		+		PGAsI
EXECSTRING	P	Överföring av en string-variabel med den de- taljprogramrad som ska utföras		+	-	PGAsI
EXECTAB	P	Genomarbete ett element från en rörelsetabell		+	-	PGAsI
EXECUTE	P	Programutförande TILL		+	-	PGAsI

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1)2)3)4)5) Förklaringar se teckenförklaring (Sida 852).						
EXP	F	Exponentialfunktion ex		+	+	PGAsl
EXTCALL	A	Genomarbete externt underprogram		+	+	PGAsl
EXTCLOSE	P	Stäng öppnad extern apparat/fil för att skriva		+	-	PGAsl
EXTERN	K	Tillkännagivande av ett underprogram med parameteröverföring		+		PGAsl
EXTOPEN	P	Öppna extern apparat/fil för kanalen för att skriva		+	-	PGAsl
F	A	Matningsvärde (i förbindelse med G4 programmeras med F också fördröjningstiden)		+	+	PGsl
FA	K	Axial matning	m	+	+	PGsl
FAD	A	Ansättnings-matning för mjuk fram- och bortkörning		+		PGsl
FALSE	K	Logisk konstant: felaktig		+	+	PGAsl
FB	A	Blockvis matning		+		PGsl
FCTDEF	P	Definiera polynomfunktion		+	-	PGAsl
FCUB	G	Matning enligt kubisk spline föränderlig	m	+		PGAsl
FD	A	Banmatning för handrattsöverlagring	s	+		PGsl
FDA	K	Axial matning för handrattsöverlagring	s	+		PGsl
FENDNORM ⁶⁾	G	Hörnfördröjning FRÅN	m	+		PGAsl
FFWOF ⁶⁾	G	Förstyrning FRÅN	m	+		PGAsl
FFWON	G	Förstyrning Till	m	+		PGAsl
FGREF	K	Referensradie vid roterande axlar eller banreferensfaktorer vid orienteringsaxlar (vektorinterpolering)	m	+		PGsl
FGROUP	P	Fastläggande av ax(el)(lar) med banmatning		+	-	PGsl
FI	K	Parameter för åtkomst till framedata: Finförflyttning		+		PGAsl
FIFOCTRL	G	Styrning av förkörningsbuffert	m	+		PGAsl
FILEDATE	P	Levererar datum för den sista skrivande åtkomsten till filen		+	-	PGAsl
FILEINFO	P	Levererar summan av FILEDATE, FILESIZE, FILESTAT och FILETIME tillsammans		+	-	PGAsl
FILESIZE	P	Levererar aktuell storlek på filen		+	-	PGAsl
FILESTAT	P	Levererar filstatus för rättigheterna läsa, skriva, execute, visa, radera (rwxsd)		+	-	PGAsl
FILETIME	P	Levererar tiden för den sista skrivande åtkomsten till filen		+	-	PGAsl
FINEA	K	Rörelseslut när "Fint precisionsstopp" uppnås	m	+		PGAsl
FL	K	Gränshastighet för synkronaxlar	m	+		PGsl
FLIN	G	Matning linjärt föränderlig	m	+		PGAsl
FMA	K	Flera matningar axiala	m	+		PGsl
FNORM ⁶⁾	G	Matning normal enligt DIN66025	m	+		PGAsl
FOC	K	Blockvis verksam moment/kraft-begränsning	s	-	+	FBSYsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
¹⁾²⁾³⁾⁴⁾⁵⁾ Förklaringar se teckenförklaring (Sida 852).						
FOCOF	K	Koppla från modal moment/kraft-begränsning	m	-	+	FBSYsl
FOCON	K	Koppla till modal moment/kraft-begränsning	m	-	+	FBSYsl
FOR	K	Räkneslinga med fast antal körningar		+		PGAsl
FP	A	Fast punkt: Nummer för den fast punkten som ska uppsökas	s	+		PGsl
FPO	K	Via ett polynom programmerat matningsförlopp		+		PGAsl
FPR	P	Beteckning roterande axel		+	-	PGsl
FPRAOF	P	Koppla från varvmatning		+	-	PGsl
FPRAON	P	Koppla till varvmatning		+	-	PGsl
FRAME	K	Datotyp för fastläggande av koordinatsystem		+		PGAsl
FRC	A	Matning för radie och fas	s	+		PGsl
FRCM	A	Matning för radie och fas modal	m	+		PGsl
FROM	K	Aktionen utförs när villkoret har uppfyllts en gång och så länge som synkronaktionen är aktiv		-	+	FBSYsl
FTOC	P	Ändra verktygsfinkompensering		-	+	FBSYsl
FTOCOF ⁶⁾	G	Online verksam verktygsfinkompensering FRÅN	m	+		PGAsl
FTOCON	G	Online verksam verktygsfinkompensering TILL	m	+		PGAsl
FXS	K	Köra till fast anslag till	m	+	+	PGsl
FXST	K	Momentgräns för körning till fast anslag	m	+	+	PGsl
FXSW	K	Övervakningsfönster för körning till fast anslag		+	+	PGsl
FZ	K	Tandmatning	m	+		PGsl

Anvisningar G ... L

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
¹⁾²⁾³⁾⁴⁾⁵⁾ Förklaringar se teckenförklaring (Sida 852).						
G0	G	Linjär interpolering med snabbgång (snabbgångsrörelse)	m	+		PGsl
G1 ⁶⁾	G	Linjärinterpolering med matning (rät linje interpolering)	m	+		PGsl
G2	G	Cirkelinterpolering medurs	m	+		PGsl
G3	G	Cirkelinterpolering moturs	m	+		PGsl
G4	G	Fördröjningstid, tiden förbestämd	s	+		PGsl
G5	G	Slipning med sned instickning	s	+		PGAsl
G7	G	Utgjavningsrörelse vid slipning med sned instickning	s	+		PGAsl
G9	G	Precisionsstopp - hastighetsreducering	s	+		PGsl
G17 ⁶⁾	G	Val av arbetsplan X/Y	m	+		PGsl
G18	G	Val av arbetsplan Z/X	m	+		PGsl

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
G19	G	Val av arbetsplan Y/Z	m	+		PGsl
G25	G	Undre arbetsfältsbegränsning	s	+		PGsl
G26	G	Övre arbetsfältsbegränsning	s	+		PGsl
G33	G	Gängskärning med konstant stigning	m	+		PGsl
G34	G	Gängskärning med linjärt tilltagande stigning	m	+		PGsl
G35	G	Gängskärning med linjärt avtagande stigning	m	+		PGsl
G40 ⁶⁾	G	Verktysradiekompensering FRÅN	m	+		PGsl
G41	G	Verktysradiekompensering till vänster om konturen	m	+		PGsl
G42	G	Verktysradiekompensering till höger om konturen	m	+		PGsl
G53	G	Undertryckning av den aktuella nollpunktsförflyttningen (blockvis)	s	+		PGsl
G54	G	1:a inställbara nollpunktsförflyttning	m	+		PGsl
G55	G	2:a inställbara nollpunktsförflyttning	m	+		PGsl
G56	G	3:e inställbara nollpunktsförflyttning	m	+		PGsl
G57	G	4:e inställbara nollpunktsförflyttning	m	+		PGsl
G58 (840D sl)	G	Absolut programmerbar nollpunktsförflyttning (grovförflyttning)	s	+		PGsl
G58 (828D)	G	5:e inställbara nollpunktsförflyttning	m	+		PGsl
G59 (840D sl)	G	Additiv programmerbar nollpunktsförflyttning (finförflyttning)	s	+		PGsl
G59 (828D)	G	6:e inställbara nollpunktsförflyttning	m	+		PGsl
G60 ⁶⁾	G	Precisionsstopp - hastighetsreducering	m	+		PGsl
G62	G	Hörnfördröjning vid innerhörn vid aktiv verktysradiekompensering (G41, G42)	m	+		PGAsl
G63	G	Gängtappning med flytande gänghållare	s	+		PGsl
G64	G	Banstyrningsdrift	m	+		PGsl
G70	G	Inch-måttuppgift för geometriska uppgifter (längder)	m	+	+	PGsl
G71 ⁶⁾	G	Metrisk måttuppgift för geometriska uppgifter (längder)	m	+	+	PGsl
G74	G	Referenspunktskörning	s	+		PGsl
G75	G	Fastpunktskörning	s	+		PGsl
G90 ⁶⁾	G	Måttuppgift absolut	m/s	+		PGsl
G91	G	Kedjemåttuppgift	m/s	+		PGsl
G93	G	Tidsreciprok matning 1/min	m	+		PGsl
G94 ⁶⁾	G	Linjär matning F i mm/min eller inch/min och grader/min)	m	+		PGsl
G95	G	Varvmatning F i mm/varv eller inch/varv	m	+		PGsl
G96	G	Varvmatning (som vid G95) och konstant skärhastighet	m	+		PGsl
G97	G	Varvmatning och konstant spindelvarvtal (konstant skärhastighet FRÅN)	m	+		PGsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
G110	G	Polprogrammering, relativ till den sist programmerade börpositionen	s	+		PGsl
G111	G	Polprogrammering relativ till nollpunkten i det aktuella arbetsstyckskoordinatsystemet	s	+		PGsl
G112	G	Polprogrammering relativ till sista giltiga pol	s	+		PGsl
G140 ⁶⁾	G	Uppsökningsriktning WAB fastlagd genom G41/G42	m	+		PGsl
G141	G	Uppsökningsriktning WAB till vänster om konturen	m	+		PGsl
G142	G	Uppsökningsriktning WAB till höger om konturen	m	+		PGsl
G143	G	Uppsökningsriktning WAB tangentberoende	m	+		PGsl
G147	G	Mjuk framkörning med rät linje	s	+		PGsl
G148	G	Mjuk bortkörning med rät linje	s	+		PGsl
G153	G	Undertryckning av aktuella frames inklusive basframe	s	+		PGsl
G247	G	Mjuk framkörning med fjärdedels cirkel	s	+		PGsl
G248	G	Mjuk bortkörning med fjärdedels cirkel	s	+		PGsl
G290 ⁶⁾	G	Omkoppling till SINUMERIK-mode TILL	m	+		FBWsl
G291	G	Omkoppling till ISO2/3-mode TILL	m	+		FBWsl
G331	G	Gängtappning utan flytande gänghållare, positiv stigning, högergång	m	+		PGsl
G332	G	Gängtappning utan flytande gänghållare, negativ stigning, vänstergång	m	+		PGsl
G335	G	Svarva en kullrig gänga medurs	m	+		PGsl
G336	G	Svarva en kullrig gänga moturs	m	+		PGsl
G340 ⁶⁾	G	Rymduppsökningsblock (djup och i planet samtidigt (helix))	m	+		PGsl
G341	G	Först ansättning i den lodräta axeln (Z), sedan uppsökning i planet	m	+		PGsl
G347	G	Mjuk framkörning med halvcirkel	s	+		PGsl
G348	G	Mjuk bortkörning med halvcirkel	s	+		PGsl
G450 ⁶⁾	G	Övergångscirkel	m	+		PGsl
G451	G	Skärningspunkt för ekvidistanterna	m	+		PGsl
G460 ⁶⁾	G	Tillkoppling av kollisionsovervakningen för fram- och bortkörningsblock	m	+		PGsl
G461	G	Infoga en cirkel i WRK-block	m	+		PGsl
G462	G	Infoga en rät linje i WRK-block	m	+		PGsl
G500 ⁶⁾	G	Koppla från alla inställbara frames, basframes är aktiva	m	+		PGsl
G505 ... G599	G	5:e ... 99:e Inställbar nollpunktsförflyttning	m	+		PGsl
G601 ⁶⁾	G	Blockbyte vid fint precisionsstopp	m	+		PGsl
G602	G	Blockbyte vid grovt precisionsstopp	m	+		PGsl
G603	G	Blockbyte vid IPO-blockslut	m	+		PGsl

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
G621	G	Hörfördröjning vid alla hörn	m	+		PGAsl
G641	G	Banstyrningsdrift med översläpning enligt vägkriterium (= programmerbart översläpningsavstånd)	m	+		PGsl
G642	G	Banstyrningsdrift med övergång under respekterande av definierade toleranser	m	+		PGsl
G643	G	Banstyrningsdrift med övergång under respekterande av definierade toleranser (block-internt)	m	+		PGsl
G644	G	Banstyrningsdrift med övergång med maximalt möjliga dynamik	m	+		PGsl
G645	G	Banstyrningsdrift med övergång av hörn och tangentiella blockövergångar under respekterande av definierade toleranser	m	+		PGsl
G700	G	Inch-måttuppgift för geometriska och teknologiska uppgifter (längder, matning)	m	+	+	PGsl
G710 ⁶⁾	G	Metrisk måttuppgift för geometriska och teknologiska uppgifter (längder, matning)	m	+	+	PGsl
G810 ⁶⁾ , ..., G819	G	För OEM-användaren reserverad G-grupp		+		PGAsl
G820 ⁶⁾ , ..., G829	G	För OEM-användaren reserverad G-grupp		+		PGAsl
G931	G	Koppla från matningsuppgift genom förflyttningstid, konstant banhastighet	m	+		
G942	G	Linjärmatning och konstant skärhastighet eller frysning av spindelvarvtal	m	+		
G952	G	Varvmatning och konstant skärhastighet eller frysning av spindelvarvtal	m	+		
G961	G	Linjärmatning (som vid G94) och konstant skärhastighet	m	+		PGsl
G962	G	Linjärmatning eller varvmatning och konstant skärhastighet	m	+		PGsl
G971	G	Linjärmatning och konstant spindelvarvtal (konstant skärhastighet FRÅN)	m	+		PGsl
G972	G	Linjärmatning eller varvmatning och konstant spindelvarvtal (konstant skärhastighet FRÅN)	m	+		PGsl
G973	G	Varvmatning eller spindelvarvtalsbegränsning och konstant spindelvarvtal (G97 utan LIMS för ISO-mode)	m	+		PGsl
GEOAX	P	Tillordna geometriaxlarna 1 - 3 nya kanalaxlar		+	-	PGAsl
GET	P	Byta frigivna axlar mellan kanalerna		+	+	PGAsl
GETACTT	F	Bestämmer det aktiva verktyget från en grupp verktyg med samma namn		+	-	FBWsl
GETACTTD	F	Bestämmer till ett absolut D-nummer det tillhörande T-numret		+	-	PGAsl
GETD	P	Byta axel direkt mellan kanaler		+	-	PGAsl
GETDNO	F	Levererar D-nummer för ett skär (CE) till ett verktyg (T)		+	-	PGAsl
GETEXET	P	Läsa det inväxlade T-numret		+	-	FBWsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
GETFREELOC	P	För ett givet verktyg söka en tom plats i magasinen		+	-	FBWsl
GETSELT	P	Leverera förvalt T-nummer		+	-	FBWsl
GETT	F	Bestämma T-nummer till verktygsnamn		+	-	FBWsl
GETTCOR	F	Läsa ut verktygslängder resp. verktygslängdkomponenter		+	-	PGAsl
GETTENV	F	Läsa T-, D-, och DL-nummer		+	-	PGAsl
GETVARAP	F	Läsa åtkomstskydd på en system-/användarvariabel		+	-	PGAsl
GETVARDFT	F	Läsa åtkomstskydd för en system-/användarvariabel		+	-	PGAsl
GETVARLIM	F	Läsa gränsvärde för en system-/användarvariabel		+	-	PGAsl
GETVARPHU	F	Läsa fysikalisk enhet för en system-/användarvariabel		+	-	PGAsl
GETVARTYP	F	Läsa datatyp för en system-/användarvariabel		+	-	PGAsl
GFRAME0 ... GFRAME100	G	Aktivering av slipramen <n> för datahållningen i kanalen	m	+		PGsl
GOTO	K	Hoppa anvisning först framåt sedan bakåt (riktning först mot programslut och sedan mot programstart)		+		PGAsl
GOTOB	K	Hoppa anvisning bakåt (riktning programstart)		+		PGAsl
GOTOC	K	Som GOTO, men undertrycka larm 14080 "Hoppmål inte hittat"		+		PGAsl
GOTOF	K	Hoppa anvisning framåt (riktning programslut)		+		PGAsl
GOTOS	K	Återhopp till programstart		+		PGAsl
GP	K	Nyckelord för indirekt programmering av positionsattribut		+		PGAsl
GROUP_ADDEND	C (T)	Slut inkörningstillägg		+		PGAsl
GROUP_BEGIN	C (T)	Början programblock		+		PGAsl
GROUP_END	C (T)	Slut programblock		+		PGAsl
GWPSOF	P	Välja bort konstant skivperiferihastighet (SUG)	s	+	-	PGsl
GWPSON	P	Välja konstant skivperiferihastighet (SUG)	s	+	-	PGsl
H...	A	Hjälpfunktionsutmatning vid PLC		+	+	PGsl/FB1sl (H2)
HOLES1	C (T)	Hålråd		+		PGAsl
HOLES2	C (T)	Hålcirkel		+		PGAsl
I	A	Interpoleringsparameter	s	+		PGsl
I1	A	Mellanpunkt-koordinat	s	+		PGsl
IC	K	Inmatning med kedjemått	s	+		PGsl
ICYCOF	P	Genomarbota alla block i en teknologicykel enligt ICYCOF i en IPO-takt		+	+	FBSYsl
ICYCON	P	Genomarbota varje block i en teknologicykel enligt ICYCON i en separat IPO-takt		+	+	FBSYsl
ID	K	Beteckning för modala synkronaktioner	m	-	+	FBSYsl

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
IDS	K	Beteckning för modala statiska synkronaktioner		-	+	FBSYsl
IF	K	Inledning av ett villkorligt hopp i detaljprogrammet / teknologicykeln		+	+	PGAsl
INDEX	F	Bestämma index för ett tecken i ingångssträngen		+	-	PGAsl
INICF	K	Initiering av variablerna vid NEWCONF		+		PGAsl
INIPO	K	Initiering av variablerna vid PowerOn		+		PGAsl
INIRE	K	Initiering av variablerna vid Reset		+		PGAsl
INIT	P	Val av ett bestämt NC-program för genomarbetning i en viss kanal		+	-	PGAsl
INITIAL		Skapa en INI-fil över alla områden		+		PGAsl
INT	K	Datatyp: Heltaligt värde med förtecken		+		PGAsl
INTERSEC	F	Beräkna skärningspunkt mellan två konturelement		+	-	PGAsl
INVCCW	G	Köra evolvent moturs	m	+		PGsl
INVCW	G	Köra evolvent medurs	m	+		PGsl
INVFRAME	F	Ur en frame beräkna den inversa framen		+	-	FB1sl (K2)
IP	K	Variabel interpoleringsparameter		+		PGAsl
IPOBRKA	P	Rörelsekriterium från insatspunkten i bromsrampen	m	+	+	
IPOENDA	K	Rörelseslut när "IPO-stopp" uppnås	m	+		PGAsl
IPTRLOCK	P	Infrysning av början till det sökningsodugliga programavsnittet till nästa maskinfunktionsblock.	m	+	-	PGAsl
IPTRUNLOCK	P	Sätta slutet för det sökningsodugliga programavsnittet på det aktuella blocket vid tidpunkten för stopp.	m	+	-	PGAsl
IR	A	Cirkelmedelpunktskoordinat (X-riktning) vid svarvning av kullrig gänga		+		PGsl
ISAXIS	F	Kontrollera om den som parameter angivna geometriaxeln är 1		+	-	PGAsl
ISD	A	Nermatningsdjup	m	+		PGAsl
ISFILE	F	Kontrollera om en fil finns i NC-användarminnet		+	-	PGAsl
ISNUMBER	F	Kontrollera om ingångsstring kan omvandlas till tal		+	-	PGAsl
ISOCALL	K	Indirekt anrop av ett i ISO-språk programmerat program		+		PGAsl
ISVAR	F	Kontrollera om överföringsparametern innehåller en i NC känd variabel		+	-	PGAsl
J	A	Interpoleringsparameter	s	+		PGsl
J1	A	Mellanpunkt-koordinat	s	+		PGsl
JERKA	P	Aktivera via MD inställt accelerationsbeteende för de programmerade axlarna		+	-	

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
JERKLIM	K	Reduktion eller förhöjning av den maximala axiala jerken	m	+		PGAsI
JERKLIMA	K	Reduktion eller förhöjning av den maximala axiala jerken	m	+	+	PGAsI
JR	A	Cirkelmedelpunktskoordinat (Y-riktning) vid svarvning av kullrig gänga		+		PGsI
K	A	Interpoleringsparameter	s	+		PGsI
K1	A	Mellanpunkt-koordinat	s	+		PGsI
KONT	G	Köra runt kontur vid verktygskompensering	m	+		PGsI
KONTC	G	Köra fram/bort med böjningskontinuerligt polynom	m	+		PGsI
KONTT	G	Köra fram/bort med tangentkontinuerligt polynom	m	+		PGsI
KR	A	Cirkelmedelpunktskoordinat (Z-riktning) vid svarvning av kullrig gänga		+		PGsI
L	A	Underprogram-nummer	s	+	+	PGAsI
LEAD	A	Försprångsvinkel 1. Verktygsorientering 2. Orienteringspolynom	m	+		PGAsI
LEADOF	P	Axial styrvärdeskoppling FRÅN		+	+	PGAsI
LEADON	P	Axial styrvärdeskoppling TILL		+	+	PGAsI
LENTOAX	F	Levererar informationer över tillordningen av verktygslängderna L1, L2 och L3 för det aktiva verktyget till abskissa, ordinata och applikata		+	-	PGAsI
LFOF ⁶⁾	G	Snabbåtergång för gängskärning FRÅN	m	+		PGsI
LFON	G	Snabbåtergång för gängskärning TILL	m	+		PGsI
LFPOS	G	Återgång för den med POLFMASK eller POLFMLIN bekantgjorda axeln till den med POLF programmerade absoluta axelpositionen	m	+		PGsI
LFTXT ⁶⁾	G	Plan för återgångsrörelse vid snabblyftning bestäms ur bantangenten och den aktuella verktygsriktningen	m	+		PGsI
LFWP	G	Plan för återgångsrörelse vid snabblyftning bestäms av det aktuella arbetsplanet (G17/G18/G19)	m	+		PGsI
LIFTFAST	K	Snabblyftning		+		PGsI
LIMS	K	Varvtalsbegränsning vid G96/G961 och G97	m	+		PGsI
LLI	K	Undre gränsvärde för variabel		+		PGAsI
LN	F	Naturlig logaritm		+	+	PGAsI
LOCK	P	Spärra synkronaktion med ID (stoppa teknologycykel)		-	+	FBSYsI
LONGHOLE	C (T)	Långhål		+		PGAsI
LOOP	K	Inledning av en ändlös slinga		+		PGAsI

Anvisningar M ... R

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
M0		Programmerat stopp		+	+	PGsl
M1		Valfritt stopp		+	+	PGsl
M2		Programslut huvudprogram (som M30)		+	+	PGsl
M3		Spindelrotation medurs		+	+	PGsl
M4		Spindelrotation moturs		+	+	PGsl
M5		Spindel stopp		+	+	PGsl
M6		Verktygsväxling		+	+	PGsl
M17		Programslut underprogram		+	+	PGsl
M19		Spindelpositionering på den i SD43240 införda positionen		+	+	PGsl
M30		Programslut huvudprogram (som M2)		+	+	PGsl
M40		Automatisk växelkoppling		+	+	PGsl
M41 ... M45		Växlesteg 1 ... 5		+	+	PGsl
M70		Övergång i axeldrift		+	+	PGsl
MASLDEF	P	Definiera master/slave-axelförband		+	+	PGAsl
MASLDEL	P	Separera master/slave-axelförband och radera definitionen av förbandet		+	+	PGAsl
MASLOF	P	Koppla från en temporär koppling		+	+	PGAsl
MASLOFS	P	Koppla från en temporär koppling med automatiskt stopp av slave-axeln		+	+	PGAsl
MASLON	P	Koppla till en temporär koppling		+	+	PGAsl
MATCH	F	Söka en string i stringen		+	-	PGAsl
MAXVAL	F	Största värdet av två variabler (aritm. funktion)		+	+	PGAsl
MCALL	K	Modalt underprogramanrop		+		PGAsl
MEAC	K	Axial kontinuerlig mätning utan restvägsradering	s	+	+	PGAsl
MEAFRAME	F	Frame-beräkning ur mätpunkter		+	-	PGAsl
MEAS	A	Mäta med restvägsradering	s	+		PGAsl
MEASA	K	Axial mätning med restvägsradering	s	+	+	PGAsl
MEASURE	F	Beräkningsmetod för arbetsstycks- och verktygsmätning		+	-	FB1sl (M5)
MEAW	A	Mäta utan restvägsradering	s	+		PGAsl
MEAWA	K	Axial mätning utan restvägsradering	s	+	+	PGAsl
MI	K	Åtkomst till frame-data: Spegling		+		PGAsl
MINDEX	F	Bestämma index för ett tecken i ingångssträngen		+	-	PGAsl
MINVAL	F	Minsta värdet av två variabler (aritm. funktion)		+	+	PGAsl
MIRROR	G	Programmerbar spegling	s	+		PGAsl
MMC	P	Från detaljprogrammet interaktivt anropa dialogfönster på HMI		+	-	PGAsl
MOD	K	Modulo-division		+		PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
MODAXVAL	F	Fastställa modulo-position för en roterande modulo-axel		+	-	PGAsl
MOV	K	Starta positioneringsaxel		-	+	FBSYsl
MOVT	A	Ange slutpunkt för en förflyttningsrörelse i verktygsriktningen				FB1(K2)
MSG	P	Programmerbara meddelanden	m	+	-	PGsl
MVTOOL	P	Språkkommando för att flytta ett verktyg		+	-	FBWsl
N	A	NC-sidoblocknummer		+		PGsl
NAMETOINT	F	Fastställa systemvariabelindex		+		PGAsl
NC	K	Specificering av giltighetsområdet för data		+		PGAsl
NEWCONF	P	Överta ändrade maskindata (motsvarar "Sätta maskindatum verksamt")		+	-	PGAsl
NEWMT	F	Lägga till nytt Multitool		+	-	FBWsl
NEWT	F	Lägga till nytt verktyg		+	-	FBWsl
NORM ⁶⁾	G	Normalinställning i begynnelse-, slutpunkten vid verktygskompenseingen	m	+		PGsl
NOT	K	Logiskt INTE (negation)		+		PGAsl
NPROT	P	Maskinspecifikt skyddsområde TILL/FRÅN		+	-	PGAsl
NPROTDEF	P	Definition av ett maskinspecifikt skyddsområde		+	-	PGAsl
NUMBER	F	Omvandla ingångsstring till tal		+	-	PGAsl
OEMIPO1	G	OEM-interpolering 1	m	+		PGAsl
OEMIPO2	G	OEM-interpolering 2	m	+		PGAsl
OF	K	Nyckelord i CASE-förgreningen		+		PGAsl
OFFN	A	Avmått till den programmerade konturen	m	+		PGsl
OMA1	A	OEM-adress 1	m	+		PGAsl
OMA2	A	OEM-adress 2	m	+		PGAsl
OMA3	A	OEM-adress 3	m	+		PGAsl
OMA4	A	OEM-adress 4	m	+		PGAsl
OMA5	A	OEM-adress 5	m	+		PGAsl
OR	K	Logisk operator, ELLER-funktion		+		PGAsl
ORIXES	G	Linjär interpolering av maskinaxlar eller orienteringsaxlar	m	+		PGAsl
ORIXPOS	G	Orienteringsvinkel via virtuella orienteringsaxlar med positioner för roterande axlar	m	+		PGAsl
ORIC ⁶⁾	G	Orienteringsändringar vid ytterhörn överlagrar det cirkelblock som ska infogas	m	+		PGAsl
ORICONCCW	G	Interpolering på en cirkulär mantelyta moturs	m	+		PGAsl/FB3sl (F3)
ORICONCW	G	Interpolering på en cirkulär mantelyta medurs	m	+		PGAsl/FB3sl (F4)
ORICONIO	G	Interpolering på en cirkulär mantelyta med uppgift av en mellanorientering	m	+		PGAsl/FB3sl (F4)
ORICONTO	G	Interpolering på en cirkulär mantelyta i tangentiell övergång (uppgift av slutorientering)	m	+		PGAsl/FB3sl (F5)

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
ORICURVE	G	Interpolering av orienteringen med uppgift om rörelsen för två kontaktpunkter till verktyget	m	+		PGAsl/FB3sl (F6)
ORID	G	Orienteringsändringar utförs före cirkelblocket	m	+		PGAsl
ORIEULER ⁶⁾	G	Orienteringsvinkel via Euler-vinkel	m	+		PGAsl
ORIMKS	G	Verktygsorientering i maskinkoordinatsystemet	m	+		PGAsl
ORIPATH	G	Verktygsorientering relaterad till banan	m	+		PGAsl
ORIPATHS	G	Verktygsorientering relaterad till banan, en brytning i orienteringsförloppet jämnas ut	m	+		PGAsl
ORIPANE	G	Interpolering i ett plan (motsvarar ORIVECT) storcirkelinterpolering	m	+		PGAsl
ORIRESET	P	Grundläge för verktygsorienteringen med upp till 3 orienteringsaxlar		+	-	PGAsl
ORIROTA ⁶⁾	G	Vridvinkel för en absolut föreskriven vridriktning	m	+		PGAsl
ORIROTC	G	Tangentiell vridvektor till bantangenten	m	+		PGAsl
ORIROTR	G	Vridvinkel relativ till planet mellan start- och slutorientering	m	+		PGAsl
ORIROTT	G	Vridvinkel relativ till ändring av orienteringsvektorn	m	+		PGAsl
ORIRPY	G	Orienteringsvinkel via RPY-vinkel (XYZ)	m	+		PGAsl
ORIRPY2	G	Orienteringsvinkel via RPY-vinkel (ZYX)	m	+		PGAsl
ORIS	A	Orienteringsändring	m	+		PGAsl
ORISOF ⁶⁾	G	Jämning av orienteringsförloppet FRÅN	m	+		PGAsl
ORISOLH	F	Beräkna orienteringar		+		PGAsl
ORISON	G	Jämning av orienteringsförloppet TILL	m	+		PGAsl
ORIVECT ⁶⁾	G	Storcirkelinterpolering (identisk med ORIPANE)	m	+		PGAsl
ORIVIRT1	G	Orienteringsvinkel via virtuella orienteringsaxlar (definition 1)	m	+		PGAsl
ORIVIRT2	G	Orienteringsvinkel via virtuella orienteringsaxlar (definition 1)	m	+		PGAsl
ORIWKS ⁶⁾	G	Verktygsorientering i arbetsstyckskoordinatsystemet	m	+		PGAsl
OS	K	Pendla till/från		+		PGAsl
OSB	K	Pending: Startpunkt	m	+		FB1sl (P5)
OSC	G	Konstant jämning verktygsorientering	m	+		PGAsl
OSCILL	K	Axis: 1 - 3 ansättningsaxlar	m	+		PGAsl
OSCTRL	K	Optioner pendla	m	+		PGAsl
OSD	G	Översläpning av verktygsorienteringen genom angivande av översläpningslängden med SD	m	+		PGAsl
OSE	K	Pendla slutpunkt	m	+		PGAsl
OSNSC	K	Pending: Antal renbränningar	m	+		PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
OSOF ⁶⁾	G	Jämning av verktygsorientering FRÅN	m	+		PGAsI
OSP1	K	Pendling: vänster vändpunkt	m	+		PGAsI
OSP2	K	Pendling höger vändpunkt	m	+		PGAsI
OSS	G	Jämning av verktygsorientering vid blockslut	m	+		PGAsI
OSSE	G	Jämning av verktygsorientering vid blockbörjan och blockslut	m	+		PGAsI
OST	G	Översläpning av verktygsorienteringen genom angivande av vinkeltoleransen i grader med SD (maximal avvikelse från det programmerade orienteringsförloppet)	m	+		PGAsI
OST1	K	Pendling: Hållningspunkt i vänster vändpunkt	m	+		PGAsI
OST2	K	Pendling: Hållningspunkt i höger vändpunkt	m	+		PGAsI
OTOL	A	Orienteringstolerans för kompressor-funktioner, orienteringsjämning och översläpningstyper	m	+		PGAsI
OVR	K	Varvtalskorrigering	m	+		PGAsI
OVRA	K	Axial varvtalskorrigering	m	+	+	PGAsI
OVRRAP	K	Snabbgångs-korrigering	m	+		PGAsI
P	A	Antal underprogramkörningar		+		PGAsI
PAROT	G	Upprikta arbetsstyckskoordinatsystemet mot arbetsstycket	m	+		PGsI
PAROTOF ⁶⁾	G	Koppla från arbetsstycksrelaterad frame-vridning	m	+		PGsI
PCALL	K	Anropa underprogram med absolut sökvägsuppgift och parameteröverföring		+		PGAsI
PDELAYOF	G	Fördröjning vid stansning FRÅN	m	+		PGAsI
PDELAYON ⁶⁾	G	Fördröjning vid stansning TILL	m	+		PGAsI
PHI	K	Vridvinkel för orienteringen runt riktningsaxeln till konen		+		PGAsI
PHU	K	Fysikalisk enhet för en variabel		+		PGAsI
PL	A	1. B-spline: Nod-avstånd 2. Polynom-interpolering: Längd på parameterintervallet vid polynom-interpolering	s	+		PGAsI
PM	K	per minut		+		PGsI
PO	K	Polynomkoefficient vid polynom-interpolering	s	+		PGAsI
POCKET3	C (T)	Fräsning av fyrkantficka		+		PGAsI
POCKET4	C (T)	Fräsning av cirkelficka		+		PGAsI
POLF	K	Återgångsposition LIFTFAST	m	+		PGsI/PGAsI
POLFA	P	Starta återgångsposition från singelaxlar med \$AA_ESR_TRIGGER	m	+	+	PGsI
POLFMASK	P	Frige axlar för återgång utan samband mellan axlarna	m	+	-	PGsI
POLFMLIN	P	Frige axlar för återgång med linjärt samband mellan axlarna	m	+	-	PGsI
POLY	G	Polynom-interpolering	m	+		PGAsI

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
POLYPATH	P	Polynom-interpolering selekterbar för axel-grupperna AXIS eller VECT	m	+	-	PGAsl
PON	G	Stansning TILL	m	+		PGAsl
PONS	G	Stansning TILL i IPO-takt	m	+		PGAsl
POS	K	Positionera axel		+	+	PGsl
POSA	K	Positionera axel över blockgräns		+	+	PGsl
POSM	P	Positionera magasin		+	-	FBWsl
POSMT	P	Positionera Multitool på WZ-hållare på plat-snummer		+	-	FBWsl
POSP	K	Positionera i delstycken (pendla)		+		PGsl
POSRANGE	F	Fastställa om den aktuellt interpolerade bör-positionen för en axel befinner sig i ett fönster runt en föreskriven referensposition		+	+	FBSYsl
POT	F	Kvadrat (aritmetisk funktion)		+	+	PGAsl
PR	K	Per varv		+		PGsl
PREPRO	PA	Känneteckna underprogram med förberedel-se		+		PGAsl
PRESETON	P	Ärvärdesinställning med förlust av referens-status		+	+	PGAsl
PRESETONS	P	Ärvärdesinställning utan förlust av referens-status		+	+	PGAsl
PRIO	K	Nyckelord för sättning av prioriteten vid be-handlingen av interrupts		+		PGAsl
PRLOC	K	Initiering av variablerna vid Reset endast efter lokal ändring		+		PGAsl
PROC	K	Första anvisningen i ett program		+		PGAsl
PROTA	P	Kräva nyberäkning av kollisionsmodellen		+		PGAsl
PROTD	F	Beräkna avstånd mellan två skyddsområden		+		PGAsl
PROTS	P	Sätta skyddsområdestillståndet		+		PGAsl
PSI	K	Öppningsvinkel för konen		+		PGAsl
PTP	G	Punkt-till-punkt-rörelse (PTP-körning)	m	+		PGAsl
PTPG0	G	Punkt-till-punkt-rörelse endast vid G0, annars banrörelse CP	m	+		PGAsl
PTPWOC	G	Punkt-till-punkt-rörelse utan utjämningsrörel-ser, som har förorsakats av orienteringsändr-ingar	m	+		PGAsl
PUNCHACC	P	Vägberoende acceleration vid nibbling		+	-	PGAsl
PUTFTOC	P	Verktygfsinkompensering för parallell slipning		+	-	PGAsl
PUTFTOCF	P	Verktygfsinkompensering beroende av en med FCTDEF fastlagd funktion för parallell slipning		+	-	PGAsl
PW	A	B-spline, punkt-vikt	s	+		PGAsl
QU	K	Snabb extra-(hjälp-)funktionsutmatning		+		PGsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
R...	A	Räkneparameter också som inställbar axelbe-teckning och med numeriskt tillägg		+		PGAsI
RAC	K	Absolut blockvis axelspecifik radieprogram-mering	s	+		PGsI
RDISABLE	P	Inläsningsspärr		-	+	FBSYsI
READ	P	Läser i den angivna filen in en eller flera rader och lägger den lästa informationen i fältet		+	-	PGAsI
REAL	K	Datatyp: Flyttalsvariabel med förtecken (reella tal)		+		PGAsI
REDEF	K	Redefinition av systemvariabler, användarva-riabler och NC-språkkommandon		+		PGAsI
RELEASE	P	Frige maskindata för axelbyte		+	+	PGAsI
REP	K	Nyckelord för initiering av alla element i ett fält med samma värde		+		PGAsI
REPEAT	K	Upprepning av en programslinga		+		PGAsI
REPEATB	K	Upprepning av en programrad		+		PGAsI
REPOSA	G	Återstart till konturen linjär med alla axlar	s	+		PGAsI
REPOSH	G	Återstart på konturen med halvcirkel	s	+		PGAsI
REPOSHA	G	Återstart på konturen med alla axlar; geomet-riaxlar i halvcirkel	s	+		PGAsI
REPOSL	G	Återstart till konturen linjär	s	+		PGAsI
REPOSQ	G	Återstart till konturen i fjärdedels cirkel	s	+		PGAsI
REPOSQA	G	Återstart på konturen linjär med alla axlar; geometriaxlar i fjärdedels cirkel	s	+		PGAsI
RESETMON	P	Språkkommando för börvärdesaktivering		+	-	FBWsI
RET	P	Underprogramslut		+	+	PGAsI
RETB	P	Underprogramslut		+	+	PGAsI
RIC	K	Relativ blockvis axelspecifik radieprogram-mering	s	+		PGsI
RINDEX	F	Bestämma index för ett tecken i ingångsstr-ingen		+	-	PGAsI
RMB	G	Återstart till blockbegränsningspunkt	m	+		PGAsI
RMBBL	G	Återstart till blockbegränsningspunkt	s	+		PGAsI
RME	G	Återstart till blockslutpunkt	m	+		PGAsI
RMEBL	G	Återstart till blockslutpunkt	s	+		PGAsI
RMI ⁶⁾	G	Återstart till stoppunkt	m	+		PGAsI
RMIBL ⁶⁾	G	Återstart till stoppunkt	s	+		PGAsI
RMN	G	Återstart till närmast liggande banpunkt	m	+		PGAsI
RMNBL	G	Återstart till närmast liggande banpunkt	s	+		PGAsI
RND	A	Runda konturhörn	s	+		PGsI
RNDM	A	Modal avrundning	m	+		PGsI
ROT	G	Programmerbar vridning	s	+		PGsI
ROTS	G	Programmerbara frame-vridningar med rymd-vinklar	s	+		PGsI

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
ROUND	F	Avrundning av siffror efter komma		+	+	PGAsl
ROUNDUP	F	Avrunda ett inmatningsvärde uppåt		+	+	PGAsl
RP	A	Polarradie	m/s	+		PGsl
RPL	A	Vridning i planet	s	+		PGsl
RT	K	Parameter för åtkomst till framedata: Vridning		+		PGAsl
RTLIOF	G	G0 utan linjärinterpolering (singelaxelinterpolering)	m	+		PGsl
RTLION ⁶⁾	G	G0 med linjärinterpolering	m	+		PGsl

Anvisningar S ... Z

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
S	A	Spindelvarvtal (vid G4, G96/G961 annan betydelse)	m/s	+	+	PGsl
SAVE	PA	Attribut för räddning av informationer vid underprogramanrop		+		PGAsl
SBLOF	P	Undertrycka blockvis		+	-	PGAsl
SBLON	P	Upphåva undertryckning av blockvis		+	-	PGAsl
SC	K	Parameter för åtkomst till framedata: Skalning		+		PGAsl
SCALE	G	Programmerbar skalning	s	+		PGsl
SCC	K	Selektiv tillordning av en planaxel till G96/G961/G962. Axelbeteckningar kan vara geo-, kanal- eller maskinaxel.		+		PGsl
SCPARA	K	Programmera servo-parameterblock		+	+	PGAsl
SD	A	Spline -grad	s	+		PGAsl
SET	K	Nyckelord för initiering av alla element i ett fält med uppräknade värden		+		PGAsl
SETAL	P	Sätta larm		+	+	PGAsl
SETDNO	F	Tillordna D-nummer för ett skär (CE) till ett verktyg (T)		+	-	PGAsl
SETINT	K	Fastlägga vilken interruptrutin som ska aktiveras när en NC-ingång väntar		+		PGAsl
SETM	P	Sätta flaggor i den egna kanalen		+	+	PGAsl
SETMS	P	Koppla tillbaka till den i maskindatum fastlagda masterspindeln		+	-	PGsl
SETMS(n)	P	Spindel n ska gälla som masterspindel		+		PGsl
SETMTH	P	Sätta verktygshållarnummer för master		+	-	FBWsl
SETPIECE	P	Ta hänsyn till stycktal för alla verktyg som är tillordnade spindeln.		+	-	FBWsl
SETTA	P	Sätta verktyg från förslitningsförband aktivt		+	-	FBWsl
SETTCOR	F	Förändring av verktygskomponenter under hänsynstagande till alla randvillkor		+	-	PGAsl
SETTIA	P	Sätta verktyg från förslitningsförband inaktivt		+	-	FBWsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
SF	A	Startpunktförskjutning för gängskärning	m	+		PGsl
SIN	F	Sinus (trigon. funktion)		+	+	PGAsl
SIRELAY	F	Aktivera de med SIRELIN, SIRELOUT och SIRELTIME parameterade säkerhetsfunktionerna		-	+	FBSlsl
SIRELIN	P	Initiera ingångsstorheterna i funktionsblocket		+	-	FBSlsl
SIRELOUT	P	Initiera utgångsstorheterna i funktionsblocket		+	-	FBSlsl
SIRELTIME	P	Initiera timer i funktionsblocket		+	-	FBSlsl
SLOT1	C (T)	Längsspår		+		PGAsl
SLOT2	C (T)	Cirkelspår		+		PGAsl
SOFT	G	Ryckbegränsad banacceleration	m	+		PGAsl
SOFTA	P	Koppla till ryckbegränsad axelacceleration för de programmerade axlarna		+	-	PGAsl
SON	G	Nibbling TILL	m	+		PGAsl
SONS	G	Nibbling TILL i IPO-takt	m	+		PGAsl
SPATH ⁶⁾	G	Banreferens för FGROUPE-axlar är båglängd	m	+		PGAsl
SPCOF	P	Koppla om masterspindel eller spindel (n) från lägesreglering till varvtalsreglering	m	+	-	PGsl
SPCON	P	Koppla om masterspindel eller spindel (n) från varvtalsreglering till lägesreglering	m	+	-	PGAsl
SPI	F	Konverterar spindelnummer till axelbeteckning		+	-	PGAsl
SPIF1 ⁶⁾	G	Snabba NC-in-/utgångar för stansning/nibbling byte 1	m	+		FB2sl (N4)
SPIF2	G	Snabba NC-in-/utgångar för stansning/nibbling byte 2	m	+		FB2sl (N4)
SPLINEPATH	P	Fastlägga spline-förband		+	-	PGAsl
SPN	A	Antal delsträckor per block	s	+		PGAsl
SPOF ⁶⁾	G	Slag FRÅN, stansning, nibbling FRÅN	m	+		PGAsl
SPOS	K	Spindelposition	m	+	+	PGsl
SPOSA	K	Spindelposition utöver blockgränser	m	+		PGsl
SPP	A	Längd för en delsträcka	m	+		PGAsl
SPRINT	F	Levererar en ingångssträng formaterad tillbaka		+		PGAsl
SQRT	F	Kvadratroten (aritmetisk funktion) (square root)		+	+	PGAsl
SR	A	Pendelåtergångsväg för synkronaktion	s	+		PGsl
SRA	K	Pendelåtergångsväg vid extern ingång axial för synkronaktion	m	+		PGsl
ST	A	Pendelrenbränningstid för synkronaktion	s	+		PGsl
STA	K	Pendelrenbränningstid axial för synkronaktion	m	+		PGsl
START	P	Start av de valda programmen i flera kanaler samtidigt från det löpande programmet		+	-	PGAsl

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
STARTFIFO ⁶⁾	G	Genomarbета; parallellt därtill fylla förkörningsbufferten	m	+		PGAsl
STAT		Länkarnas position	s	+		PGAsl
STOLF	K	G0-toleransfaktor	m	+		PGAsl
STOPFIFO	G	Stoppande av bearbetningen; fyllning av förkörningsbufferten tills STARTFIFO identifieras, förkörningsbufferten full eller programslut	m	+		PGAsl
STOPRE	P	Fördekoderingsstopp, tills alla förberedda block har genomarbetats i huvudkörningen		+	-	PGAsl
STOPREOF	P	Upphåva fördekoderingsstopp		-	+	FBSYsl
STRING	K	Datotyp: Teckenkedja		+		PGAsl
STRINGIS	F	Kontrollerar förefintlig NC-språkomfattning och speciellt för detta kommando tillhörande NC-cykelnamn, användarvariabler, makros och labelnamn, om dessa existerar, är giltiga, definierade eller aktiva.		+	-	PGAsl
STRLEN	F	Bestämma längd för en string		+	-	PGAsl
SUBSTR	F	Bestämma index för ett tecken i ingångssträngen		+	-	PGAsl
SUPA	G	Undertryckning av den aktuella nollpunktsförflyttningen, inklusive programmerade förflyttningar, systemframes, handrattsförflyttningar (DRF), extern nollpunktsförflyttning och överlagrad rörelse	s	+		PGsl
SVC	K	Verktögs-skärhastighet	m	+		PGsl
SYNFCT	P	Utvärdering av ett polynom beroende på ett villkor i rörelse-synkronaktionen		-	+	FBSYsl
SYNR	K	Läsning av variablen sker synkront, dvs. vid tidpunkten för genomarbetningen		+		PGAsl
SYNRW	K	Läsning och skrivning av variablen sker synkront, dvs. vid tidpunkten för genomarbetningen		+		PGAsl
SYNW	K	Skrivning av variablen sker synkront, dvs. vid tidpunkten för genomarbetningen		+		PGAsl
T	A	Anropa verktyg (växling endast när fastlagt i maskindatum; annars M6-kommando nödvändigt)		+		PGsl
TAN	F	Tangens (trigon. funktion)		+	+	PGAsl
TANG	P	Tangentialstyrning: Definiera koppling		+	-	PGAsl
TANGDEL	P	Tangentialstyrning: Radera koppling		+	-	PGAsl
TANGOF	P	Tangentialstyrning: Koppla från koppling		+	-	PGAsl
TANGON	P	Tangentialstyrning: Koppla till koppling		+	-	PGAsl
TCA (828D: _TCA)	P	Verktögsval / verktygsväxling oberoende av verktygets status		+	-	FBWsl
TCARR	A	Begära verktygsbärare (nummer "m")		+		PGAsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
TCI	P	Växla verktyg från mellanlagringen till magasinet		+	-	FBWsl
TCOABS ⁶⁾	G	Bestämma verktygslängdkomponenterna ur den aktuella verktygsorienteringen	m	+		PGAsl
TCOFR	G	Bestämma verktygslängdkomponenterna ur orienteringen för den aktiva framen	m	+		PGAsl
TCOFRX	G	Bestämma verktygsorienteringen för en aktiv frame vid verktygsvalet, verktyg pekar i X-riktningen	m	+		PGAsl
TCOFRY	G	Bestämma verktygsorienteringen för en aktiv frame vid verktygsvalet, verktyg pekar i Y-riktningen	m	+		PGAsl
TCOFRZ	G	Bestämma verktygsorienteringen för en aktiv frame vid verktygsvalet, verktyg pekar i Z-riktningen	m	+		PGAsl
THETA	A	Vridningsvinkel	s	+		PGAsl
TILT	A	Sidledsvinkel	m	+		PGAsl
TLIFT	P	Tangentialstyrning: Koppla till mellanblocks-generering		+	-	PGAsl
TML	P	Verktygsval med magasin-platsnummer		+	-	FBWsl
TMOF	P	Välja bort verktygsövervakning		+	-	PGAsl
TMON	P	Välja verktygsövervakning		+	-	PGAsl
TO	K	Betecknar slutvärdet i en FOR-räkneslinga		+		PGAsl
TOFF	K	Verktygslängd-offset i riktningen för den verktygslängdkomponent, som verkar parallellt till den i index angivna geometriaxeln.	m	+		PGsl
TOFFL	K	Verktygslängd-offset i riktningen för verktygslängdkomponenten L1, L2 resp. L3	m	+		PGsl
TOFFOF	P	Återställa online-verktygslängdkompensering		+	-	PGAsl
TOFFON	P	Aktivera online-verktygslängdkompensering		+	-	PGAsl
TOFFR	A	Verktygsradie-offset	m	+		PGsl
TOFRAME	G	Rikta upp Z-axeln i WKS med frame-rotation parallellt till verktygsorienteringen	m	+		PGsl
TOFRAMEX	G	Rikta upp X-axeln i WKS med frame-rotation parallellt till verktygsorienteringen	m	+		PGsl
TOFRAMEY	G	Rikta upp Y-axeln i WKS med frame-rotation parallellt till verktygsorienteringen	m	+		PGsl
TOFRAMEZ	G	som TOFRAME	m	+		PGsl
TOLOWER	F	Omvandla bokstäver i en string till små bokstäver		+	-	PGAsl
TOOLENV	F	Spara alla aktuella tillstånd som har betydelse för värderingen av de i minnet lagrade verktygsdata		+	-	PGAsl
TOOLGNT	F	Fastställa antalet verktyg i en verktygsgrupp		+	-	FBWsl
TOOLGT	F	Fastställa T-nummer för ett verktyg ur en verktygsgrupp		+	-	FBWsl

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
TOROT	G	Rikta upp Z-axeln i WKS med frame-rotation parallellt till verktygsorienteringen	m	+		PGsl
TOROTOF ⁶⁾	G	Framevridningar i verktygssriktning FRÅN	m	+		PGsl
TOROTX	G	Rikta upp X-axeln i WKS med frame-rotation parallellt till verktygsorienteringen	m	+		PGsl
TOROTY	G	Rikta upp Y-axeln i WKS med frame-rotation parallellt till verktygsorienteringen	m	+		PGsl
TOROTZ	G	som TOROT	m	+		PGsl
TOUPPER	F	Omvandla bokstäver i en string till stora bokstäver		+	-	PGAsl
TOWBCS	G	Förslitningsvärden i baskoordinatsystemet (BKS)	m	+		PGAsl
TOWKCS	G	Förslitningsvärden i koordinatsystemet för verktygshuvudet vid kinetisk transformation (skiljer sig från MKS genom verktygsvridning)	m	+		PGAsl
TOWMCS	G	Förslitningsvärden i maskinkoordinatsystemet (MKS)	m	+		PGAsl
TOWSTD ⁶⁾	G	Grundlägesvärde för kompenseringar i verktygslängden	m	+		PGAsl
TOWTCS	G	Förslitningsvärden i verktygs-koordinatsystemet (verktygsbärar-referenspunkt T vid verktygshållar-infästningen)	m	+		PGAsl
TOWWCS	G	Förslitningsvärden i arbetsstyckskoordinatsystemet (WKS)	m	+		PGAsl
TR	K	Förflyttnings-komponent för en frame-variabel		+		PGAsl
TRAANG	P	Transformation lutande axel		+	-	PGAsl
TRACON	P	Kaskaderad transformation		+	-	PGAsl
TRACYL	P	Cylinder: Mantelyts-transformation		+	-	PGAsl
TRAFOOF	P	Koppla från i kanalen aktiva transformationer		+	-	PGAsl
TRAFOON	P	Koppla till med kinematiska följder definierade transformationer		+	-	PGAsl
TRAILOF	P	Axelsynkron medsläpning FRÅN		+	+	PGAsl
TRAILON	P	Asynkron medsläpning TILL		+	+	PGAsl
TRANS	G	Absolut programmerbar nollpunktsförflyttning	s	+		PGsl
TRANSMIT	P	Polar-transformation (planbearbetning)		+	-	PGAsl
TRAORI	P	4-, 5-axeltransformation, generisk transformation		+	-	PGAsl
TRUE	K	Logisk konstant: sann		+		PGAsl
TRUNC	F	Avskärning av siffror efter komma		+	+	PGAsl
TU		Axelvinkel	s	+		PGAsl
TURN	A	Antal vindlar för skruvlinje	s	+		PGsl
ULI	K	Övre gränsvärde för variabel		+		PGAsl
UNLOCK	P	Frige synkronaktion med ID (fortsätta teknologicykel)		-	+	FBSYsl

21.1 Anvisningar

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
1) 2) 3) 4) 5) Förklaringar se teckenförklaring (Sida 852).						
UNTIL	K	Villkor för att avsluta en REPEAT-slinga		+		PGAsI
UPATH	G	Banreferens för FGROUP-axlar är kurvparametrar	m	+		PGAsI
VAR	K	Nyckelord: Typ av parameteröverföring		+		PGAsI
VELOLIM	K	Reduktion av den maximala axiala hastigheten	m	+		PGAsI
VELOLIMA	K	Reduktion eller förhöjning av den maximala axiala hastigheten för följdaxeln	m	+	+	PGAsI
WAITC	P	Vänta till kopplingsblockbytes-kriteriet för axlar/spindlar är uppfyllt		+	-	PGAsI
WAITE	P	Vänta på programslut i en annan kanal.		+	-	PGAsI
WAITENC	P	Vänta på synkroniserade resp. restaurerade axelpositioner		+	-	PGAsI
WAITM	P	Vänta på flagga i den angivna kanalen; avsluta föregående block med precisionsstopp.		+	-	PGAsI
WAITMC	P	Vänta på flagga i den angivna kanalen; precisionsstopp endast när de andra kanalerna ännu inte har nått flaggan.		+	-	PGAsI
WAITP	P	Vänta på förflyttningsslut för positioneringsaxeln		+	-	PGsI
WAITS	P	Vänta på att uppnå spindelposition		+	-	PGsI
WALCS0 ⁶⁾	G	WKS-arbetsfältsbegränsning bortvald	m	+	-	PGsI
WALCS1	G	WKS-arbetsfältsbegränsningsgrupp 1 aktiv	m	+	-	PGsI
WALCS2	G	WKS-arbetsfältsbegränsningsgrupp 2 aktiv	m	+	-	PGsI
WALCS3	G	WKS-arbetsfältsbegränsningsgrupp 3 aktiv	m	+	-	PGsI
WALCS4	G	WKS-arbetsfältsbegränsningsgrupp 4 aktiv	m	+	-	PGsI
WALCS5	G	WKS-arbetsfältsbegränsningsgrupp 5 aktiv	m	+	-	PGsI
WALCS6	G	WKS-arbetsfältsbegränsningsgrupp 6 aktiv	m	+	-	PGsI
WALCS7	G	WKS-arbetsfältsbegränsningsgrupp 7 aktiv	m	+	-	PGsI
WALCS8	G	WKS-arbetsfältsbegränsningsgrupp 8 aktiv	m	+	-	PGsI
WALCS9	G	WKS-arbetsfältsbegränsningsgrupp 9 aktiv	m	+	-	PGsI
WALCS10	G	WKS-arbetsfältsbegränsningsgrupp 10 aktiv	m	+	-	PGsI
WALIMOF	G	BKS-arbetsfältsbegränsning FRÅN	m	+	-	PGsI
WALIMON ⁶⁾	G	BKS-arbetsfältsbegränsning TILL	m	+	-	PGsI
WHEN	K	Aktionen utförs en gång, när villkoret är uppfyllt en gång.		-	+	FBSYsI
WHENEVER	K	Aktionen utförs cykliskt i varje interpolatortakt, när villkoret är uppfyllt.		-	+	FBSYsI
WHILE	K	Början på WHILE-programslingan		+		PGAsI
WRITE	P	Skriva text i filsystemet. Infogar ett block i slutet av den angivna filen.		+	-	PGAsI
WRTPR	P	Skriva teckenföljd i BTSS-variabel (WRTPR)		+	-	PGsI
X	A	Axelnamn	m/s	+	+	PGsI
XOR	O	Logisk exklusiv- ELLER		+		PGAsI

Anvisning	Typ ¹⁾	Betydelse	W ²⁾	TP ³⁾	SA ⁴⁾	Beskrivning se ⁵⁾
^{1) 2) 3) 4) 5)} Förklaringar se teckenförklaring (Sida 852).						
Y	A	Axelnamn	m/s	+	+	PGsl
Z	A	Axelnamn	m/s	+	+	PGsl

Bildtext

1) Typ av anvisning:

A Adress

Beteckning som tillordnas ett värde (t. ex. OVR=10). Det finns också några adresser, som kopplar till eller från en funktion utan värdetilldelning (t. ex. CPLON och CPLOF).

C (A) AST-cykel

Fördefinierat NC-program för den automatiska efteroptimeringen med AST (= Automatisk Servo Optimering). Anpassningen till den konkreta optimeringssituationen sker via parametrar, som överlämnas till cykeln vid anrop.

C (M) Mätcykel

Fördefinierat NC-program, i vilket ett visst mätförlopp, som t. ex. bestämning av innerdiametern för ett cylindriskt arbetsstycke, är programmerat allmängiltigt. Anpassningen till den konkreta mätsituationen sker via parametrar, som överlämnas till cykeln vid anrop.

C (T) Teknologisk cykel

Fördefinierat NC-program, i vilket ett visst bearbetningsförlopp, som t. ex. borrar av en gänga eller fräsning av en ficka, är programmerat allmängiltigt. Anpassningen till den konkreta bearbetningssituationen sker via parametrar, som överlämnas till cykeln vid anrop.

F Fördefinierad funktion (levererar returvärde)

Anropet av den fördefinierade funktionen kan stå som operand i uttrycket.

G G-kommando

G-kommandona är indelade i grupper. Det kan bara skrivas ett G-kommando från en grupp i ett block. Ett G-kommando kan vara modalt verksamt (tills annullering genom ett annat kommando i samma grupp), eller det är endast verksamt för det block i vilket det står (blockvis verksamt).

K Nyckelord

Beteckning som bestämmer syntaxen för ett block. Ett nyckelord tilldelas inget värde och med ett nyckelord kan inte heller någon NC-funktion kopplas till/från.

Exempel: Kontrollstrukturer (IF, ELSE, ENDIF, WHEN, ...), programförlopp (GOTOB, GOTO, RET ...)

O Operator

Operator för en matematisk, jämförelse- eller logisk operation

P Fördefinierad procedur (levererar inte något returvärde)

PA Programattribut

Programattribut står i slutet av definitionsraden i ett underprogram:

```
PROC <Programmname>(...) <Programmattribut>
```

De bestämmer beteendet vid körningen av underprogrammet.

2) Anvisningens verkan:

m modal

s blockvis

3) Programmermöjlighet i detaljprogram:

+ programmerbar

- inte programmerbar

M endast programmerbar av maskintillverkaren

- 4) Programmermöjlighet i synkronaktioner:
- + programmerbar
 - inte programmerbar
 - T programmerbar endast i teknologicykler
- 5) Hänvisar till det dokument som innehåller den utförliga beskrivningen av anvisningen:
- | | |
|------------------|--|
| <i>PGsl</i> | Programmeringshandbok Grunder |
| <i>PGAsl</i> | Programmeringshandbok Arbetsförberedelse |
| <i>BNMsl</i> | Programmeringshandbok Mätcykler: |
| <i>BHDsl</i> | Användarhandbok Svarva |
| <i>BHFsl</i> | Användarhandbok Fräsa |
| <i>FB1sl</i> () | Funktionshandbok Grundfunktioner (med alfanumerisk förkortning för den ifrågavarande funktionsbeskrivningen inom parentes) |
| <i>FB2sl</i> () | Funktionshandbok Tilläggfunktioner (med alfanumerisk förkortning för den ifrågavarande funktionsbeskrivningen inom parentes) |
| <i>FB3sl</i> () | Funktionshandbok Specialfunktioner (med alfanumerisk förkortning för den ifrågavarande funktionsbeskrivningen inom parentes) |
| <i>FBSsl</i> | Funktionshandbok Safety Integrated |
| <i>FBSYsl</i> | Funktionshandbok Synkronaktioner |
| <i>FBWsl</i> | Funktionshandbok Verktygsförvaltning |
- 6) Standardinställning vid programbörjan (i styrningens leveranstillstånd, om inget annat är programmerat).

Bild 21-1 Betydelse av fotnoter i anvisningstabellerna

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Märk

Cykler

Cyklerna markeras som "option", när de beror av följande licenspliktiga optioner:

- Utökade teknologifunktioner (artikelnummer: 6FC5800-0AP58-0YB0)
- Mätcykler (artikelnummer: 6FC5800-0AP28-0YB0)
- Mäta kinematik (artikelnummer: 6FC5800-0AP18-0YB0)
- SINUMERIK Slipa Advanced (artikelnummer: 6FC5800-0AS35-0YB0)

Det markerades inte, när cykler genom optionen "Utökade teknologifunktioner" i vissa system endast innehåller delfunktionaliteter.

21.2.1 Styrningsvarianter fräsa / svarva

Anvisningar A ... C

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
:	•	•	•	•	•	•	•	•
*	•	•	•	•	•	•	•	•
+	•	•	•	•	•	•	•	•
-	•	•	•	•	•	•	•	•
<	•	•	•	•	•	•	•	•
<<	•	•	•	•	•	•	•	•
<=	•	•	•	•	•	•	•	•
=	•	•	•	•	•	•	•	•
>=	•	•	•	•	•	•	•	•
/	•	•	•	•	•	•	•	•
/0 ... /7	•	•	•	•	•	•	•	•
A	•	•	•	•	•	•	•	•
A2	-	-	-	-	-	-	-	-
A3	-	-	-	-	-	-	-	-
A4	-	-	-	-	-	-	-	-
A5	-	-	-	-	-	-	-	-
A6	-	-	-	-	-	-	-	-
A7	-	-	-	-	-	-	-	-
ABS	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
AC	•	•	•	•	•	•	•	•
ACC	•	•	•	•	•	•	•	•
ACCLIMA	•	•	•	•	•	•	•	•
ACN	•	•	•	•	•	•	•	•
ACOS	•	•	•	•	•	•	•	•
ACP	•	•	•	•	•	•	•	•
ACTBLOCNO	•	•	•	•	•	•	•	•
ADDFRAME	•	•	•	•	•	•	•	•
ADIS	•	•	•	•	•	•	•	•
ADISPOS	•	•	•	•	•	•	•	•
ADISPOSA	•	•	•	•	•	•	•	•
ALF	•	•	•	•	•	•	•	•
AMIRROR	•	•	•	•	•	•	•	•
AND	•	•	•	•	•	•	•	•
ANG	•	•	•	•	•	•	•	•
AP	•	•	•	•	•	•	•	•
APR	•	•	•	•	•	•	•	•
APRB	•	•	•	•	•	•	•	•
APRP	•	•	•	•	•	•	•	•
APW	•	•	•	•	•	•	•	•
APWB	•	•	•	•	•	•	•	•
APWP	•	•	•	•	•	•	•	•
APX	•	•	•	•	•	•	•	•
AR	•	•	•	•	•	•	•	•
AROT	•	•	•	•	•	•	•	•
AROTS	•	•	•	•	•	•	•	•
AS	•	•	•	•	•	•	•	•
ASCALE	•	•	•	•	•	•	•	•
ASIN	•	•	•	•	•	•	•	•
ASPLINE	○	○	○	○	○	○	○	○
ATAN2	•	•	•	•	•	•	•	•
ATOL	•	•	•	•	•	•	•	•
ATRANS	•	•	•	•	•	•	•	•
AUXFUDEL	•	•	•	•	•	•	•	•
AUXFUDELG	•	•	•	•	•	•	•	•
AUXFUMSEQ	•	•	•	•	•	•	•	•
AUXFUSYNC	•	•	•	•	•	•	•	•
AX	•	•	•	•	•	•	•	•
AXCTSWE	-	-	-	-	-	-	-	-

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
AXCTSWEC	-	-	-	-	-	-	-	-
AXCTSWED	-	-	-	-	-	-	-	-
AXIS	•	•	•	•	•	•	•	•
AXNAME	•	•	•	•	•	•	•	•
AXSTRING	•	•	•	•	•	•	•	•
AXTOCHAN	•	•	•	•	•	•	•	•
AXTOSPI	•	•	•	•	•	•	•	•
B	•	•	•	•	•	•	•	•
B2	-	-	-	-	-	-	-	-
B3	-	-	-	-	-	-	-	-
B4	-	-	-	-	-	-	-	-
B5	-	-	-	-	-	-	-	-
B6	-	-	-	-	-	-	-	-
B7	-	-	-	-	-	-	-	-
B_AND	•	•	•	•	•	•	•	•
B_OR	•	•	•	•	•	•	•	•
B_NOT	•	•	•	•	•	•	•	•
B_XOR	•	•	•	•	•	•	•	•
BAUTO	○	○	○	○	○	○	○	○
BLOCK	•	•	•	•	•	•	•	•
BLSYNC	•	•	•	•	•	•	•	•
BNAT	○	○	○	○	○	○	○	○
BOOL	•	•	•	•	•	•	•	•
BOUND	•	•	•	•	•	•	•	•
BRISK	•	•	•	•	•	•	•	•
BRISKA	•	•	•	•	•	•	•	•
BSPLINE	○	○	○	○	○	○	○	○
BTAN	○	○	○	○	○	○	○	○
C	•	•	•	•	•	•	•	•
C2	-	-	-	-	-	-	Kanalaxel- namn	Kanalaxel- namn
C3	-	-	-	-	-	-	-	-
C4	-	-	-	-	-	-	-	-
C5	-	-	-	-	-	-	-	-
C6	-	-	-	-	-	-	-	-
C7	-	-	-	-	-	-	-	-
CAC	•	•	•	•	•	•	•	•
CACN	•	•	•	•	•	•	•	•
CACP	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
CALCDAT	•	•	•	•	•	•	•	•
CALCPOSI	•	•	•	•	•	•	•	•
CALL	•	•	•	•	•	•	•	•
CALLPATH	•	•	•	•	•	•	•	•
CANCEL	•	•	•	•	•	•	•	•
CASE	•	•	•	•	•	•	•	•
CDC	•	•	•	•	•	•	•	•
CDOF	-	-	-	-	-	-	-	-
CDOF2	-	-	-	-	-	-	-	-
CDON	-	-	-	-	-	-	-	-
CFC	•	•	•	•	•	•	•	•
CFIN	•	•	•	•	•	•	•	•
CFINE	•	•	•	•	•	•	•	•
CFTCP	•	•	•	•	•	•	•	•
CHAN	•	•	•	•	•	•	•	•
CHANDATA	•	•	•	•	•	•	•	•
CHAR	•	•	•	•	•	•	•	•
CHF	•	•	•	•	•	•	•	•
CHKDM	•	•	•	•	•	•	•	•
CHKDNO	•	•	•	•	•	•	•	•
CHR	•	•	•	•	•	•	•	•
CIC	•	•	•	•	•	•	•	•
CIP	•	•	•	•	•	•	•	•
CLEARM	-	-	-	-	-	○	-	•
CLRINT	•	•	•	•	•	•	•	•
CMIRROR	•	•	•	•	•	•	•	•
COARSEA	•	•	•	•	•	•	•	•
COLLPAIR	-	-	-	-	-	-	-	-
COMPCAD	•	-	•	-	•	•	-	-
COMPCURV	•	-	•	-	•	•	-	-
COMPLETE	•	•	•	•	•	•	•	•
COMPOF	•	-	•	-	•	•	-	-
COMPON	•	-	•	-	•	•	-	-
COMPSURF	-	-	○	-	○	○	-	-
CONTDCON	•	•	•	•	•	•	•	•
CONTPRON	•	•	•	•	•	•	•	•
CORROF	•	•	•	•	•	•	•	•
CORRTRAFO	-	-	-	-	-	-	-	-
COS	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
COUPDEF	-	○	-	○	-	-	○	○
COUPDEL	-	○	-	○	-	-	○	○
COUPOF	-	○	-	○	-	-	○	○
COUPOFS	-	○	-	○	-	-	○	○
COUPON	-	○	-	○	-	-	○	○
COUPONC	-	○	-	○	-	-	○	○
COUPRES	-	○	-	○	-	-	○	○
CP	●	●	●	●	●	●	●	●
CPBC	○	○	○	○	○	○	○	○
CPDEF	○	○	○	○	○	○	○	○
CPDEL	○	○	○	○	○	○	○	○
CPFMOF	○	○	○	○	○	○	○	○
CPFMON	○	○	○	○	○	○	○	○
CPFMSON	○	○	○	○	○	○	○	○
CPFPOS	○	○	○	○	○	○	○	○
CPFRS	○	○	○	○	○	○	○	○
CPLA	○	○	○	○	○	○	○	○
CPLCTID	○	○	○	○	○	○	○	○
CPLDEF	○	○	○	○	○	○	○	○
CPLDEL	○	○	○	○	○	○	○	○
CPLDEN	○	○	○	○	○	○	○	○
CPLINSC	○	○	○	○	○	○	○	○
CPLINTR	○	○	○	○	○	○	○	○
CPLNUM	○	○	○	○	○	○	○	○
CPLOF	○	○	○	○	○	○	○	○
CPLON	○	○	○	○	○	○	○	○
CPLOUTSC	○	○	○	○	○	○	○	○
CPLOUTTR	○	○	○	○	○	○	○	○
CPLPOS	○	○	○	○	○	○	○	○
CPLSETVAL	○	○	○	○	○	○	○	○
CPMALARM	○	○	○	○	○	○	○	○
CPMBRAKE	○	○	○	○	○	○	○	○
CPMPRT	○	○	○	○	○	○	○	○
CPMRESET	○	○	○	○	○	○	○	○
CPMSTART	○	○	○	○	○	○	○	○
CPMVDI	○	○	○	○	○	○	○	○
CPOF	○	○	○	○	○	○	○	○
CPON	○	○	○	○	○	○	○	○
CPRECOF	●	●	●	●	●	●	●	●

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
CPRECON	•	•	•	•	•	•	•	•
CPRES	○	○	○	○	○	○	○	○
CPROT	•	•	•	•	•	•	•	•
CPROTDEF	•	•	•	•	•	•	•	•
CPSETTYPE	○	○	○	○	○	○	○	○
CPSYNCOV	○	○	○	○	○	○	○	○
CPSYNCOV2	○	○	○	○	○	○	○	○
CPSYNCOV	○	○	○	○	○	○	○	○
CPSYNFIP	○	○	○	○	○	○	○	○
CPSYNFIP2	○	○	○	○	○	○	○	○
CPSYNFIV	○	○	○	○	○	○	○	○
CR	•	•	•	•	•	•	•	•
CROT	•	•	•	•	•	•	•	•
CROTS	•	•	•	•	•	•	•	•
CRPL	•	•	•	•	•	•	•	•
CSCALE	•	•	•	•	•	•	•	•
CSPLINE	○	○	○	○	○	○	○	○
CT	•	•	•	•	•	•	•	•
CTAB	-	-	-	-	-	-	-	-
CTABDEF	-	-	-	-	-	-	-	-
CTABDEL	-	-	-	-	-	-	-	-
CTABEND	-	-	-	-	-	-	-	-
CTABEXISTS	-	-	-	-	-	-	-	-
CTABFNO	-	-	-	-	-	-	-	-
CTABFPOL	-	-	-	-	-	-	-	-
CTABFSEG	-	-	-	-	-	-	-	-
CTABID	-	-	-	-	-	-	-	-
CTABINV	-	-	-	-	-	-	-	-
CTABISLOCK	-	-	-	-	-	-	-	-
CTABLOCK	-	-	-	-	-	-	-	-
CTABMEMTYP	-	-	-	-	-	-	-	-
CTABMPOL	-	-	-	-	-	-	-	-
CTABMSEG	-	-	-	-	-	-	-	-
CTABNO	-	-	-	-	-	-	-	-
CTABNOMEM	-	-	-	-	-	-	-	-
CTABPERIOD	-	-	-	-	-	-	-	-
CTABPOL	-	-	-	-	-	-	-	-
CTABPOLID	-	-	-	-	-	-	-	-
CTABSEG	-	-	-	-	-	-	-	-

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
CTABSEGID	-	-	-	-	-	-	-	-
CTABSEV	-	-	-	-	-	-	-	-
CTABSSV	-	-	-	-	-	-	-	-
CTABTEP	-	-	-	-	-	-	-	-
CTABTEV	-	-	-	-	-	-	-	-
CTABTMAX	-	-	-	-	-	-	-	-
CTABTMIN	-	-	-	-	-	-	-	-
CTABTSP	-	-	-	-	-	-	-	-
CTABTSV	-	-	-	-	-	-	-	-
CTABUNLOCK	-	-	-	-	-	-	-	-
CTOL	•	•	•	•	•	•	•	•
CTRANS	•	•	•	•	•	•	•	•
CUT2D	•	•	•	•	•	•	•	•
CUT2DD	•	•	•	•	•	•	•	•
CUT2DF	•	•	•	•	•	•	•	•
CUT2DFD	•	•	•	•	•	•	•	•
CUT3DC	-	-	-	-	-	-	-	-
CUT3DCC	-	-	-	-	-	-	-	-
CUT3DCCD	-	-	-	-	-	-	-	-
CUT3DCD	-	-	-	-	-	-	-	-
CUT3DF	-	-	-	-	-	-	-	-
CUT3DFD	-	-	-	-	-	-	-	-
CUT3DFF	-	-	-	-	-	-	-	-
CUT3DFS	-	-	-	-	-	-	-	-
CUTCONOF	•	•	•	•	•	•	•	•
CUTCONON	•	•	•	•	•	•	•	•
CUTMOD	•	•	•	•	•	•	•	•
CUTMODK	-	-	-	-	-	-	-	-
CYCLE60	○	○	•	•	•	•	•	•
CYCLE61	•	•	•	•	•	•	•	•
CYCLE62	•	•	•	•	•	•	•	•
CYCLE63	○	○	•	•	•	•	•	•
CYCLE64	○	○	•	•	•	•	•	•
CYCLE70	○	○	•	•	•	•	•	•
CYCLE72	•	•	•	•	•	•	•	•
CYCLE76	•	•	•	•	•	•	•	•
CYCLE77	•	•	•	•	•	•	•	•
CYCLE78	○	○	•	•	•	•	•	•
CYCLE79	○	○	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
CYCLE81	•	•	•	•	•	•	•	•
CYCLE82	•	•	•	•	•	•	•	•
CYCLE83	•	•	•	•	•	•	•	•
CYCLE84	•	•	•	•	•	•	•	•
CYCLE85	•	•	•	•	•	•	•	•
CYCLE86	•	•	•	•	•	•	•	•
CYCLE92	•	•	•	•	•	•	•	•
CYCLE95	•	•	•	•	•	•	•	•
CYCLE98	•	•	•	•	•	•	•	•
CYCLE99	•	•	•	•	•	•	•	•
CYCLE150	○	○	○	○	○	○	○	○
CYCLE435	-	-	-	-	-	-	-	-
CYCLE495	-	-	-	-	-	-	-	-
CYCLE750	•	•	•	•	•	•	•	•
CYCLE751	•	•	•	•	•	•	•	•
CYCLE752	•	•	•	•	•	•	•	•
CYCLE753	•	•	•	•	•	•	•	•
CYCLE754	•	•	•	•	•	•	•	•
CYCLE755	•	•	•	•	•	•	•	•
CYCLE756	•	•	•	•	•	•	•	•
CYCLE757	•	•	•	•	•	•	•	•
CYCLE758	•	•	•	•	•	•	•	•
CYCLE759	•	•	•	•	•	•	•	•
CYCLE800	-	-	•	•	•	•	•	•
CYCLE801	•	•	•	•	•	•	•	•
CYCLE802	•	•	•	•	•	•	•	•
CYCLE830	○	○	•	•	•	•	•	•
CYCLE832	•	•	•	•	•	•	•	•
CYCLE840	•	•	•	•	•	•	•	•
CYCLE899	○	○	•	•	•	•	•	•
CYCLE930	•	•	•	•	•	•	•	•
CYCLE940	•	•	•	•	•	•	•	•
CYCLE951	•	•	•	•	•	•	•	•
CYCLE952	○	○	•	•	•	•	•	•
CYCLE961	○	○	○	○	○	○	○	○
CYCLE971	○	○	○	○	○	○	○	○
CYCLE973	○	○	○	○	○	○	○	○
CYCLE974	○	○	○	○	○	○	○	○
CYCLE976	○	○	○	○	○	○	○	○

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
CYCLE977	○	○	○	○	○	○	○	○
CYCLE978	○	○	○	○	○	○	○	○
CYCLE979	○	○	○	○	○	○	○	○
CYCLE982	○	○	○	○	○	○	○	○
CYCLE994	○	○	○	○	○	○	○	○
CYCLE995	○	○	○	○	○	○	○	○
CYCLE996	○	-	○	-	○	○	-	-
CYCLE997	○	○	○	○	○	○	○	○
CYCLE998	○	○	○	○	○	○	○	○
CYCLE4071	-	-	-	-	-	-	-	-
CYCLE4072	-	-	-	-	-	-	-	-
CYCLE4073	-	-	-	-	-	-	-	-
CYCLE4074	-	-	-	-	-	-	-	-
CYCLE4075	-	-	-	-	-	-	-	-
CYCLE4077	-	-	-	-	-	-	-	-
CYCLE4078	-	-	-	-	-	-	-	-
CYCLE4079	-	-	-	-	-	-	-	-

Anvisningar D ... F

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
D	•	•	•	•	•	•	•	•
D0	•	•	•	•	•	•	•	•
DAC	•	•	•	•	•	•	•	•
DC	•	•	•	•	•	•	•	•
DCI	•	•	•	•	•	•	•	•
DCM	•	•	•	•	•	•	•	•
DCU	•	•	•	•	•	•	•	•
DEF	•	•	•	•	•	•	•	•
DEFINE	•	•	•	•	•	•	•	•
DEFAULT	•	•	•	•	•	•	•	•
DELAYFSTON	•	•	•	•	•	•	•	•
DELAYFSTOF	•	•	•	•	•	•	•	•
DELDL	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
DELDTG	•	•	•	•	•	•	•	•
DELETE	•	•	•	•	•	•	•	•
DELMOWNER	•	•	•	•	•	•	•	•
DEMLRES	•	•	•	•	•	•	•	•
DELMT	•	•	•	•	•	•	•	•
DELOBJ	-	-	-	-	-	-	-	-
DELT	•	•	•	•	•	•	•	•
DELTC	•	•	•	•	•	•	•	•
DELTOOLNV	•	•	•	•	•	•	•	•
DIACYCOFA	•	•	•	•	•	•	•	•
DIAM90	•	•	•	•	•	•	•	•
DIAM90A	•	•	•	•	•	•	•	•
DIAMCHAN	•	•	•	•	•	•	•	•
DIAMCHANA	•	•	•	•	•	•	•	•
DIAMCYCOF	•	•	•	•	•	•	•	•
DIAMOF	•	•	•	•	•	•	•	•
DIAMOFA	•	•	•	•	•	•	•	•
DIAMON	•	•	•	•	•	•	•	•
DIAMONA	•	•	•	•	•	•	•	•
DIC	•	•	•	•	•	•	•	•
DILF	•	•	•	•	•	•	•	•
DISABLE	•	•	•	•	•	•	•	•
DISC	•	•	•	•	•	•	•	•
DISCL	•	•	•	•	•	•	•	•
DISPLOF	•	•	•	•	•	•	•	•
DISPLON	•	•	•	•	•	•	•	•
DISPR	•	•	•	•	•	•	•	•
DISR	•	•	•	•	•	•	•	•
DISRP	•	•	•	•	•	•	•	•
DITE	•	•	•	•	•	•	•	•
DITS	•	•	•	•	•	•	•	•
DIV	•	•	•	•	•	•	•	•
DL	-	-	-	-	-	-	-	-
DO	•	•	•	•	•	•	•	•
DRFOF	•	•	•	•	•	•	•	•
DRIVE	•	•	•	•	•	•	•	•
DRIVEA	•	•	•	•	•	•	•	•
DYNFINISH	•	•	•	•	•	•	•	•
DYNNORM	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
DYNPOS	•	•	•	•	•	•	•	•
DYNROUGH	•	•	•	•	•	•	•	•
DYNSEMIFIN	•	•	•	•	•	•	•	•
DZERO	•	•	•	•	•	•	•	•
EAUTO	○	○	○	○	○	○	○	○
EGDEF	-	○	-	○	-	-	○	○
EGDEL	-	○	-	○	-	-	○	○
EGOFC	-	○	-	○	-	-	○	○
EGOFS	-	○	-	○	-	-	○	○
EGON	-	○	-	○	-	-	○	○
EGONSYN	-	○	-	○	-	-	○	○
EGONSYNE	-	○	-	○	-	-	○	○
ELSE	•	•	•	•	•	•	•	•
ENABLE	•	•	•	•	•	•	•	•
ENAT	○	○	○	○	○	○	○	○
ENDFOR	•	•	•	•	•	•	•	•
ENDIF	•	•	•	•	•	•	•	•
ENDLABEL	•	•	•	•	•	•	•	•
ENDLOOP	•	•	•	•	•	•	•	•
ENDPROC	•	•	•	•	•	•	•	•
ENDWHILE	•	•	•	•	•	•	•	•
ESRR	○	○	○	○	○	○	○	○
ESRS	○	○	○	○	○	○	○	○
ETAN	○	○	○	○	○	○	○	○
EVERY	•	•	•	•	•	•	•	•
EX	•	•	•	•	•	•	•	•
EXECSTRING	•	•	•	•	•	•	•	•
EXECTAB	•	•	•	•	•	•	•	•
EXECUTE	•	•	•	•	•	•	•	•
EXP	•	•	•	•	•	•	•	•
EXTCALL	•	•	•	•	•	•	•	•
EXTCLOSE	•	•	•	•	•	•	•	•
EXTERN	•	•	•	•	•	•	•	•
EXTOPEN	•	•	•	•	•	•	•	•
F	•	•	•	•	•	•	•	•
FA	•	•	•	•	•	•	•	•
FAD	•	•	•	•	•	•	•	•
FALSE	•	•	•	•	•	•	•	•
FB	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
FCTDEF	•	•	•	•	•	•	•	•
FCUB	•	•	•	•	•	•	•	•
FD	•	•	•	•	•	•	•	•
FDA	•	•	•	•	•	•	•	•
FENDNORM	•	•	•	•	•	•	•	•
FFWOF	•	•	•	•	•	•	•	•
FFWON	•	•	•	•	•	•	•	•
FGREF	•	•	•	•	•	•	•	•
FGROUP	•	•	•	•	•	•	•	•
FI	•	•	•	•	•	•	•	•
FIFOCTRL	•	•	•	•	•	•	•	•
FILEDATE	•	•	•	•	•	•	•	•
FILEINFO	•	•	•	•	•	•	•	•
FILESIZE	•	•	•	•	•	•	•	•
FILESTAT	•	•	•	•	•	•	•	•
FILETIME	•	•	•	•	•	•	•	•
FINEA	•	•	•	•	•	•	•	•
FL	•	•	•	•	•	•	•	•
FLIN	•	•	•	•	•	•	•	•
FMA	•	•	•	•	•	•	•	•
FNORM	•	•	•	•	•	•	•	•
FOCOF	○	○	○	○	○	○	○	○
FOCON	○	○	○	○	○	○	○	○
FOR	•	•	•	•	•	•	•	•
FP	•	•	•	•	•	•	•	•
FPO	-	-	-	-	-	-	-	-
FPR	•	•	•	•	•	•	•	•
FPRAOF	•	•	•	•	•	•	•	•
FPRAON	•	•	•	•	•	•	•	•
FRAME	•	•	•	•	•	•	•	•
FRC	•	•	•	•	•	•	•	•
FRCM	•	•	•	•	•	•	•	•
FROM	•	•	•	•	•	•	•	•
FTOC	•	•	•	•	•	•	•	•
FTOCOF	•	•	•	•	•	•	•	•
FTOCON	•	•	•	•	•	•	•	•
FXS	•	•	•	•	•	•	•	•
FXST	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
FXSW	•	•	•	•	•	•	•	•
FZ	•	•	•	•	•	•	•	•

Anvisningar G ... L

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
G0	•	•	•	•	•	•	•	•
G1	•	•	•	•	•	•	•	•
G2	•	•	•	•	•	•	•	•
G3	•	•	•	•	•	•	•	•
G4	•	•	•	•	•	•	•	•
G5	•	•	•	•	•	•	•	•
G7	•	•	•	•	•	•	•	•
G9	•	•	•	•	•	•	•	•
G17	•	•	•	•	•	•	•	•
G18	•	•	•	•	•	•	•	•
G19	•	•	•	•	•	•	•	•
G25	•	•	•	•	•	•	•	•
G26	•	•	•	•	•	•	•	•
G33	•	•	•	•	•	•	•	•
G34	•	•	•	•	•	•	•	•
G35	•	•	•	•	•	•	•	•
G40	•	•	•	•	•	•	•	•
G41	•	•	•	•	•	•	•	•
G42	•	•	•	•	•	•	•	•
G53	•	•	•	•	•	•	•	•
G54	•	•	•	•	•	•	•	•
G55	•	•	•	•	•	•	•	•
G56	•	•	•	•	•	•	•	•
G57	•	•	•	•	•	•	•	•
G58	→ G505							
G59	→ G506							
G60	•	•	•	•	•	•	•	•
G62	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
G63	•	•	•	•	•	•	•	•
G64	•	•	•	•	•	•	•	•
G70	•	•	•	•	•	•	•	•
G71	•	•	•	•	•	•	•	•
G74	•	•	•	•	•	•	•	•
G75	•	•	•	•	•	•	•	•
G90	•	•	•	•	•	•	•	•
G91	•	•	•	•	•	•	•	•
G93	•	•	•	•	•	•	•	•
G94	•	•	•	•	•	•	•	•
G95	•	•	•	•	•	•	•	•
G96	•	•	•	•	•	•	•	•
G97	•	•	•	•	•	•	•	•
G110	•	•	•	•	•	•	•	•
G111	•	•	•	•	•	•	•	•
G112	•	•	•	•	•	•	•	•
G140	•	•	•	•	•	•	•	•
G141	•	•	•	•	•	•	•	•
G142	•	•	•	•	•	•	•	•
G143	•	•	•	•	•	•	•	•
G147	•	•	•	•	•	•	•	•
G148	•	•	•	•	•	•	•	•
G153	•	•	•	•	•	•	•	•
G247	•	•	•	•	•	•	•	•
G248	•	•	•	•	•	•	•	•
G290	•	•	•	•	•	•	•	•
G291	•	•	•	•	•	•	•	•
G331	•	•	•	•	•	•	•	•
G332	•	•	•	•	•	•	•	•
G335	•	•	•	•	•	•	•	•
G336	•	•	•	•	•	•	•	•
G340	•	•	•	•	•	•	•	•
G341	•	•	•	•	•	•	•	•
G347	•	•	•	•	•	•	•	•
G348	•	•	•	•	•	•	•	•
G450	•	•	•	•	•	•	•	•
G451	•	•	•	•	•	•	•	•
G460	•	•	•	•	•	•	•	•
G461	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
G462	•	•	•	•	•	•	•	•
G500	•	•	•	•	•	•	•	•
G505 ... G599	•	•	•	•	•	•	•	•
G601	•	•	•	•	•	•	•	•
G602	•	•	•	•	•	•	•	•
G603	•	•	•	•	•	•	•	•
G621	•	•	•	•	•	•	•	•
G641	•	•	•	•	•	•	•	•
G642	•	•	•	•	•	•	•	•
G643	•	•	•	•	•	•	•	•
G644	•	•	•	•	•	•	•	•
G645	•	•	•	•	•	•	•	•
G700	•	•	•	•	•	•	•	•
G710	•	•	•	•	•	•	•	•
G810 ... G819	-	-	-	-	-	-	-	-
G820 ... G829	-	-	-	-	-	-	-	-
G931	•	•	•	•	•	•	•	•
G942	•	•	•	•	•	•	•	•
G952	•	•	•	•	•	•	•	•
G961	•	•	•	•	•	•	•	•
G962	•	•	•	•	•	•	•	•
G971	•	•	•	•	•	•	•	•
G972	•	•	•	•	•	•	•	•
G973	•	•	•	•	•	•	•	•
GEOAX	•	•	•	•	•	•	•	•
GET	•	•	•	•	•	•	•	•
GETACTT	•	•	•	•	•	•	•	•
GETACTTD	•	•	•	•	•	•	•	•
GETD	-	-	-	-	-	○	-	•
GETDNO	•	•	•	•	•	•	•	•
GETEXET	•	•	•	•	•	•	•	•
GETFREELOC	•	•	•	•	•	•	•	•
GETSELT	•	•	•	•	•	•	•	•
GETT	•	•	•	•	•	•	•	•
GETTCOR	•	•	•	•	•	•	•	•
GETTENV	•	•	•	•	•	•	•	•
GETVARAP	•	•	•	•	•	•	•	•
GETVARDFT	•	•	•	•	•	•	•	•
GETVARLIM	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
GETVARPHU	•	•	•	•	•	•	•	•
GETVARTYP	•	•	•	•	•	•	•	•
GFRAME0 ... GFRAME100	-	-	-	-	-	-	-	-
GOTO	•	•	•	•	•	•	•	•
GOTOB	•	•	•	•	•	•	•	•
GOTOC	•	•	•	•	•	•	•	•
GOTOF	•	•	•	•	•	•	•	•
GOTOS	•	•	•	•	•	•	•	•
GP	•	•	•	•	•	•	•	•
GWPSOF	•	•	•	•	•	•	•	•
GROUP_ ADDEND	•	•	•	•	•	•	•	•
GROUP_BEGIN	•	•	•	•	•	•	•	•
GROUP_END	•	•	•	•	•	•	•	•
GWPSON	•	•	•	•	•	•	•	•
H...	•	•	•	•	•	•	•	•
HOLES1	•	•	•	•	•	•	•	•
HOLES2	•	•	•	•	•	•	•	•
I	•	•	•	•	•	•	•	•
I1	•	•	•	•	•	•	•	•
IC	•	•	•	•	•	•	•	•
ICYCOF	•	•	•	•	•	•	•	•
ICYCON	•	•	•	•	•	•	•	•
ID	•	•	•	•	•	•	•	•
IDS	•	•	•	•	•	•	•	•
IF	•	•	•	•	•	•	•	•
INDEX	•	•	•	•	•	•	•	•
INIPO	•	•	•	•	•	•	•	•
INIRE	•	•	•	•	•	•	•	•
INICF	•	•	•	•	•	•	•	•
INIT	-	-	-	-	-	○	-	•
INITIAL								
INT	•	•	•	•	•	•	•	•
INTERSEC	•	•	•	•	•	•	•	•
INVCCW	-	-	-	-	-	-	-	-
INVCW	-	-	-	-	-	-	-	-
INVFRAME	•	•	•	•	•	•	•	•
IP	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
IPOBRKA	•	•	•	•	•	•	•	•
IPOENDA	•	•	•	•	•	•	•	•
IPTRLOCK	•	•	•	•	•	•	•	•
IPTRUNLOCK	•	•	•	•	•	•	•	•
IR	•	•	•	•	•	•	•	•
ISAXIS	•	•	•	•	•	•	•	•
ISD	-	-	-	-	-	-	-	-
ISFILE	•	•	•	•	•	•	•	•
ISNUMBER	•	•	•	•	•	•	•	•
ISOCALL	•	•	•	•	•	•	•	•
ISVAR	•	•	•	•	•	•	•	•
J	•	•	•	•	•	•	•	•
J1	•	•	•	•	•	•	•	•
JERKA	•	•	•	•	•	•	•	•
JERKLIM	•	•	•	•	•	•	•	•
JERKLIMA	•	•	•	•	•	•	•	•
JR	•	•	•	•	•	•	•	•
K	•	•	•	•	•	•	•	•
K1	•	•	•	•	•	•	•	•
KONT	•	•	•	•	•	•	•	•
KONTC	•	•	•	•	•	•	•	•
KONTT	•	•	•	•	•	•	•	•
KR	•	•	•	•	•	•	•	•
L	•	•	•	•	•	•	•	•
LEAD								
Verktygsgs-orientering	-	-	-	-	-	-	-	-
Orienterings-polyn.	-	-	-	-	-	-	-	-
LEADOF	-	-	-	•	-	-	•	•
LEADON	-	-	-	•	-	-	•	•
LENTOAX	•	•	•	•	•	•	•	•
LFOF	•	•	•	•	•	•	•	•
LFON	•	•	•	•	•	•	•	•
LFPOS	•	•	•	•	•	•	•	•
LFTXT	•	•	•	•	•	•	•	•
LFWP	•	•	•	•	•	•	•	•
LIFTFAST	•	•	•	•	•	•	•	•
LIMS	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
LLI	•	•	•	•	•	•	•	•
LN	•	•	•	•	•	•	•	•
LOCK	•	•	•	•	•	•	•	•
LONGHOLE	•	•	•	•	•	•	•	•
LOOP	•	•	•	•	•	•	•	•

Anvisningar M ... R

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
M0	•	•	•	•	•	•	•	•
M1	•	•	•	•	•	•	•	•
M2	•	•	•	•	•	•	•	•
M3	•	•	•	•	•	•	•	•
M4	•	•	•	•	•	•	•	•
M5	•	•	•	•	•	•	•	•
M6	•	•	•	•	•	•	•	•
M17	•	•	•	•	•	•	•	•
M19	•	•	•	•	•	•	•	•
M30	•	•	•	•	•	•	•	•
M40	•	•	•	•	•	•	•	•
M41 ... M45	•	•	•	•	•	•	•	•
M70	•	•	•	•	•	•	•	•
MASLDEF	-	-	-	-	-	-	-	-
MASLDEL	-	-	-	-	-	-	-	-
MASLOF	-	-	-	-	-	-	-	-
MASLOFS	-	-	-	-	-	-	-	-
MASLON	-	-	-	-	-	-	-	-
MATCH	•	•	•	•	•	•	•	•
MAXVAL	•	•	•	•	•	•	•	•
MCALL	•	•	•	•	•	•	•	•
MEAC	-	-	-	-	-	-	-	-
MEAFRAME	•	•	•	•	•	•	•	•
MEAS	•	•	•	•	•	•	•	•
MEASA	-	-	-	-	-	-	-	-

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
MEASURE	•	•	•	•	•	•	•	•
MEAW	•	•	•	•	•	•	•	•
MEAWA	-	-	-	-	-	-	-	-
MI	•	•	•	•	•	•	•	•
MINDEX	•	•	•	•	•	•	•	•
MINVAL	•	•	•	•	•	•	•	•
MIRROR	•	•	•	•	•	•	•	•
MMC	•	•	•	•	•	•	•	•
MOD	•	•	•	•	•	•	•	•
MODAXVAL	•	•	•	•	•	•	•	•
MOV	•	•	•	•	•	•	•	•
MOVT	•	•	•	•	•	•	•	•
MSG	•	•	•	•	•	•	•	•
MVTOOL	•	•	•	•	•	•	•	•
N	•	•	•	•	•	•	•	•
NAMETOINT	•	•	•	•	•	•	•	•
NC	•	•	•	•	•	•	•	•
NEWCONF	•	•	•	•	•	•	•	•
NEWMT	•	•	•	•	•	•	•	•
NEWT	•	•	•	•	•	•	•	•
NORM	•	•	•	•	•	•	•	•
NOT	•	•	•	•	•	•	•	•
NPROT	•	•	•	•	•	•	•	•
NPROTDEF	•	•	•	•	•	•	•	•
NUMBER	•	•	•	•	•	•	•	•
OEMIPO1	-	-	-	-	-	-	-	-
OEMIPO2	-	-	-	-	-	-	-	-
OF	•	•	•	•	•	•	•	•
OFFN	•	•	•	•	•	•	•	•
OMA1	-	-	-	-	-	-	-	-
OMA2	-	-	-	-	-	-	-	-
OMA3	-	-	-	-	-	-	-	-
OMA4	-	-	-	-	-	-	-	-
OMA5	-	-	-	-	-	-	-	-
OR	•	•	•	•	•	•	•	•
ORIXES	-	-	-	-	-	-	-	-
ORIXPOS	-	-	-	-	-	-	-	-
ORIC	-	-	-	-	-	-	-	-
ORICONCCW	-	-	-	-	-	-	-	-

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
ORICONCW	-	-	-	-	-	-	-	-
ORICONIO	-	-	-	-	-	-	-	-
ORICONTO	-	-	-	-	-	-	-	-
ORICURVE	-	-	-	-	-	-	-	-
ORID	-	-	-	-	-	-	-	-
ORIEULER	-	-	-	-	-	-	-	-
ORIMKS	-	-	-	-	-	-	-	-
ORIPATH	-	-	-	-	-	-	-	-
ORIPATHS	-	-	-	-	-	-	-	-
ORIPANE	-	-	-	-	-	-	-	-
ORIRESET	-	-	-	-	-	-	-	-
ORIROTA	-	-	-	-	-	-	-	-
ORIROTC	-	-	-	-	-	-	-	-
ORIROTR	-	-	-	-	-	-	-	-
ORIROTT	-	-	-	-	-	-	-	-
ORIRPY	-	-	-	-	-	-	-	-
ORIRPY2	-	-	-	-	-	-	-	-
ORIS	-	-	-	-	-	-	-	-
ORISOF	-	-	-	-	-	-	-	-
ORISOLH	-	-	-	-	-	-	-	-
ORISON	-	-	-	-	-	-	-	-
ORIVECT	-	-	-	-	-	-	-	-
ORIVIRT1	-	-	-	-	-	-	-	-
ORIVIRT2	-	-	-	-	-	-	-	-
ORIWKS	-	-	-	-	-	-	-	-
OS	-	-	-	-	-	-	-	-
OSB	-	-	-	-	-	-	-	-
OSC	-	-	-	-	-	-	-	-
OSCILL	-	-	-	-	-	-	-	-
OSCTRL	-	-	-	-	-	-	-	-
OSD	-	-	-	-	-	-	-	-
OSE	-	-	-	-	-	-	-	-
OSNSC	•	•	•	•	•	•	•	•
OSOF	-	-	-	-	-	-	-	-
OSP1	•	•	•	•	•	•	•	•
OSP2	•	•	•	•	•	•	•	•
OSS	-	-	-	-	-	-	-	-
OSSE	-	-	-	-	-	-	-	-
OST	-	-	-	-	-	-	-	-

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
OST1	•	•	•	•	•	•	•	•
OST2	•	•	•	•	•	•	•	•
OTOL	•	•	•	•	•	•	•	•
OVR	•	•	•	•	•	•	•	•
OVRA	•	•	•	•	•	•	•	•
OVRRAP	•	•	•	•	•	•	•	•
P	•	•	•	•	•	•	•	•
PAROT	•	•	•	•	•	•	•	•
PAROTOF	•	•	•	•	•	•	•	•
PCALL	•	•	•	•	•	•	•	•
PDELAYOF	-	-	-	-	-	-	-	-
PDELAYON	-	-	-	-	-	-	-	-
PHI	-	-	-	-	-	-	-	-
PHU	•	•	•	•	•	•	•	•
PL	-	-	-	-	-	-	-	-
PM	•	•	•	•	•	•	•	•
PO	-	-	-	-	-	-	-	-
POCKET3	•	•	•	•	•	•	•	•
POCKET4	•	•	•	•	•	•	•	•
POLF	•	•	•	•	•	•	•	•
POLFA	•	•	•	•	•	•	•	•
POLFMASK	•	•	•	•	•	•	•	•
POLFMLIN	•	•	•	•	•	•	•	•
POLY	-	-	-	-	-	-	-	-
POLYPATH	-	-	-	-	-	-	-	-
PON	-	-	-	-	-	-	-	-
PONS	-	-	-	-	-	-	-	-
POS	•	•	•	•	•	•	•	•
POSA	•	•	•	•	•	•	•	•
POSM	•	•	•	•	•	•	•	•
POSMT	•	•	•	•	•	•	•	•
POSP	•	•	•	•	•	•	•	•
POSRANGE	•	•	•	•	•	•	•	•
POT	•	•	•	•	•	•	•	•
PR	•	•	•	•	•	•	•	•
PREPRO	•	•	•	•	•	•	•	•
PRESETON	•	•	•	•	•	•	•	•
PRESETONS	•	•	•	•	•	•	•	•
PRIO	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
PRLOC	•	•	•	•	•	•	•	•
PROC	•	•	•	•	•	•	•	•
PROTA	•	•	•	•	•	•	•	•
PROTD	•	•	•	•	•	•	•	•
PROTS	•	•	•	•	•	•	•	•
PSI	-	-	-	-	-	-	-	-
PTP	•	•	•	•	•	•	•	•
PTPG0	•	•	•	•	•	•	•	•
PTPWOC	•	•	•	•	•	•	•	•
PUNCHACC	-	-	-	-	-	-	-	-
PUTFTOC	•	•	•	•	•	•	•	•
PUTFTOCF	•	•	•	•	•	•	•	•
PW	○	○	○	○	○	○	○	○
QU	•	•	•	•	•	•	•	•
R...	•	•	•	•	•	•	•	•
RAC	•	•	•	•	•	•	•	•
RDISABLE	•	•	•	•	•	•	•	•
READ	•	•	•	•	•	•	•	•
REAL	•	•	•	•	•	•	•	•
RELEASE	•	•	•	•	•	•	•	•
REP	•	•	•	•	•	•	•	•
REPEAT	•	•	•	•	•	•	•	•
REPEATB	•	•	•	•	•	•	•	•
REPOSA	•	•	•	•	•	•	•	•
REPOSH	•	•	•	•	•	•	•	•
REPOSHA	•	•	•	•	•	•	•	•
REPOSL	•	•	•	•	•	•	•	•
REPOSQ	•	•	•	•	•	•	•	•
REPOSQA	•	•	•	•	•	•	•	•
RESETMON	•	•	•	•	•	•	•	•
RET	•	•	•	•	•	•	•	•
RETB	•	•	•	•	•	•	•	•
RIC	•	•	•	•	•	•	•	•
RINDEX	•	•	•	•	•	•	•	•
RMB	•	•	•	•	•	•	•	•
RME	•	•	•	•	•	•	•	•
RMI	•	•	•	•	•	•	•	•
RMN	•	•	•	•	•	•	•	•
RND	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
RNDM	•	•	•	•	•	•	•	•
ROT	•	•	•	•	•	•	•	•
ROTS	•	•	•	•	•	•	•	•
ROUND	•	•	•	•	•	•	•	•
ROUNDUP	•	•	•	•	•	•	•	•
RP	•	•	•	•	•	•	•	•
RPL	•	•	•	•	•	•	•	•
RT	•	•	•	•	•	•	•	•
RTLIOF	•	•	•	•	•	•	•	•
RTLION	•	•	•	•	•	•	•	•

Anvisningar S ... Z

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
S	•	•	•	•	•	•	•	•
SAVE	•	•	•	•	•	•	•	•
SBLOF	•	•	•	•	•	•	•	•
SBLON	•	•	•	•	•	•	•	•
SC	•	•	•	•	•	•	•	•
SCALE	•	•	•	•	•	•	•	•
SCC	•	•	•	•	•	•	•	•
SCPARA	•	•	•	•	•	•	•	•
SD	○	○	○	○	○	○	○	○
SET	•	•	•	•	•	•	•	•
SETAL	•	•	•	•	•	•	•	•
SETDNO	•	•	•	•	•	•	•	•
SETINT	•	•	•	•	•	•	•	•
SETM	-	-	-	-	-	○	-	•
SETMS	•	•	•	•	•	•	•	•
SETMS(n)	•	•	•	•	•	•	•	•
SETMTH	•	•	•	•	•	•	•	•
SETPIECE	•	•	•	•	•	•	•	•
SETTA	•	•	•	•	•	•	•	•
SETTCOR	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
SETTIA	•	•	•	•	•	•	•	•
SF	•	•	•	•	•	•	•	•
SIN	•	•	•	•	•	•	•	•
SIRELAY	-	-	-	-	-	-	-	-
SIRELIN	-	-	-	-	-	-	-	-
SIRELOUT	-	-	-	-	-	-	-	-
SIRELTIME	-	-	-	-	-	-	-	-
SLOT1	•	•	•	•	•	•	•	•
SLOT2	•	•	•	•	•	•	•	•
SOFT	•	•	•	•	•	•	•	•
SOFTA	•	•	•	•	•	•	•	•
SON	-	-	-	-	-	-	-	-
SONS	-	-	-	-	-	-	-	-
SPATH	•	•	•	•	•	•	•	•
SPCOF	•	•	•	•	•	•	•	•
SPCON	•	•	•	•	•	•	•	•
SPI	•	•	•	•	•	•	•	•
SPIF1	-	-	-	-	-	-	-	-
SPIF2	-	-	-	-	-	-	-	-
SPLINEPATH	○	○	○	○	○	○	○	○
SPN	-	-	-	-	-	-	-	-
SPOF	-	-	-	-	-	-	-	-
SPOS	•	•	•	•	•	•	•	•
SPOSA	•	•	•	•	•	•	•	•
SPP	-	-	-	-	-	-	-	-
SPRINT	•	•	•	•	•	•	•	•
SQRT	•	•	•	•	•	•	•	•
SR	•	•	•	•	•	•	•	•
SRA	•	•	•	•	•	•	•	•
ST	•	•	•	•	•	•	•	•
STA	•	•	•	•	•	•	•	•
START	-	-	-	-	-	○	-	•
STARTFIFO	•	•	•	•	•	•	•	•
STAT	•	•	•	•	•	•	•	•
STOLF	•	•	•	•	•	•	•	•
STOPFIFO	•	•	•	•	•	•	•	•
STOPRE	•	•	•	•	•	•	•	•
STOPREOF	•	•	•	•	•	•	•	•
STRING	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
STRINGFELD	•	•	•	•	•	•	•	•
STRINGIS	•	•	•	•	•	•	•	•
STRLEN	•	•	•	•	•	•	•	•
SUBSTR	•	•	•	•	•	•	•	•
SUPA	•	•	•	•	•	•	•	•
SVC	•	•	•	•	•	•	•	•
SYNFCT	•	•	•	•	•	•	•	•
SYNR	•	•	•	•	•	•	•	•
SYNRW	•	•	•	•	•	•	•	•
SYNW	•	•	•	•	•	•	•	•
T	•	•	•	•	•	•	•	•
TAN	•	•	•	•	•	•	•	•
TANG	-	-	-	-	-	-	-	-
TANGDEL	-	-	-	-	-	-	-	-
TANGOF	-	-	-	-	-	-	-	-
TANGON	-	-	-	-	-	-	-	-
TCA (828D: _TCA)	•	•	•	•	•	•	•	•
TCARR	•	-	•	-	•	•	-	-
TCI	•	•	•	•	•	•	•	•
TCOABS	•	-	•	-	•	•	-	-
TCOFR	•	-	•	-	•	•	-	-
TCOFRX	•	-	•	-	•	•	-	-
TCOFRY	•	-	•	-	•	•	-	-
TCOFRZ	•	-	•	-	•	•	-	-
THETA	-	-	-	-	-	-	-	-
TILT	-	-	-	-	-	-	-	-
TLIFT	-	-	-	-	-	-	-	-
TML	•	•	•	•	•	•	•	•
TMOF	•	•	•	•	•	•	•	•
TMON	•	•	•	•	•	•	•	•
TO	•	•	•	•	•	•	•	•
TOFF	•	•	•	•	•	•	•	•
TOFFL	•	•	•	•	•	•	•	•
TOFFOF	•	•	•	•	•	•	•	•
TOFFON	•	•	•	•	•	•	•	•
TOFFR	•	•	•	•	•	•	•	•
TOFRAME	•	•	•	•	•	•	•	•
TOFRAMEX	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fråsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fråsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fråsa Export (me821)	SW28x(1) CNC-SW Fråsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
TOFRAMEY	•	•	•	•	•	•	•	•
TOFRAMEZ	•	•	•	•	•	•	•	•
TOLOWER	•	•	•	•	•	•	•	•
TOOLENV	•	•	•	•	•	•	•	•
TOOLGNT	•	•	•	•	•	•	•	•
TOOLGT	•	•	•	•	•	•	•	•
TOROT	•	•	•	•	•	•	•	•
TOROTOF	•	•	•	•	•	•	•	•
TOROTX	•	•	•	•	•	•	•	•
TOROTY	•	•	•	•	•	•	•	•
TOROTZ	•	•	•	•	•	•	•	•
TOUPPER	•	•	•	•	•	•	•	•
TOWBCS	•	-	•	-	•	•	-	-
TOWKCS	•	-	•	-	•	•	-	-
TOWMCS	•	-	•	-	•	•	-	-
TOWSTD	•	-	•	-	•	•	-	-
TOWTCS	•	-	•	-	•	•	-	-
TOWWCS	•	-	•	-	•	•	-	-
TR	•	•	•	•	•	•	•	•
TRAANG	-	-	-	-	-	-	○	○
TRACON	-	-	-	-	-	-	○	○
TRACYL	○	○	○	○	○	○	○	○
TRAFOOF	•	•	•	•	•	•	•	•
TRAFOON	-	-	-	-	-	-	-	-
TRAILOF	•	•	•	•	•	•	•	•
TRAILON	•	•	•	•	•	•	•	•
TRANS	•	•	•	•	•	•	•	•
TRANSMIT	○	○	○	○	○	○	○	○
TRAORI	-	-	-	-	-	-	-	-
TRUE	•	•	•	•	•	•	•	•
TRUNC	•	•	•	•	•	•	•	•
TU	•	•	•	•	•	•	•	•
TURN	•	•	•	•	•	•	•	•
ULI	•	•	•	•	•	•	•	•
UNLOCK	•	•	•	•	•	•	•	•
UNTIL	•	•	•	•	•	•	•	•
UPATH	•	•	•	•	•	•	•	•
VAR	•	•	•	•	•	•	•	•
VELOLIM	•	•	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D							
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW Fräsa Export (me42)	SW24x(5) CNC-SW Svarva Export (te42)	SW26x(3) CNC-SW Fräsa Export (me62)	SW26x(3) CNC-SW Svarva Export (te62)	SW28x(2) CNC-SW Fräsa Export (me821)	SW28x(1) CNC-SW Fräsa Adv. Export (me822)	SW28x(2) CNC-SW Svarva Export (te821)	SW28x(1) CNC-SW Svarva Adv. Export (te822)
VEOLIMA	•	•	•	•	•	•	•	•
WAITC	•	•	•	•	•	•	•	•
WAITE	-	-	-	-	-	○	-	•
WAITENC	•	•	•	•	•	•	•	•
WAITM	-	-	-	-	-	○	-	•
WAITMC	-	-	-	-	-	○	-	•
WAITP	•	•	•	•	•	•	•	•
WAITS	•	•	•	•	•	•	•	•
WALCS0	•	•	•	•	•	•	•	•
WALCS1	•	•	•	•	•	•	•	•
WALCS2	•	•	•	•	•	•	•	•
WALCS3	•	•	•	•	•	•	•	•
WALCS4	•	•	•	•	•	•	•	•
WALCS5	•	•	•	•	•	•	•	•
WALCS6	•	•	•	•	•	•	•	•
WALCS7	•	•	•	•	•	•	•	•
WALCS8	•	•	•	•	•	•	•	•
WALCS9	•	•	•	•	•	•	•	•
WALCS10	•	•	•	•	•	•	•	•
WALIMOF	•	•	•	•	•	•	•	•
WALIMON	•	•	•	•	•	•	•	•
WHEN	•	•	•	•	•	•	•	•
WHENEVER	•	•	•	•	•	•	•	•
WHILE	•	•	•	•	•	•	•	•
WORKPIECE	•	•	•	•	•	•	•	•
WRITE	•	•	•	•	•	•	•	•
WRTPR	•	•	•	•	•	•	•	•
X	•	•	•	•	•	•	•	•
XOR	•	•	•	•	•	•	•	•
Y	•	•	•	•	•	•	•	•
Z	•	•	•	•	•	•	•	•

21.2.2 Styrningsvarianter slipa

Anvisningar A ... C

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
:	•	•	•	•	•	•
*	•	•	•	•	•	•
+	•	•	•	•	•	•
-	•	•	•	•	•	•
<	•	•	•	•	•	•
<<	•	•	•	•	•	•
<=	•	•	•	•	•	•
=	•	•	•	•	•	•
>=	•	•	•	•	•	•
/	•	•	•	•	•	•
/0 ... /7	•	•	•	•	•	•
A	•	•	•	•	•	•
A2	-	-	-	-	-	-
A3	-	-	-	-	-	-
A4	-	-	-	-	-	-
A5	-	-	-	-	-	-
A6	-	-	-	-	-	-
A7	-	-	-	-	-	-
ABS	•	•	•	•	•	•
AC	•	•	•	•	•	•
ACC	•	•	•	•	•	•
ACCLIMA	•	•	•	•	•	•
ACN	•	•	•	•	•	•
ACOS	•	•	•	•	•	•
ACP	•	•	•	•	•	•
ACTBLOCNO	•	•	•	•	•	•
ADDFRAME	•	•	•	•	•	•
ADIS	•	•	•	•	•	•
ADISPOS	•	•	•	•	•	•
ADISPOSA	•	•	•	•	•	•
ALF	•	•	•	•	•	•
AMIRROR	•	•	•	•	•	•
AND	•	•	•	•	•	•
ANG	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
AP	•	•	•	•	•	•
APR	•	•	•	•	•	•
APRB	•	•	•	•	•	•
APRP	•	•	•	•	•	•
APW	•	•	•	•	•	•
APWB	•	•	•	•	•	•
APWP	•	•	•	•	•	•
APX	•	•	•	•	•	•
AR	•	•	•	•	•	•
AROT	•	•	•	•	•	•
AROTS	•	•	•	•	•	•
AS	•	•	•	•	•	•
ASCALE	•	•	•	•	•	•
ASIN	•	•	•	•	•	•
ASPLINE	○	○	○	○	○	○
ATAN2	•	•	•	•	•	•
ATOL	•	•	•	•	•	•
ATRANS	•	•	•	•	•	•
AUXFUDEL	•	•	•	•	•	•
AUXFUDELG	•	•	•	•	•	•
AUXFUMSEQ	•	•	•	•	•	•
AUXFUSYNC	•	•	•	•	•	•
AX	•	•	•	•	•	•
AXCTSWE	-	-	-	-	-	-
AXCTSWEC	-	-	-	-	-	-
AXCTSWED	-	-	-	-	-	-
AXIS	•	•	•	•	•	•
AXNAME	•	•	•	•	•	•
AXSTRING	•	•	•	•	•	•
AXTOCHAN	•	•	•	•	•	•
AXTOSPI	•	•	•	•	•	•
B	•	•	•	•	•	•
B2	-	-	-	-	-	-
B3	-	-	-	-	-	-
B4	-	-	-	-	-	-
B5	-	-	-	-	-	-
B6	-	-	-	-	-	-
B7	-	-	-	-	-	-
B_AND	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
B_OR	•	•	•	•	•	•
B_NOT	•	•	•	•	•	•
B_XOR	•	•	•	•	•	•
BAUTO	○	○	○	○	○	○
BLOCK	•	•	•	•	•	•
BLSYNC	•	•	•	•	•	•
BNAT	○	○	○	○	○	○
BOOL	•	•	•	•	•	•
BOUND	•	•	•	•	•	•
BRISK	•	•	•	•	•	•
BRISKA	•	•	•	•	•	•
BSPLINE	○	○	○	○	○	○
BTAN	○	○	○	○	○	○
C	•	•	•	•	•	•
C2	-	-	Kanalaxel- namn	-	-	-
C3	-	-	-	-	-	-
C4	-	-	-	-	-	-
C5	-	-	-	-	-	-
C6	-	-	-	-	-	-
C7	-	-	-	-	-	-
CAC	•	•	•	•	•	•
CACN	•	•	•	•	•	•
CACP	•	•	•	•	•	•
CALCDAT	•	•	•	•	•	•
CALCPOSI	•	•	•	•	•	•
CALL	•	•	•	•	•	•
CALLPATH	•	•	•	•	•	•
CANCEL	•	•	•	•	•	•
CASE	•	•	•	•	•	•
CDC	•	•	•	•	•	•
CDOF	-	-	-	-	-	-
CDOF2	-	-	-	-	-	-
CDON	-	-	-	-	-	-
CFC	•	•	•	•	•	•
CFIN	•	•	•	•	•	•
CFINE	•	•	•	•	•	•
CFTCP	•	•	•	•	•	•
CHAN	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
CHANDATA	•	•	•	•	•	•
CHAR	•	•	•	•	•	•
CHF	•	•	•	•	•	•
CHKDM	•	•	•	•	•	•
CHKDNO	•	•	•	•	•	•
CHR	•	•	•	•	•	•
CIC	•	•	•	•	•	•
CIP	•	•	•	•	•	•
CLEARM	-	-	•	-	-	•
CLRINT	•	•	•	•	•	•
CMIRROR	•	•	•	•	•	•
COARSEA	•	•	•	•	•	•
COLLPAIR	-	-	-	-	-	-
COMPCAD	•	•	•	•	•	•
COMPCURV	•	•	•	•	•	•
COMPLETE	•	•	•	•	•	•
COMPOF	•	•	•	•	•	•
COMPON	•	•	•	•	•	•
COMPSURF	-	-	-	-	-	-
CONTDCON	•	•	•	•	•	•
CONTPRON	•	•	•	•	•	•
CORROF	•	•	•	•	•	•
CORRTRAFO	-	-	-	-	-	-
COS	•	•	•	•	•	•
COUPDEF	○	○	○	○	○	○
COUPDEL	○	○	○	○	○	○
COUPOF	○	○	○	○	○	○
COUPOFS	○	○	○	○	○	○
COUPON	○	○	○	○	○	○
COUPONC	○	○	○	○	○	○
COUPRES	○	○	○	○	○	○
CP	•	•	•	•	•	•
CPBC	○	○	○	○	○	○
CPDEF	○	○	○	○	○	○
CPDEL	○	○	○	○	○	○
CPFMOF	○	○	○	○	○	○
CPFMON	○	○	○	○	○	○
CPFMSON	○	○	○	○	○	○
CPFPOS	○	○	○	○	○	○

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
CPFRS	○	○	○	○	○	○
CPLA	○	○	○	○	○	○
CPLCTID	○	○	○	○	○	○
CPLDEF	○	○	○	○	○	○
CPLDEL	○	○	○	○	○	○
CPLDEN	○	○	○	○	○	○
CPLINSC	○	○	○	○	○	○
CPLINTR	○	○	○	○	○	○
CPLNUM	○	○	○	○	○	○
CPLOF	○	○	○	○	○	○
CPLON	○	○	○	○	○	○
CPLOUTSC	○	○	○	○	○	○
CPLOUTTR	○	○	○	○	○	○
CPLPOS	○	○	○	○	○	○
CPLSETVAL	○	○	○	○	○	○
CPMALARM	○	○	○	○	○	○
CPMBRAKE	○	○	○	○	○	○
CPMPRT	○	○	○	○	○	○
CPMRESET	○	○	○	○	○	○
CPMSTART	○	○	○	○	○	○
CPMVDI	○	○	○	○	○	○
CPOF	○	○	○	○	○	○
CPON	○	○	○	○	○	○
CPRECOF	•	•	•	•	•	•
CPRECON	•	•	•	•	•	•
CPRES	○	○	○	○	○	○
CPROT	•	•	•	•	•	•
CPROTDEF	•	•	•	•	•	•
CPSETTYPE	○	○	○	○	○	○
CPSYNCOF	○	○	○	○	○	○
CPSYNCOF2	○	○	○	○	○	○
CPSYNCOV	○	○	○	○	○	○
CPSYNFIP	○	○	○	○	○	○
CPSYNFIP2	○	○	○	○	○	○
CPSYNFIV	○	○	○	○	○	○
CR	•	•	•	•	•	•
CROT	•	•	•	•	•	•
CROTS	•	•	•	•	•	•
CRPL	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
CSCALE	•	•	•	•	•	•
CSPLINE	○	○	○	○	○	○
CT	•	•	•	•	•	•
CTAB	-	-	-	-	-	-
CTABDEF	-	-	-	-	-	-
CTABDEL	-	-	-	-	-	-
CTABEND	-	-	-	-	-	-
CTABEXISTS	-	-	-	-	-	-
CTABFNO	-	-	-	-	-	-
CTABFPOL	-	-	-	-	-	-
CTABFSEG	-	-	-	-	-	-
CTABID	-	-	-	-	-	-
CTABINV	-	-	-	-	-	-
CTABISLOCK	-	-	-	-	-	-
CTABLOCK	-	-	-	-	-	-
CTABMEMTYP	-	-	-	-	-	-
CTABMPOL	-	-	-	-	-	-
CTABMSEG	-	-	-	-	-	-
CTABNO	-	-	-	-	-	-
CTABNOMEM	-	-	-	-	-	-
CTABPERIOD	-	-	-	-	-	-
CTABPOL	-	-	-	-	-	-
CTABPOLID	-	-	-	-	-	-
CTABSEG	-	-	-	-	-	-
CTABSEGID	-	-	-	-	-	-
CTABSEV	-	-	-	-	-	-
CTABSSV	-	-	-	-	-	-
CTABTEP	-	-	-	-	-	-
CTABTEV	-	-	-	-	-	-
CTABTMAX	-	-	-	-	-	-
CTABTMIN	-	-	-	-	-	-
CTABTSP	-	-	-	-	-	-
CTABTSV	-	-	-	-	-	-
CTABUNLOCK	-	-	-	-	-	-
CTOL	•	•	•	•	•	•
CTTRANS	•	•	•	•	•	•
CUT2D	•	•	•	•	•	•
CUT2DD	•	•	•	•	•	•
CUT2DF	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
CUT2DFD	•	•	•	•	•	•
CUT3DC	-	-	-	-	-	-
CUT3DCC	-	-	-	-	-	-
CUT3DCCD	-	-	-	-	-	-
CUT3DCD	-	-	-	-	-	-
CUT3DF	-	-	-	-	-	-
CUT3DFD	-	-	-	-	-	-
CUT3DFF	-	-	-	-	-	-
CUT3DFS	-	-	-	-	-	-
CUTCONOF	•	•	•	•	•	•
CUTCONON	•	•	•	•	•	•
CUTMOD	•	•	•	•	•	•
CUTMODK	-	-	-	-	-	-
CYCLE60	-	-	-	-	-	-
CYCLE61	-	-	-	-	-	-
CYCLE62	•	•	•	•	•	•
CYCLE63	-	-	-	-	-	-
CYCLE64	-	-	-	-	-	-
CYCLE70	-	-	-	-	-	-
CYCLE72	-	-	-	-	-	-
CYCLE76	-	-	-	-	-	-
CYCLE77	-	-	-	-	-	-
CYCLE78	-	-	-	-	-	-
CYCLE79	-	-	-	-	-	-
CYCLE81	-	-	-	-	-	-
CYCLE82	-	-	-	-	-	-
CYCLE83	-	-	-	-	-	-
CYCLE84	-	-	-	-	-	-
CYCLE85	-	-	-	-	-	-
CYCLE86	-	-	-	-	-	-
CYCLE92	-	-	-	-	-	-
CYCLE95	-	-	-	-	-	-
CYCLE98	-	-	-	-	-	-
CYCLE99	-	-	-	-	-	-
CYCLE150	-	-	-	-	-	-
CYCLE435	○	○	○	○	○	○
CYCLE495	○	○	○	○	○	○
CYCLE750	•	•	•	•	•	•
CYCLE751	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
• Standard						
○ Option						
- inte disponibel						
CYCLE752	•	•	•	•	•	•
CYCLE753	•	•	•	•	•	•
CYCLE754	•	•	•	•	•	•
CYCLE755	•	•	•	•	•	•
CYCLE756	•	•	•	•	•	•
CYCLE757	•	•	•	•	•	•
CYCLE758	•	•	•	•	•	•
CYCLE759	•	•	•	•	•	•
CYCLE800	○	○	○	○	○	○
CYCLE801	-	-	-	-	-	-
CYCLE802	-	-	-	-	-	-
CYCLE830	-	-	-	-	-	-
CYCLE832	•	•	•	•	•	•
CYCLE840	-	-	-	-	-	-
CYCLE899	-	-	-	-	-	-
CYCLE930	-	-	-	-	-	-
CYCLE940	-	-	-	-	-	-
CYCLE951	-	-	-	-	-	-
CYCLE952	-	-	-	-	-	-
CYCLE961	-	-	-	-	-	-
CYCLE971	-	-	-	-	-	-
CYCLE973	-	-	-	-	-	-
CYCLE974	-	-	-	-	-	-
CYCLE976	-	-	-	-	-	-
CYCLE977	-	-	-	-	-	-
CYCLE978	-	-	-	-	-	-
CYCLE979	-	-	-	-	-	-
CYCLE982	-	-	-	-	-	-
CYCLE994	-	-	-	-	-	-
CYCLE995	-	-	-	-	-	-
CYCLE996	-	-	-	-	-	-
CYCLE997	-	-	-	-	-	-
CYCLE998	-	-	-	-	-	-
CYCLE4071	•	•	•	•	•	•
CYCLE4072	•	•	•	•	•	•
CYCLE4073	•	•	•	•	•	•
CYCLE4074	•	•	•	•	•	•
CYCLE4075	•	•	•	•	•	•
CYCLE4077	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tek Export (gce42)	SW26x(3) CNC-SW G-tek Export (gce62)	SW28x(1) CNC-SW G-tek Adv. Export (gce82)	SW24x(5) CNC-SW G-tek Export (gse42)	SW26x(3) CNC-SW G-tek Export (gse62)	SW28x(1) CNC-SW G-tek Adv. Export (gse82)
CYCLE4078	•	•	•	•	•	•
CYCLE4079	•	•	•	•	•	•

Anvisningar D ... F

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tek Export (gce42)	SW26x(3) CNC-SW G-tek Export (gce62)	SW28x(1) CNC-SW G-tek Adv. Export (gce82)	SW24x(5) CNC-SW G-tek Export (gse42)	SW26x(3) CNC-SW G-tek Export (gse62)	SW28x(1) CNC-SW G-tek Adv. Export (gse82)
D	•	•	•	•	•	•
D0	•	•	•	•	•	•
DAC	•	•	•	•	•	•
DC	•	•	•	•	•	•
DCI	•	•	•	•	•	•
DCM	•	•	•	•	•	•
DCU	•	•	•	•	•	•
DEF	•	•	•	•	•	•
DEFINE	•	•	•	•	•	•
DEFAULT	•	•	•	•	•	•
DELAYFSTON	•	•	•	•	•	•
DELAYFSTOF	•	•	•	•	•	•
DELDL	•	•	•	•	•	•
DELDTG	•	•	•	•	•	•
DELETE	•	•	•	•	•	•
DELMLOWNER	•	•	•	•	•	•
DEMLRES	•	•	•	•	•	•
DELMT	-	-	-	-	-	-
DELOBJ	-	-	-	-	-	-
DELT	•	•	•	•	•	•
DELTC	•	•	•	•	•	•
DELTOOLNV	•	•	•	•	•	•
DIACYCOFA	•	•	•	•	•	•
DIAM90	•	•	•	•	•	•
DIAM90A	•	•	•	•	•	•
DIAMCHAN	•	•	•	•	•	•
DIAMCHANA	•	•	•	•	•	•
DIAMCYCOF	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
DIAMOF	•	•	•	•	•	•
DIAMOF A	•	•	•	•	•	•
DIAMON	•	•	•	•	•	•
DIAMON A	•	•	•	•	•	•
DIC	•	•	•	•	•	•
DILF	•	•	•	•	•	•
DISABLE	•	•	•	•	•	•
DISC	•	•	•	•	•	•
DISCL	•	•	•	•	•	•
DISPLOF	•	•	•	•	•	•
DISPLON	•	•	•	•	•	•
DISPR	•	•	•	•	•	•
DISR	•	•	•	•	•	•
DISRP	•	•	•	•	•	•
DITE	•	•	•	•	•	•
DITS	•	•	•	•	•	•
DIV	•	•	•	•	•	•
DL	-	-	-	-	-	-
DO	•	•	•	•	•	•
DRFOF	•	•	•	•	•	•
DRIVE	•	•	•	•	•	•
DRIVE A	•	•	•	•	•	•
DYNFINISH	•	•	•	•	•	•
DYNNORM	•	•	•	•	•	•
DYNPOS	•	•	•	•	•	•
DYNROUGH	•	•	•	•	•	•
DYNSEMIFIN	•	•	•	•	•	•
DZERO	•	•	•	•	•	•
EAUTO	○	○	○	○	○	○
EGDEF	○	○	○	○	○	○
EGDEL	○	○	○	○	○	○
EGOFC	○	○	○	○	○	○
EGOFS	○	○	○	○	○	○
EGON	○	○	○	○	○	○
EGONSYN	○	○	○	○	○	○
EGONSYNE	○	○	○	○	○	○
ELSE	•	•	•	•	•	•
ENABLE	•	•	•	•	•	•
ENAT	○	○	○	○	○	○

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
• Standard						
○ Option						
- inte disponibel						
ENDFOR	•	•	•	•	•	•
ENDIF	•	•	•	•	•	•
ENDLABEL	•	•	•	•	•	•
ENDLOOP	•	•	•	•	•	•
ENDPROC	•	•	•	•	•	•
ENDWHILE	•	•	•	•	•	•
ESRR	○	○	○	○	○	○
ESRS	○	○	○	○	○	○
ETAN	○	○	○	○	○	○
EVERY	•	•	•	•	•	•
EX	•	•	•	•	•	•
EXECSTRING	•	•	•	•	•	•
EXECTAB	•	•	•	•	•	•
EXECUTE	•	•	•	•	•	•
EXP	•	•	•	•	•	•
EXTCALL	•	•	•	•	•	•
EXTCLOSE	•	•	•	•	•	•
EXTERN	•	•	•	•	•	•
EXTOPEN	•	•	•	•	•	•
F	•	•	•	•	•	•
FA	•	•	•	•	•	•
FAD	•	•	•	•	•	•
FALSE	•	•	•	•	•	•
FB	•	•	•	•	•	•
FCTDEF	•	•	•	•	•	•
FCUB	•	•	•	•	•	•
FD	•	•	•	•	•	•
FDA	•	•	•	•	•	•
FENDNORM	•	•	•	•	•	•
FFWOF	•	•	•	•	•	•
FFWON	•	•	•	•	•	•
FGREF	•	•	•	•	•	•
FGROUP	•	•	•	•	•	•
FI	•	•	•	•	•	•
FIFOCTRL	•	•	•	•	•	•
FILEDATE	•	•	•	•	•	•
FILEINFO	•	•	•	•	•	•
FILESIZE	•	•	•	•	•	•
FILESTAT	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
FILETIME	•	•	•	•	•	•
FINEA	•	•	•	•	•	•
FL	•	•	•	•	•	•
FLIN	•	•	•	•	•	•
FMA	•	•	•	•	•	•
FNORM	•	•	•	•	•	•
FOCOF	○	○	○	○	○	○
FOCON	○	○	○	○	○	○
FOR	•	•	•	•	•	•
FP	•	•	•	•	•	•
FPO	-	-	-	-	-	-
FPR	•	•	•	•	•	•
FPRAOF	•	•	•	•	•	•
FPRAON	•	•	•	•	•	•
FRAME	•	•	•	•	•	•
FRC	•	•	•	•	•	•
FRCM	•	•	•	•	•	•
FROM	•	•	•	•	•	•
FTOC	•	•	•	•	•	•
FTOCOF	•	•	•	•	•	•
FTOCON	•	•	•	•	•	•
FXS	•	•	•	•	•	•
FXST	•	•	•	•	•	•
FXSW	•	•	•	•	•	•
FZ	•	•	•	•	•	•

Anvisningar G ... L

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
G0	•	•	•	•	•	•
G1	•	•	•	•	•	•
G2	•	•	•	•	•	•
G3	•	•	•	•	•	•
G4	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tek Export (gce42)	SW26x(3) CNC-SW G-tek Export (gce62)	SW28x(1) CNC-SW G-tek Adv. Export (gce82)	SW24x(5) CNC-SW G-tek Export (gse42)	SW26x(3) CNC-SW G-tek Export (gse62)	SW28x(1) CNC-SW G-tek Adv. Export (gse82)
G5	•	•	•	•	•	•
G7	•	•	•	•	•	•
G9	•	•	•	•	•	•
G17	•	•	•	•	•	•
G18	•	•	•	•	•	•
G19	•	•	•	•	•	•
G25	•	•	•	•	•	•
G26	•	•	•	•	•	•
G33	•	•	•	•	•	•
G34	•	•	•	•	•	•
G35	•	•	•	•	•	•
G40	•	•	•	•	•	•
G41	•	•	•	•	•	•
G42	•	•	•	•	•	•
G53	•	•	•	•	•	•
G54	•	•	•	•	•	•
G55	•	•	•	•	•	•
G56	•	•	•	•	•	•
G57	•	•	•	•	•	•
G58	→ G505					
G59	→ G506					
G60	•	•	•	•	•	•
G62	•	•	•	•	•	•
G63	•	•	•	•	•	•
G64	•	•	•	•	•	•
G70	•	•	•	•	•	•
G71	•	•	•	•	•	•
G74	•	•	•	•	•	•
G75	•	•	•	•	•	•
G90	•	•	•	•	•	•
G91	•	•	•	•	•	•
G93	•	•	•	•	•	•
G94	•	•	•	•	•	•
G95	•	•	•	•	•	•
G96	•	•	•	•	•	•
G97	•	•	•	•	•	•
G110	•	•	•	•	•	•
G111	•	•	•	•	•	•
G112	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
G140	•	•	•	•	•	•
G141	•	•	•	•	•	•
G142	•	•	•	•	•	•
G143	•	•	•	•	•	•
G147	•	•	•	•	•	•
G148	•	•	•	•	•	•
G153	•	•	•	•	•	•
G247	•	•	•	•	•	•
G248	•	•	•	•	•	•
G290	•	•	•	•	•	•
G291	-	-	-	-	-	-
G331	•	•	•	•	•	•
G332	•	•	•	•	•	•
G335	•	•	•	•	•	•
G336	•	•	•	•	•	•
G340	•	•	•	•	•	•
G341	•	•	•	•	•	•
G347	•	•	•	•	•	•
G348	•	•	•	•	•	•
G450	•	•	•	•	•	•
G451	•	•	•	•	•	•
G460	•	•	•	•	•	•
G461	•	•	•	•	•	•
G462	•	•	•	•	•	•
G500	•	•	•	•	•	•
G505 ... G599	•	•	•	•	•	•
G601	•	•	•	•	•	•
G602	•	•	•	•	•	•
G603	•	•	•	•	•	•
G621	•	•	•	•	•	•
G641	•	•	•	•	•	•
G642	•	•	•	•	•	•
G643	•	•	•	•	•	•
G644	•	•	•	•	•	•
G645	•	•	•	•	•	•
G700	•	•	•	•	•	•
G710	•	•	•	•	•	•
G810 ... G819	-	-	-	-	-	-
G820 ... G829	-	-	-	-	-	-

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
G931	•	•	•	•	•	•
G942	•	•	•	•	•	•
G952	•	•	•	•	•	•
G961	•	•	•	•	•	•
G962	•	•	•	•	•	•
G971	•	•	•	•	•	•
G972	•	•	•	•	•	•
G973	•	•	•	•	•	•
GEOAX	•	•	•	•	•	•
GET	•	•	•	•	•	•
GETACTT	•	•	•	•	•	•
GETACTTD	•	•	•	•	•	•
GETD	-	-	•	-	-	•
GETDNO	•	•	•	•	•	•
GETEXET	•	•	•	•	•	•
GETFREELOC	•	•	•	•	•	•
GETSELT	•	•	•	•	•	•
GETT	•	•	•	•	•	•
GETTCOR	•	•	•	•	•	•
GETTENV	•	•	•	•	•	•
GETVARAP	•	•	•	•	•	•
GETVARDFT	•	•	•	•	•	•
GETVARLIM	•	•	•	•	•	•
GETVARPHU	•	•	•	•	•	•
GETVARTYP	•	•	•	•	•	•
GFRAME0 ... GFRA- ME100	< 50	< 100	< 100	< 50	< 100	< 100
GOTO	•	•	•	•	•	•
GOTOB	•	•	•	•	•	•
GOTOC	•	•	•	•	•	•
GTOF	•	•	•	•	•	•
GOTOS	•	•	•	•	•	•
GP	•	•	•	•	•	•
GWPSOF	•	•	•	•	•	•
GROUP_ADDEND	•	•	•	•	•	•
GROUP_BEGIN	•	•	•	•	•	•
GROUP_END	•	•	•	•	•	•
GWPSON	•	•	•	•	•	•
H...	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tek Export (gce42)	SW26x(3) CNC-SW G-tek Export (gce62)	SW28x(1) CNC-SW G-tek Adv. Export (gce82)	SW24x(5) CNC-SW G-tek Export (gse42)	SW26x(3) CNC-SW G-tek Export (gse62)	SW28x(1) CNC-SW G-tek Adv. Export (gse82)
HOLES1	-	-	-	-	-	-
HOLES2	-	-	-	-	-	-
I	•	•	•	•	•	•
I1	•	•	•	•	•	•
IC	•	•	•	•	•	•
ICYCOF	•	•	•	•	•	•
ICYCON	•	•	•	•	•	•
ID	•	•	•	•	•	•
IDS	•	•	•	•	•	•
IF	•	•	•	•	•	•
INDEX	•	•	•	•	•	•
INIPO	•	•	•	•	•	•
INIRE	•	•	•	•	•	•
INICF	•	•	•	•	•	•
INIT	-	-	•	-	-	•
INITIAL						
INT	•	•	•	•	•	•
INTERSEC	•	•	•	•	•	•
INVCCW	-	-	-	-	-	-
INVCW	-	-	-	-	-	-
INVFRAME	•	•	•	•	•	•
IP	•	•	•	•	•	•
IPOBRKA	•	•	•	•	•	•
IPOENDA	•	•	•	•	•	•
IPTRLOCK	•	•	•	•	•	•
IPTRUNLOCK	•	•	•	•	•	•
IR	•	•	•	•	•	•
ISAXIS	•	•	•	•	•	•
ISD	-	-	-	-	-	-
ISFILE	•	•	•	•	•	•
ISNUMBER	•	•	•	•	•	•
ISOCALL	-	-	-	-	-	-
ISVAR	•	•	•	•	•	•
J	•	•	•	•	•	•
J1	•	•	•	•	•	•
JERKA	•	•	•	•	•	•
JERKLIM	•	•	•	•	•	•
JERKLIMA	•	•	•	•	•	•
JR	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
K	•	•	•	•	•	•
K1	•	•	•	•	•	•
KONT	•	•	•	•	•	•
KONTC	•	•	•	•	•	•
KONTT	•	•	•	•	•	•
KR	•	•	•	•	•	•
L	•	•	•	•	•	•
LEAD						
Verktygsorientering	-	-	-	-	-	-
Orienteringspolyn.	-	-	-	-	-	-
LEADOF	-	-	-	-	-	-
LEADON	-	-	-	-	-	-
LENTOAX	•	•	•	•	•	•
LFOF	•	•	•	•	•	•
LFON	•	•	•	•	•	•
LFPOS	•	•	•	•	•	•
LFTXT	•	•	•	•	•	•
LFWP	•	•	•	•	•	•
LIFTFAST	•	•	•	•	•	•
LIMS	•	•	•	•	•	•
LLI	•	•	•	•	•	•
LN	•	•	•	•	•	•
LOCK	•	•	•	•	•	•
LONGHOLE	-	-	-	-	-	-
LOOP	•	•	•	•	•	•

Anvisningar M ... R

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
M0	•	•	•	•	•	•
M1	•	•	•	•	•	•
M2	•	•	•	•	•	•
M3	•	•	•	•	•	•
M4	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
M5	•	•	•	•	•	•
M6	•	•	•	•	•	•
M17	•	•	•	•	•	•
M19	•	•	•	•	•	•
M30	•	•	•	•	•	•
M40	•	•	•	•	•	•
M41 ... M45	•	•	•	•	•	•
M70	•	•	•	•	•	•
MASLDEF	-	-	-	-	-	-
MASLDEL	-	-	-	-	-	-
MASLOF	-	-	-	-	-	-
MASLOFS	-	-	-	-	-	-
MASLON	-	-	-	-	-	-
MATCH	•	•	•	•	•	•
MAXVAL	•	•	•	•	•	•
MCALL	•	•	•	•	•	•
MEAC	-	○	○	-	○	○
MEAFRAME	•	•	•	•	•	•
MEAS	•	•	•	•	•	•
MEASA	-	○	○	-	○	○
MEASURE	•	•	•	•	•	•
MEAW	•	•	•	•	•	•
MEAWA	-	○	○	-	○	○
MI	•	•	•	•	•	•
MINDEX	•	•	•	•	•	•
MINVAL	•	•	•	•	•	•
MIRROR	•	•	•	•	•	•
MMC	•	•	•	•	•	•
MOD	•	•	•	•	•	•
MODAXVAL	•	•	•	•	•	•
MOV	•	•	•	•	•	•
MOVT	•	•	•	•	•	•
MSG	•	•	•	•	•	•
MVTOOL	•	•	•	•	•	•
N	•	•	•	•	•	•
NAMETOINT	•	•	•	•	•	•
NCK	•	•	•	•	•	•
NEWCONF	•	•	•	•	•	•
NEWMT	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
• Standard						
○ Option						
- inte disponibel						
NEWT	-	-	-	-	-	-
NORM	•	•	•	•	•	•
NOT	•	•	•	•	•	•
NPROT	•	•	•	•	•	•
NPROTDEF	•	•	•	•	•	•
NUMBER	•	•	•	•	•	•
OEMIPO1	-	-	-	-	-	-
OEMIPO2	-	-	-	-	-	-
OF	•	•	•	•	•	•
OFFN	•	•	•	•	•	•
OMA1	-	-	-	-	-	-
OMA2	-	-	-	-	-	-
OMA3	-	-	-	-	-	-
OMA4	-	-	-	-	-	-
OMA5	-	-	-	-	-	-
OR	•	•	•	•	•	•
ORIXES	-	-	-	-	-	-
ORIXPOS	-	-	-	-	-	-
ORIC	-	-	-	-	-	-
ORICONCCW	-	-	-	-	-	-
ORICONCW	-	-	-	-	-	-
ORICONIO	-	-	-	-	-	-
ORICONTO	-	-	-	-	-	-
ORICURVE	-	-	-	-	-	-
ORID	-	-	-	-	-	-
ORIEULER	-	-	-	-	-	-
ORIMKS	-	-	-	-	-	-
ORIPATH	-	-	-	-	-	-
ORIPATHS	-	-	-	-	-	-
ORIPANE	-	-	-	-	-	-
ORIRESET	-	-	-	-	-	-
ORIROTA	-	-	-	-	-	-
ORIROTC	-	-	-	-	-	-
ORIROTR	-	-	-	-	-	-
ORIROTT	-	-	-	-	-	-
ORIRPY	-	-	-	-	-	-
ORIRPY2	-	-	-	-	-	-
ORIS	-	-	-	-	-	-
ORISOF	-	-	-	-	-	-

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
ORISOLH	-	-	-	-	-	-
ORISON	-	-	-	-	-	-
ORIVECT	-	-	-	-	-	-
ORIVIRT1	-	-	-	-	-	-
ORIVIRT2	-	-	-	-	-	-
ORIWKS	-	-	-	-	-	-
OS	○	○	○	○	○	○
OSB	○	○	○	○	○	○
OSC	-	-	-	-	-	-
OSCILL	○	○	○	○	○	○
OSCTRL	○	○	○	○	○	○
OSD	-	-	-	-	-	-
OSE	○	○	○	○	○	○
OSNSC	●	●	●	●	●	●
OSOF	-	-	-	-	-	-
OSP1	●	●	●	●	●	●
OSP2	●	●	●	●	●	●
OSS	-	-	-	-	-	-
OSSE	-	-	-	-	-	-
OST	-	-	-	-	-	-
OST1	●	●	●	●	●	●
OST2	●	●	●	●	●	●
OTOL	●	●	●	●	●	●
OVR	●	●	●	●	●	●
OVRA	●	●	●	●	●	●
OVRRAP	●	●	●	●	●	●
P	●	●	●	●	●	●
PAROT	●	●	●	●	●	●
PAROTOF	●	●	●	●	●	●
PCALL	●	●	●	●	●	●
PDELAYOF	-	-	-	-	-	-
PDELAYON	-	-	-	-	-	-
PHI	-	-	-	-	-	-
PHU	●	●	●	●	●	●
PL	-	-	-	-	-	-
PM	●	●	●	●	●	●
PO	-	-	-	-	-	-
POCKET3	-	-	-	-	-	-
POCKET4	-	-	-	-	-	-

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
POLF	•	•	•	•	•	•
POLFA	•	•	•	•	•	•
POLFMASK	•	•	•	•	•	•
POLFMLIN	•	•	•	•	•	•
POLY	-	-	-	-	-	-
POLYPATH	-	-	-	-	-	-
PON	-	-	-	-	-	-
PONS	-	-	-	-	-	-
POS	•	•	•	•	•	•
POSA	•	•	•	•	•	•
POSM	•	•	•	•	•	•
POSMT	-	-	-	-	-	-
POSP	•	•	•	•	•	•
POSRANGE	•	•	•	•	•	•
POT	•	•	•	•	•	•
PR	•	•	•	•	•	•
PREPRO	•	•	•	•	•	•
PRESETON	•	•	•	•	•	•
PRESETONS	•	•	•	•	•	•
PRIO	•	•	•	•	•	•
PRLOC	•	•	•	•	•	•
PROC	•	•	•	•	•	•
PROTA	•	•	•	•	•	•
PROTD	•	•	•	•	•	•
PROTS	•	•	•	•	•	•
PSI	-	-	-	-	-	-
PTP	•	•	•	•	•	•
PTPG0	•	•	•	•	•	•
PTPWOC	•	•	•	•	•	•
PUNCHACC	-	-	-	-	-	-
PUTFTOC	•	•	•	•	•	•
PUTFTOCF	•	•	•	•	•	•
PW	○	○	○	○	○	○
QU	•	•	•	•	•	•
R...	•	•	•	•	•	•
RAC	•	•	•	•	•	•
RDISABLE	•	•	•	•	•	•
READ	•	•	•	•	•	•
REAL	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
RELEASE	•	•	•	•	•	•
REP	•	•	•	•	•	•
REPEAT	•	•	•	•	•	•
REPEATB	•	•	•	•	•	•
REPOSA	•	•	•	•	•	•
REPOSH	•	•	•	•	•	•
REPOSHA	•	•	•	•	•	•
REPOSL	•	•	•	•	•	•
REPOSQ	•	•	•	•	•	•
REPOSQA	•	•	•	•	•	•
RESETMON	•	•	•	•	•	•
RET	•	•	•	•	•	•
RETB	•	•	•	•	•	•
RIC	•	•	•	•	•	•
RINDEX	•	•	•	•	•	•
RMB	•	•	•	•	•	•
RME	•	•	•	•	•	•
RMI	•	•	•	•	•	•
RMN	•	•	•	•	•	•
RND	•	•	•	•	•	•
RNDM	•	•	•	•	•	•
ROT	•	•	•	•	•	•
ROTS	•	•	•	•	•	•
ROUND	•	•	•	•	•	•
ROUNDUP	•	•	•	•	•	•
RP	•	•	•	•	•	•
RPL	•	•	•	•	•	•
RT	•	•	•	•	•	•
RTLIOF	•	•	•	•	•	•
RTLION	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisningar S ... Z

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
S	•	•	•	•	•	•
SAVE	•	•	•	•	•	•
SBLOF	•	•	•	•	•	•
SBLON	•	•	•	•	•	•
SC	•	•	•	•	•	•
SCALE	•	•	•	•	•	•
SCC	•	•	•	•	•	•
SCPARA	•	•	•	•	•	•
SD	○	○	○	○	○	○
SET	•	•	•	•	•	•
SETAL	•	•	•	•	•	•
SETDNO	•	•	•	•	•	•
SETINT	•	•	•	•	•	•
SETM	-	-	•	-	-	•
SETMS	•	•	•	•	•	•
SETMS(n)	•	•	•	•	•	•
SETMTH	•	•	•	•	•	•
SETPIECE	•	•	•	•	•	•
SETTA	•	•	•	•	•	•
SETTCOR	•	•	•	•	•	•
SETTIA	•	•	•	•	•	•
SF	•	•	•	•	•	•
SIN	•	•	•	•	•	•
SIRELAY	-	-	-	-	-	-
SIRELIN	-	-	-	-	-	-
SIRELOUT	-	-	-	-	-	-
SIRELTIME	-	-	-	-	-	-
SLOT1	-	-	-	-	-	-
SLOT2	-	-	-	-	-	-
SOFT	•	•	•	•	•	•
SOFTA	•	•	•	•	•	•
SON	-	-	-	-	-	-
SONS	-	-	-	-	-	-
SPATH	•	•	•	•	•	•
SPCOF	•	•	•	•	•	•
SPCON	•	•	•	•	•	•
SPI	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tek Export (gce42)	SW26x(3) CNC-SW G-tek Export (gce62)	SW28x(1) CNC-SW G-tek Adv. Export (gce82)	SW24x(5) CNC-SW G-tek Export (gse42)	SW26x(3) CNC-SW G-tek Export (gse62)	SW28x(1) CNC-SW G-tek Adv. Export (gse82)
SPIF1	-	-	-	-	-	-
SPIF2	-	-	-	-	-	-
SPLINEPATH	○	○	○	○	○	○
SPN	-	-	-	-	-	-
SPOF	-	-	-	-	-	-
SPOS	•	•	•	•	•	•
SPOSA	•	•	•	•	•	•
SPP	-	-	-	-	-	-
SPRINT	•	•	•	•	•	•
SQRT	•	•	•	•	•	•
SR	•	•	•	•	•	•
SRA	•	•	•	•	•	•
ST	•	•	•	•	•	•
STA	•	•	•	•	•	•
START	-	-	•	-	-	•
STARTFIFO	•	•	•	•	•	•
STAT	•	•	•	•	•	•
STOLF	•	•	•	•	•	•
STOPFIFO	•	•	•	•	•	•
STOPRE	•	•	•	•	•	•
STOPREOF	•	•	•	•	•	•
STRING	•	•	•	•	•	•
STRINGFELD	•	•	•	•	•	•
STRINGIS	•	•	•	•	•	•
STRLEN	•	•	•	•	•	•
SUBSTR	•	•	•	•	•	•
SUPA	•	•	•	•	•	•
SVC	•	•	•	•	•	•
SYNFCT	•	•	•	•	•	•
SYNR	•	•	•	•	•	•
SYNRW	•	•	•	•	•	•
SYNW	•	•	•	•	•	•
T	•	•	•	•	•	•
TAN	•	•	•	•	•	•
TANG	○	○	○	○	○	○
TANGDEL	○	○	○	○	○	○
TANGOF	○	○	○	○	○	○
TANGON	○	○	○	○	○	○

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tek Export (gce42)	SW26x(3) CNC-SW G-tek Export (gce62)	SW28x(1) CNC-SW G-tek Adv. Export (gce82)	SW24x(5) CNC-SW G-tek Export (gse42)	SW26x(3) CNC-SW G-tek Export (gse62)	SW28x(1) CNC-SW G-tek Adv. Export (gse82)
TCA (828D: _TCA)	•	•	•	•	•	•
TCARR	•	•	•	•	•	•
TCI	•	•	•	•	•	•
TCOABS	•	•	•	•	•	•
TCOFR	•	•	•	•	•	•
TCOFRX	•	•	•	•	•	•
TCOFRY	•	•	•	•	•	•
TCOFRZ	•	•	•	•	•	•
THETA	-	-	-	-	-	-
TILT	-	-	-	-	-	-
TLIFT	○	○	○	○	○	○
TML	•	•	•	•	•	•
TMOF	•	•	•	•	•	•
TMON	•	•	•	•	•	•
TO	•	•	•	•	•	•
TOFF	•	•	•	•	•	•
TOFFL	•	•	•	•	•	•
TOFFOF	•	•	•	•	•	•
TOFFON	•	•	•	•	•	•
TOFFR	•	•	•	•	•	•
TOFRAME	•	•	•	•	•	•
TOFRAMEX	•	•	•	•	•	•
TOFRAMEY	•	•	•	•	•	•
TOFRAMEZ	•	•	•	•	•	•
TOLOWER	•	•	•	•	•	•
TOLENV	•	•	•	•	•	•
TOOLGNT	•	•	•	•	•	•
TOOLGT	•	•	•	•	•	•
TOROT	•	•	•	•	•	•
TOROTOF	•	•	•	•	•	•
TOROTX	•	•	•	•	•	•
TOROTY	•	•	•	•	•	•
TOROTZ	•	•	•	•	•	•
TOUPPER	•	•	•	•	•	•
TOWBCS	•	•	•	•	•	•
TOWKCS	•	•	•	•	•	•
TOWMCS	•	•	•	•	•	•
TOWSTD	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
TOWTCS	•	•	•	•	•	•
TOWWCS	•	•	•	•	•	•
TR	•	•	•	•	•	•
TRAANG	○	○	○	-	-	-
TRACON	○	○	○	-	-	-
TRACYL	○	○	○	○	○	○
TRAFOOF	•	•	•	•	•	•
TRAFOON	-	-	-	-	-	-
TRAILOF	•	•	•	•	•	•
TRAILON	•	•	•	•	•	•
TRANS	•	•	•	•	•	•
TRANSMIT	○	○	○	○	○	○
TRAORI	-	-	-	-	-	-
TRUE	•	•	•	•	•	•
TRUNC	•	•	•	•	•	•
TU	•	•	•	•	•	•
TURN	•	•	•	•	•	•
ULI	•	•	•	•	•	•
UNLOCK	•	•	•	•	•	•
UNTIL	•	•	•	•	•	•
UPATH	•	•	•	•	•	•
VAR	•	•	•	•	•	•
VELOLIM	•	•	•	•	•	•
VELOLIMA	•	•	•	•	•	•
WAITC	•	•	•	•	•	•
WAITE	-	-	•	-	-	•
WAITENC	•	•	•	•	•	•
WAITM	-	-	•	-	-	•
WAITMC	-	-	•	-	-	•
WAITP	•	•	•	•	•	•
WAITS	•	•	•	•	•	•
WALCS0	•	•	•	•	•	•
WALCS1	•	•	•	•	•	•
WALCS2	•	•	•	•	•	•
WALCS3	•	•	•	•	•	•
WALCS4	•	•	•	•	•	•
WALCS5	•	•	•	•	•	•
WALCS6	•	•	•	•	•	•
WALCS7	•	•	•	•	•	•

21.2 Anvisningar: Disponibilitet vid SINUMERIK 828D

Anvisning	SINUMERIK 828D					
• Standard ○ Option - inte disponibel	SW24x(5) CNC-SW G-tec Export (gce42)	SW26x(3) CNC-SW G-tec Export (gce62)	SW28x(1) CNC-SW G-tec Adv. Export (gce82)	SW24x(5) CNC-SW G-tec Export (gse42)	SW26x(3) CNC-SW G-tec Export (gse62)	SW28x(1) CNC-SW G-tec Adv. Export (gse82)
WALCS8	•	•	•	•	•	•
WALCS9	•	•	•	•	•	•
WALCS10	•	•	•	•	•	•
WALIMOF	•	•	•	•	•	•
WALIMON	•	•	•	•	•	•
WHEN	•	•	•	•	•	•
WHENEVER	•	•	•	•	•	•
WHILE	•	•	•	•	•	•
WORKPIECE	•	•	•	•	•	•
WRITE	•	•	•	•	•	•
WRTPR	•	•	•	•	•	•
X	•	•	•	•	•	•
XOR	•	•	•	•	•	•
Y	•	•	•	•	•	•
Z	•	•	•	•	•	•

21.3 Aktuellt språk i HMI

Den följande tabellen innehåller alla språk som finns tillgängliga på användargränssnittet.

Det aktuellt inställda språket kan avfrågas i detaljprogrammet och i synkronaktioner via följande systemvariabel:

\$AN_LANGUAGE_ON_HMI = <Wert>

<Wert>	Språk	Språkförkortning
1	Tyska (Tyskland)	DEU
2	Franska	FRA
3	Engelska (Förenade Kungariket)	ENG
4	Spanska	ESP
6	Italienska	ITA
7	Nederländska	NLD
8	Kinesiska (förenklad)	CHS
9	Svenska	SVE
18	Ungerska	HUN
19	Finska	FIN
28	Tjeckiska	CSY
50	Portugisiska (Brasilien)	PTB
53	Polska	PLK
55	Danska	DAN
57	Ryska	RUS
68	Slovakiska	SKY
72	Rumänska	ROM
80	Kinesiska (traditionell)	CHT
85	Koreanska	KOR
87	Japanska	JPN
89	Turkiska	TRK

Märk

En aktualisering av \$AN_LANGUAGE_ON_HMI görs:

- efter systemstart.
- efter NC- och/eller PLC-Reset.
- efter omkoppling till en annan NC inom ramen av M2N.
- efter språkomkoppling på HMI.

Bilaga

A.1 Förteckning över förkortningar

A	
A	Utgång
ADI4	Analog Drive Interface for 4 Axes
AC	Adaptive Control
ALM	Active Line Module
ARM	Asynkron rotationsmotorer
AS	Automatiseringssystem
ASCII	American Standard Code for Information Interchange: Amerikansk kod-norm för informationsutbyte
ASIC	Application Specific Integrated Circuit: Användar-kopplingskrets
ASUP	Asynkront underprogram
AUXFU	Auxiliary Function: Hjälpfunktion
AWL	Anvisningslista
AWP	Användarprogram

B	
BA	Driftssätt
BAG	Driftssättgrupp
BCD	Binary Coded Decimals: I binärkod kodade decimaltal
BERO	Beröringsfri lägeskopplare
BI	Binector Input
BICO	Binector Connector
BIN	Binary Files: Binärfiler
BIOS	Basic Input Output System
BKS	Baskoordinatsystem
BO	Binector Output
BTSS	Manöverpanelgränssnitt

C	
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CC	Compile Cycle: Compile-cykler
CEC	Cross Error Compensation
CI	Connector Input
CF-Card	Compact Flash-Card

C	
CNC	Computerized Numerical Control: Datoriserad numerisk styrning
CO	Connector Output
CoL	Certificate of License
COM	Communication
CPA	Compiler Projecting Data: Projekteringsdata för compilern
CRT	Cathode Ray Tube: Bildrör
CSB	Central Service Board: PLC-komponent
CU	Control Unit
CP	Communication Processor
CPU	Central Processing Unit: Central räknarenhet
CR	Carriage Return
CTS	Clear To Send: Meddelande om sändningsberedskap hos seriella data-gränssnitt
CUTCOM	Cutter Radius Compensation: Verktygsradiekompensering

D	
DAU	Digital-analog-omvandlare
DB	Datakomponent (PLC)
DBB	Datakomponent-byte (PLC)
DBD	Datakomponent-dubbelord (PLC)
DBW	Datakomponent-ord (PLC)
DBX	Datakomponent-bit (PLC)
DDE	Dynamic Data Exchange
DDS	Drive Data Set: Servodatablock
DIN	Tyska industrinormen
DIO	Data Input/Output: Dataöverförings-indikering
DIR	Directory: Katalog
DLL	Dynamic Link Library
DO	Drive Object
DPM	Dual Port Memory
DPR	Dual Port RAM
DRAM	Dynamiskt minne (obuffrat)
DRF	Differential Resolver Function: Differential-vridningsgivar-funktion (handratt)
DRIVE-CLiQ	Drive Component Link with IQ
DRY	Dry Run: Provkörningsmatning
DSB	Decoding Single Block: Dekoderingsenkelblock
DSC	Dynamic Servo Control / Dynamic Stiffness Control
DW	Dataord
DWORD	Dubbelord (aktuellt 32 bit)

E	
E	Ingång
EES	Execution from External Storage
E/A	In-/utmatning
ENC	Encoder: Ärvärdesgivare
EPF	Enkel periferimodul (PLC–E/A–komponent)
EGB	Elektroniskt utsatta komponenter/element
EMC	Elektromagnetisk kompatibilitet
EN	Europeisk Standard
ENC	Encoder: Ärvärdesgivare
EnDat	Givargränssnitt
EPROM	Erasable Programmable Read Only Memory: Raderbart, elektriskt programmerbart endast läsminne
ePS Network Services	Tjänster för fjärrservice för maskiner via internet
EQN	Typbeteckning för en absolutvärdesgivare med 2048 sinussignaler/varv
ES	Engineering System
ESR	Utvidgat stoppande och återdragande
ETC	ETC–knapp ">"; Utvidgning av funktionstangentraden i samma meny

F	
FB	Funktionskomponent (PLC)
FC	Function Call: Funktionskomponent (PLC)
FEPROM	Flash–EPROM: Läs- och skrivbart minne
FIFO	First In First Out: Minne, som arbetar utan adressuppgift och vars data läses i samma ordningsföljd som de sparades
FIPO	Fininterpolator
FPU	Floating Point Unit: Flyttalsenhet
FRK	Fräsradielkompensering
FST	Feed Stop: Matning stopp
FUP	Funktionsplan (programmeringsmetod för PLC)
FW	Firmware

G	
GC	Global Control (PROFIBUS: Broadcast-Telegramm)
GDIR	Globalt detaljprogramminne
GEO	Geometri, t.ex. geometriaxel
GIA	Gear Interpolation Data: Växelinterpoleringsdata
GND	Signal Ground
GP	Grundprogram (PLC)
GS	Växelsteg
GSD	Instrumenthuvudfil för beskrivning av en PROFIBUS slave

G	
GSDML	Generic Station Description Markup Language: XML-baserat beskrivningsspråk för upprättande av en GSD-fil
GUD	Global User Data: Globala användardata

H	
HEX	Kort beteckning för hexadecimalt tal
HiFu	Hjälpfunktion
HLA	Hydraulisk linjär drivning
HMI	Human Machine Interface: SINUMERIK-användargränssnitt
HSA	Huvudspindel drift
HW	Hardware

I	
IBN	Idrifttagning
IKA	Interpolerings-kompensation
IM	Interface-Modul: Tillkopplingskomponent
IMR	Interface-Modul Receive: Tillkopplingskomponent för mottagningsdrift
IMS	Interface-Modul Send: Tillkopplingskomponent för sändningsdrift
INC	Increment: Stegmått
INI	Initializing Data: Initieringsdata
IPO	Interpolator
ISA	International Standard Architecture
ISO	International Standard Organization

J	
JOG	Jogging: Rigningsdrift

K	
K_v	Förstärkningsfaktor för regleringskretsen
K_p	Proportionalförstärkning
K_U	Utväxlingsförhållande
KOP	Kontaktplan (programmeringsmetod för PLC)

L	
LAI	Logic Machine Axis Image: Logisk maskinaxel-avbild
LAN	Local Area Network
LCD	Liquid-Crystal Display: Bildskärm med flytande kristaller
LED	Light Emitting Diode: Lysdiod
LF	Line Feed

L	
LMS	Lägesmätsystem
LR	Lägesregulator
LSB	Least Significant Bit: Bi med lägsta värde
LUD	Local User Data: Användardata (lokal)

M	
MAC	Media Access Control
MAIN	Main program: Huvudprogram (OB1, PLC)
MB	Megabyte
MCI	Motion Control Interface
MCIS	Motion–Control–Information–System
MCP	Machine Control Panel: Maskinens styrpanel
MD	Maskindatum resp. maskindata
MDA	Manual Data Automatic: Manuell inmatning
MDS	Motor Data Set: Motordatablock
MELDW	Meddelandeord
MKS	Maskinkoordinatsystem
MM	Motor Module
MPF	Main Program File: Huvudprogram (NC)
MSTT	Maskinens styrpanel

N	
NC	Numerical Control: Numerisk styrning med blockförberedelse, förflyttningsområde osv.
NCU	Numerical Control Unit: Hardware-enhet till NC
NRK	Beteckning för operativsystemet till NC
NST	Gränssnittssignal
NURBS	Non-Uniform Rational B-Spline
NV	Nollpunktsförflyttning
NX	Numerical Extension: Axelutvidgningskomponent

O	
OB	Organisationskomponent i PLC
OEM	Original Equipment Manufacturer
OP	Operation Panel: Manöveranordning
OPI	Operation Panel Interface: Manöverpanel-gränssnitt
OPT	Options: Optioner
OLP	Optical Link Plug: Busskontakt för fiberoptikledare
OSI	Open Systems Interconnection: Normering för räknarkommunikation

P	
PAA	Processavbild för utgångarna
PAE	Processavbild för ingångarna
PC	Personal Computer
PCIN	Namn på SW för datautbyte med styrningen
PCMCIA	Personal Computer Memory Card International Association: Minneskort-standardisering
PCU	PC Unit: PC-box (räknarenhet)
PG	Programmeringsinstrument
PKE	Parameterkännetecken: Del av ett PKW
PKW	Parameterkännetecken: Värde (parameterdel av ett PPO)
PLC	Programmable Logic Control: Anpassnings-styrning
PN	PROFINET
PNO	PROFIBUS-användarorganisation
PO	POWER ON
POE	Programorganisationsenhet
POS	Position/positionera
POSMO A	Positioning Motor Actuator: Positionermotor
POSMO CA	Positioning Motor Compact AC: Komplet drivenhet med integrerad effekt- och regleringskomponent samt positionerenhet och programminne; växelströms-inmatning
POSMO CD	Positioning Motor Compact DC: som CA, dock likströmsinmatning
POSMO SI	Positioning Motor Servo Integrated: Positionermotor; likströmsinmatning
PPO	Parameter Processdata Objekt ; cykliskt datatelegram vid en överföring med PROFIBUS-DP och profil "Varvtalsföränderliga drivningar"
PPU	Panel Processing Unit (central hardware för en panel-baserad CNC-styrning t.ex. SI-NUMERIK 828D)
PROFIBUS	Process Field Bus: Seriell databuss
PRT	Programtest
PSW	Programstyrord
PTP	Point to Point: Punkt till punkt
PUD	Program Global User Data: Programglobal användarvariabel
PZD	Processdata: Processdatadel i en PPO

Q	
QFK	Kvadrantfel kompensering

R	
RAM	Random Access Memory: Skriv-/läs-minne
REF	Funktion Referenspunktköra
REPOS	Funktion Repositionera
RISC	Reduced Instruction Set Computer: Processortyp med litet kommandoblock och snabbt kommandoflöde
ROV	Rapid Override: Ingångskorrigerig

R	
RP	R-parameter, räkneparameter, fördefinierad användarvariabel
RPA	R-Parameter Active: Minnesområde i NC för R-parameternummer
RPY	Roll Pitch Yaw: Vridningstyp för ett koordinatsystem
RTL	Rapid Traverse Linear Interpolation: Linjär interpolering vid snabbtransportrörelse
RTS	Request To Send: Koppla till sändningsdel, styrsignal från seridella data-gränssnitt
RTCP	Real Time Control Protocol

S	
SA	Synkronaktion
SBC	Safe Break Control: Säker bromsutstyrning
SBL	Single Block: Enkelblock
SBR	Subroutine: Underprogram (PLC)
SD	Settingdatum resp. settingdata
SDB	System datakomponent
SEA	Setting Data Active: Kännetecken (filtyp) för settingdata
SERUPRO	Search-Run by Program Test: Blocksökning via programtest
SFB	System funktionskomponent
SFC	System Function Call
SGE	Säkerhetsriktad ingång
SGA	Säkerhetsriktad utgång
SH	Säkert stopp
SIM	Single in Line Module
SK	Softkey
SKP	Skip: Funktion för att gömma ett detaljprogramblock
SLM	Synkron linjärmotor
SM	Stegmotor
SMC	Sensor Module Cabinet Mounted
SME	Sensor Module Externally Mounted
SMI	Sensor Module Integrated
SPF	Sub Program File: Underprogram (NC)
SPS	Minnesprogrammerbar styrning = PLC
SRAM	Statiskt minne (buffrat)
SRK	Skärradiekompensering
SRM	Synkron rotationsmotorer
SSFK	Spindelstigningsfelkompensering
SSI	Serial Synchron Interface: Seriellt synkront gränssnitt
SSL	Blocksökning
STW	Styrord
SUG	Skivperiferihastighet
SW	Software
SYF	System Files: Systemfiler
SYNACT	Synchronized Action: Synkronaktion

T	
TB	Terminal Board (SINAMICS)
TCP	Tool Center Point: Verktygsspets
TCP/IP	Transport Control Protocol / Internet Protocol
TCU	Thin Client Unit
TEA	Testing Data Active: Kännetecken för maskindata
TIA	Totally Integrated Automation
TM	Terminal Module (SINAMICS)
TO	Tool Offset: Verktygskompensering
TOA	Tool Offset Active: Kännetecken (filtyp) för verktygskompenseringar
TRANSMIT	Transform Milling Into Turning: Koordinattransformation för fräsbearbetningar på en svarvmaskin
TTL	Transistor–Transistor–Logik (gränssnitt-typ)
TZ	Teknologicykel

U	
UFR	User Frame: Nollpunktsförflyttning
UP	Underprogram
USB	Universal Serial Bus
USV	Avbrottsfri strömförsörjning

V	
VDI	Internt kommunikationsgränssnitt mellan NC och PLC
VDI	Verein Deutscher Ingenieure (tyskt ingenjörsförbund)
VDE	Verband Deutscher Elektrotechniker (tyskt elektrikerförbund)
VI	Voltage Input
VO	Voltage Output
VSA	Matningsdrift

W	
WAB	Funktion Mjuk fram- och bortkörning
WKS	Arbetskoordinatsystem
WKZ	Verktyg
WLK	Verktygslängdkompensering
WOP	Verkstads-orienterad programmering
WPD	Work Piece Directory: Arbetsstycks katalog
WRK	Verktygsradiekompensering
WZ	Verktyg
WZK	Verktygskompensering
WZV	Verktygsförvaltning
WZW	Verktygsväxling

X	
XML	Extensible Markup Language

Z	
ZOA	Zero Offset Active: Kännetecken för nollpunktsförflyttningar
ZSW	Tillståndsort (för drivningen)

A.2 Dokumentationsöversikt

Allmän dokumentation



Reklamhäfte
– SINUMERIK 840D sl
– SINUMERIK 828D
– SINUMERIK 828D BASIC



Katalog NC 62
SINUMERIK 840D sl



Katalog NC 82
SINUMERIK 828D



Katalog PM 21
SIMOTION,
SINAMICS S120



Systemhandbok
Smart verktygs-
maskinmanövrering

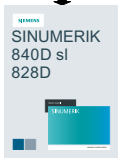


Systemhandbok
Ctrl-Energy

Användardokumentation



Användarhandbok
– Universellt
– Svarva
– Fräsa
– Slipa



Programmeringshandbok
– Grunder
– Arbetsförberedelse
– Mätcykler



Programmeringshandbok
– ISO svarva
– ISO fräsa



Diagnoshandbok
Larm

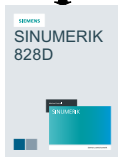


Diagnoshandbok
Larm

Tillverkar- / service-dokumentation



Instrumenthandbok
– NCU
– Manöverkomponenter
och nätverk
– ADI4



Instrumenthandbok
Idriftagningshandbok
Servicehandbok



Idriftagningshandbok
– CNC: NCK, PLC,
Servo
– Bassoftware och
manöver-software



Listhandbok
– Maskindata
– Gränssnittssignaler
– Variabler



Listhandbok
– Maskindata
– Gränssnittssignaler
– Parametrar
– Variabler

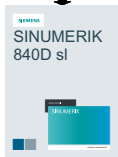


Systemhandbok
Vägledning för
maskin-
projektering

Tillverkar- / service-dokumentation



Funktionshandbok
– Grundfunktioner
– Tilläggfunktioner
– Specialfunktioner
– Synkronaktioner
– ISO-dialekter



Funktionshandbok
Verktögsförvaltning



Funktionshandbok
Safety Integrated



Funktionshandbok
Safety Integrated



Projekteringshandboken
– EMC-uppbyggnadsdirektiv
– Industrial Security

Info/utbildning



Utbildningsdokument
– Enkel fräsning
med ShopMill
– Enkel svarvning
med ShopTurn



Handbok
Verktögs- och
formtillverkning

Elektronisk dokumentation



DOConCD



Industry Online
Support (SIOS)



Industry Mail

Ordlista

Absolut måttsättning

Uppgift av målet för en axelrörelse med ett mått, som hänför sig till nollpunkten för det momentant giltiga koordinatsystemet. Se → kedjemått.

Acceleration med jerkbegränsning

För att uppnå ett optimalt accelerationsbeteende för maskinen under det att mekaniken samtidigt skonas kan i bearbetningsprogrammet kopplas om mellan språngartad acceleration och jämn (ryckfri) acceleration.

Adress

En adress är kännetecknet för en bestämd operand eller ett operandområde, t.ex. ingång, utgång osv.

Användardefinierad variabel

Användare kan för valfri användning i → detaljprogrammet eller datablocket (globala användardata) komma överens om användardefinierade variabler. En definition innehåller en datatypuppgift och variabelnamnet. Se → systemvariabel.

Användargränssnitt

Användargränssnittet (BOF) är visningsmediet för en CNC-styrning i form av en bildskärm. Den är gestaltad med horisontella och vertikala funktionstangenter.

Användarminne

Alla program och data som detaljprogram, underprogram, kommentarer, verktygskompenseringar, nollpunktsförflyttningar/frames samt kanal- och programanvändardata kan läggas i det gemensamma CNC-användarminnet.

Användarprogram

Användarprogram för automatiseringssystem S7-300 upprättas med programmeringsspråket STEP 7. Användarprogrammet är uppbyggt av moduler och består av enskilda block.

De grundläggande blocktyperna är:

- Code-block
Dessa block innehåller STEP 7-kommandon.
- Datablock
Dessa block innehåller konstanter och variabler för STEP 7-programmet.

Arbetsfältsbegränsning

Med arbetsfältsbegränsning kan axlarnas förflyttningsområde, förutom med gränsbrytare, inskränkas ytterligare. Per axel är ett värdepar för beskrivning av det skyddade arbetsutrymmet möjligt.

Arbetskoordinatsystem

Arbetsstyckskoordinatsystemet har sin utgångspunkt i → arbetsstycksnollpunkten. Vid programmering i arbetsstyckskoordinatsystemet hänför sig mått och ritningar till detta system.

Arbetsminne

Arbetsminnet är ett RAM-minne i → CPU, genom vilket processorn har åtkomst till användarprogrammet under programbearbetningen.

Arbetsstycke

Detalj som ska tillverkas/bearbetas av verktygsmaskinen.

Arbetsstyckskontur

Börkontur för det → arbetsstycke som ska tillverkas/bearbetas.

Arbetsstycksnollpunkt

Arbetsstycksnollpunkten bildar utgångspunkten för → arbetsstyckskoordinatsystemet. Den definieras genom avstånd till → maskinnollpunkten.

Arbetsutrymme

Tredimensionellt utrymme i vilket verktygsspetsen kan köra in på grund av verktygsmaskinens konstruktion. Siehe → skyddsutrymme.

Arkivera

Utläsning av filer och/eller kataloger till ett **externt** minne.

Asynkront underprogram

Detaljprogram, som asynkront (oberoende) till det aktuella programtillståndet kan startas med en interruptsignal (t.ex. signal "snabb NC-ingång").

Automatik

Driftsläge hos styrningen (blockföljddrift enligt DIN): Driftsläge hos NC-systemen, i vilket ett → detaljprogram väljs och kontinuerligt genomarbetas.

Axeladress

Se → axelnamn

Axelnamn

För entydig identifikation måste alla kanal- och → maskinaxlar i styrningen betecknas med kanal- resp. styrningsomfattande entydiga namn. → geometriaxlarna benämns med X, Y, Z. De runt geometriaxlarna vridande → roterande axlarna benämns med A, B, C.

Axlar

CNC-axlarna delas enligt omfattningen av deras funktioner in i:

- Axlar: interpolerande banaxlar
- Hjälpaxlar: ej interpolerande ansättnings- och positioneringsaxlar med axelspecifikt matning. Hjälpaxlar är inte delaktiga i den egentliga bearbetningen, t.ex. verktygstransportör, verktygsmagasin.

Banaxel

Banaxlar är alla bearbetningsaxlar i → kanalen som styrs så av → interpolatorn, att de startar, accelererar, stoppar och uppnår slutpunkten samtidigt.

Banhastighet

Den maximalt programmerbara banhastigheten är beroende av finheten hos inmatningen. Vid en upplösning på till exempel 0,1 mm uppgår den maximalt programmerbara banhastigheten till 1000 m/min.

Banmatning

Banmatningen verkar på → banaxlarna. Den utgör den geometriska summan av matningarna för de deltagande → geometriaxlarna.

Banstyrningsdrift

Målet för banstyrningsdriften är att undvika en kraftigare nedbromsning av → banaxlarna vid detaljprogrammets blockgränser för att växla till nästa block med så lika banhastighet som möjligt.

Basaxel

Axel, vars bör- eller ärvärde används för beräkningen av ett kompenseringsvärde.

Baskoordinatsystem

Kartesiskt koordinatsystem, avbildas genom transformation på maskinkoordinatsystemet.

I → detaljprogrammet använder programmeraren axelnamnen i baskoordinatsystemet. Det står, när ingen → transformation är aktiv, parallellt till → maskinkoordinatsystemet. Skillnaden till detta ligger i → axelnamnen.

Baudrate

Hastighet vid dataöverföringen (bit/s).

Bearbetningskanal

Bitider kan via en kanalstruktur förkortas genom parallella rörelseförlopp, t.ex. förflyttning av en laddningsportal simultant till bearbetningen. En CNC-kanal ska därvid anses som egen CNC-styrning med avkodning, blockförberedelse och interpolering.

Beteckning

Orden enligt 66025 kompletteras med beteckning (namn) för variabeln (räknevariabel, systemvariabel, användarvariabel), för underprogram, för nyckelord och ord med flera adressbokstäver. Dessa kompletteringar är lika betydande som orden vid blockuppbyggnaden. Beteckningar måste vara entydiga. Samma beteckning får inte användas för olika objekt.

Block

Som block betecknas alla filer, som behövs för upprättande och genomarbetning av program.

Blocksökning

För att testa detaljprogram eller efter avbrott av en bearbetning kan ett valfritt ställe i detaljprogrammet, vid vilket bearbetningen ska startas eller fortsättas, väljas via funktionen "Blocksökning".

Boot

Laddning av systemprogrammet efter Power On.

Buffertbatteri

Buffertbatteriet garanterar att → användarprogrammet i → CPU är säkert lagrat i händelse av strömbavbrott och att fastlagda dataområden och flaggor, tider och räknare bevaras remanent.

Böjning

Böjningen k för en kontur är den inverterade radien r för den cirkel som ligger an i en konturpunkt ($k = 1/r$).

C-axel

Axel runt vilken en styrd vridrörelse och positionering med arbetsstycksspindel görs.

Cirkelinterpolering

Mellan fastlagda punkter på konturen ska → verktyget köra på en cirkel med en given matning och därvid bearbeta arbetsstycket.

CNC

Se → NC

Computerized Numerical Control: omfattar komponenterna → NC, → PLC, HMI, → COM.

CNC

Se → NC

Computerized Numerical Control: omfattar komponenterna → NC, → PLC, HMI, → COM.

COM

Komponent i NC-styrningen för genomförande och koordination av kommunikation.

CPU

Central Processing Unit, se → minnesprogrammerbar styrning

C-spline

C-splinen är den mest kända och mest använda splinen. Övergångarna vid stödpunkterna är tangent- och böjningskontinuerliga. Polynom av 3:e graden används.

Cykler

Skyddat underprogram för utförande av upprepat uppträdande bearbetningsförlopp på → arbetsstycket.

Datablock

1. Dataenhet i → PLC, till vilken → HIGHSTEP-programmen har åtkomst.
2. Dataenhet i → NC: Datablock innehåller datadefinitioner för globala användardata. Data kan initieras direkt vid definitionen.

Dataord

Två byte stor dataenhet inom ett → Datablock.

Dataöverföringsprogram PCIN

PCIN är ett hjälpprogram för att sända och mottaga CNC-användardata vid det seriella gränssnittet, som t.ex. detaljprogram, verktygskompenseringar etc. PCIN-programmet kan köras under MS-DOS på standard-industri-PC.

Detaljkontur

Kontur på det färdigbearbetade arbetsstycket. Se → råämne.

Detaljprogram

Följd av anvisningar till NC-styrningen som totalt förorsakar skapandet av ett visst → arbetsstycke. Likaså namn på en viss bearbetning av ett givet → råämne.

Detaljprogramblock

Del i ett → detaljprogram avgränsad genom Line Feed. Det skiljs mellan → huvudblock och → sidoblock.

Detaljprogramförvaltning

Detaljprogramförvaltningen kan organiseras efter → arbetsstycken. Användarminnets storlek bestämmer antalet program och data som kan förvaltas. Varje fil (program och data) kan föras med ett namn på maximalt 24 alfanumeriska tecken.

Diagnos

1. Manöverområde i styrningen
2. Styrningen har både ett självdiagnosprogram och även testhjälp för servicen: Status-, larm- och serviceindikeringar

DRF

Differential Resolver Function: NC-funktion, som i förbindelse med en elektronisk handratt skapar en inkrementell nollpunktsförflyttning i automatikdrift.

Driftsläge

Förloppskoncept för driften av en SINUMERIK-styrning. Driftslägena → Jog, → MDA, → Automatik är definierade.

Driftslägesgrupp

Teknologist sammanhörande axlar och spindlar kan sammanfattas till en driftslägesgrupp (BAG). Axlar/spindlar i en BAG kan styras från en eller flera → kanaler. Kanalerna i BAG är alltid tillordnade samma → driftsläge.

Drivning

Drivningen är den enhet i CNC som utför varvtals- och momentregleringen med hjälp av uppgifter från NC.

Editor

Editorn möjliggör upprättande, ändring, komplettering, sammanskjutning och infogande av program/texter/programblock.

Extern nollpunktsförflyttning

Av → PLC föreskriven nollpunktsförflyttning.

Fast maskinpunkt

Av verktygsmaskinen entydigt definierad punkt, t.ex. maskin-referenspunkt.

Frame

En frame är en räkneförskrift, som överför ett kartesiskt koordinatsystem till ett annat kartesiskt koordinatsystem. En frame innehåller komponenterna → nollpunktsförflyttning, → rotation, → skalning, → spegling.

Förflyttningsområde

Det maximalt tillåtna förflyttningsområdet för linjäraxlar uppgår till ± 9 dekader. Det absoluta värdet är beroende av den valda finheten för inmatning och lägesreglering och enhetssystemet (inch eller metriskt).

Förkoincidens

Blockbyte redan när banvägen har närmast sig slutpositionen med ett föreskrivet delta.

Förstyrning, dynamisk

Bristande noggrannhet i → konturen, orsakat av släpfel, låter sig nästan elimineras genom den dynamiska accelerationsberoende förstyrningen. Detta resulterar i en utomordentlig bearbetningsnoggrannhet också vid höga → banhastigheter. Förstyrningen kan axelspecifikt väljas till och från via → detaljprogrammet.

Geometri

Beskrivning av ett → arbetsstycke i → arbetsstyckskoordinatsystemet.

Geometriaxel

Geometriaxlarna bildar det 2- resp. 3-dimensionella → arbetsstyckskoordinatsystemet i vilket i → detaljprogrammen geometrin för arbetsstycket programmeras.

Gränsvarvtal

Maximalt/minimalt (spindel-)varvtal: Genom föreskrivna maskindata, → PLC eller → settingdata kan det maximala varvtalet för en spindel varv begränsat.

Gängtappning utan flytande gänghållare

Med denna funktion kan gängor borrar utan flytande gänghållare. Genom det interpolerande beteendet hos spindeln som roterande axel och borraraxeln skärs gängor exakt till slutborrdjup, t.ex. gängat bottenhål (förutsättning: axeldrift hos spindeln).

Hastighetsstyrning

För att vid förflyttningsrörelser med mycket små belopp per block kunna uppnå en acceptabel förflyttningshastighet, kan framåtblickande utvärdering över flera block (→ Look Ahead) ställas in.

HIGHSTEP

Sammanfattning av programmeringsmöjligheterna för → PLC i systemet AS300/AS400.

Hjälpfunktioner

Med hjälpfunktioner kan i → detaljprogram → parametrar överföras till → PLC som där utlöser av maskintillverkaren definierade reaktioner.

Huvudblock

Med ":" inlett block som innehåller alla uppgifter för att kunna starta arbetsförloppet i ett → detaljprogram.

Huvudprogram

Beteckningen "huvudprogram" stammar ännu från den tiden när detaljprogram var fast indelade i huvud- och → underprogram. Denna fasta indelning finns inte längre med det nutida SINUMERIK NC-språket. Principiellt kan varje detaljprogram i kanalen väljas och startas. Det går då i → programnivå 0 (huvudprogram-nivå). I huvudprogrammet kan ytterligare detaljprogram eller → cykler anropas som underprogram

HW-konfig

SIMATIC S7-Tool för konfigurering och parametrering av hardware-komponenter inom ett S7-projekt.

Högnivåspråk CNC

Högnivåspråket tjänar till att skriva NC-program, → synkronaktioner och → cykler. Det erbjuder: Kontrollstrukturer, → användardefinierad variabel, → systemvariabel, → makroteknik.

Interpolator

Logisk enhet i → NC, som efter angivande av målpositioner i detaljprogrammet bestämmer mellanvärden för de rörelser som ska köras i de enskilda axlarna.

Interpolerings-kompensation

Med interpoleringskompensationer som → spindelstigningsfel-, nedhållnings-, vinklighets- och temperaturkompensation kompenseras maskinens mekaniska fel.

Interruptrutin

Interruptrutiner är speciella → underprogram, som genom händelser (externa signaler) kan startas av bearbetningsprocessen. Ett detaljprogramblock som befinner sig under genomarbetning stoppas, stoppositionen för axlarna sparas automatiskt.

JOG

Driftläge i styrningen (riggningsdrift): I driftläget JOG kan maskinen ställas in. Enskilda axlar och spindlar kan förflyttas med riktningstangenterna i stegvis drift. Ytterligare funktioner i driftläget JOG är → referenspunktkörning, → Repos samt → Preset (sätta ärvärde).

Kanal

En kanal är kännetecknad av att den oberoende av andra kanaler kan genomföra ett → detaljprogram. En kanal styr exklusivt de axlar och spindlar som tillordnats den. Detaljprogramförlopp för olika kanaler kan koordineras genom → synkronisering.

Kedjemått

Också inkrementmått: Angivande av ett mål för en axels rörelse genom en vägsträcka som ska köras och en riktning relaterad till en redan uppnådd punkt. Se → absolutmått.

Kompenseringsaxel

Axel vars bör- eller ärvärde modifieras av kompenseringsvärdet.

Kompenseringsminne

Dataområde i styrningen i vilket verktygskompenseringsdata finns lagrade.

Kompenseringsstabell

Tabell med stödpunkter. Den levererar för utvalda positioner hos basaxeln kompenseringsvärdena för kompenseringssaxeln.

Kompenseringsvärde

Skillnaden mellan den av mätgivaren uppmätta axelpositionen och den önskade, programmerade axelpositionen.

Kontur

Konturen till → arbetsstycket

Konturövervakning

Som mått på konturprecisionen övervakas släpfelet inom ett definierbart toleransband. Ett otillåtet högt släpfel kan uppstå t.ex. genom överbelastning av drivningen. I detta fall leder det till ett larm och axlarna stoppas.

Koordinatsystem

Se → maskinkoordinatsystem, → arbetsstyckskoordinatsystem

KV

Kretsförstärkningsfaktor, regleringsteknisk storhet för en regleringskrets

Kvadrantfelkompensering

Konturfel vid kvadrantövergångar, som uppstår genom växlande friktionsförhållanden vid styrbanorna, kan i stor utsträckning elimineras med kvadrantfelkompensering. Parametreringen för kvadrantfelkompenseringen sker med en cirkelformtest.

KÜ

Utväxlingsförhållande

Köra till fast punkt

Verktygsmaskiner kan uppsöka fasta punkter som verktygsväxlingspunkt, inladdningspunkt, palettväxlingspunkt etc. definierat. Koordinaterna för dessa punkter finns lagrade i styrningen. Styrningen förflyttar de berörda axlarna, om möjligt i → snabbgång.

Laddningsminne

Laddningsminnet är för CPU 314 till → SPS lika med → arbetsminnet.

Larm

Alla → meddelanden och larm visas på manöverpanelen i klartext med datum och tid och den tillhörande symbolen för raderkriteriet. Indikeringen sker åtskilt efter larm och meddelanden.

1. Larm och meddelanden i detaljprogrammet
Larm och meddelanden kan föras till indikering i klartext direkt från detaljprogrammet.
2. Larm och meddelanden från PLC
Larm och meddelanden från maskinen kan föras till indikering i klartext direkt från PLC-programmet. För detta är inga ytterligare funktionsblock-paket nödvändiga.

Linjäraxel

Linjäraxeln är en axel vilken i motsats till den roterande axeln beskriver en rät linje.

Linjärinterpolering

Verktyget flyttas längs en rät linje till målpunkten och bearbetar därvid arbetsstycket.

Look Ahead

Med funktionen **Look Ahead** uppnås genom att "Titta i förväg" via ett parameterbart antal förflyttningsblock en optimal bearbetningshastighet.

Lutande bearbetning

Borr- och fräsbearbetningar på arbetsstycksytor som inte ligger i maskinens koordinatplan, kan komfortabelt utföras med stöd av funktionen "Lutande bearbetning".

Makroteknik

Sammanfattning av en mängd anvisningar under en beteckning. Beteckningen representerar i programmet mängden sammanfattade anvisningar.

Maskinaxlar

I verktygsmaskinen fysikaliskt existerande axlar.

Maskinens styrpanel

Verktygsmaskinens manöverpanel med manöverelementen tangenter, vred osv. ock enkla indikeringselement som LEDs. De tjänar till omedelbar påverkan av verktygsmaskinen via PLC.

Maskinkoordinatsystem

Koordinatsystem som är relaterat till axlarna i verktygsmaskinen.

Maskinnollpunkt

Fast punkt i verktygsmaskinen till vilken alla (härledda) mätsystem låter sig föras tillbaka.

Massa

Som massa gäller summan av alla med varandra förbundna inaktiva delar i en drivutrustning, som inte heller vid fel kan anta farlig beröringsspänning.

Matningsövermanning

Den programmerade hastigheten överlagras av den aktuella hastighetsinställningen via → maskinens styrpanel eller av → PLC (0-200%). Matningshastigheten kan dessutom korrigeras i bearbetningsprogrammet av en programmerbar procentfaktor (1-200%).

MDA

Driftläge i styrningen: Manuell Data Automatik. I driftläget MDA kan enskilda programblock eller blockföljder matas in utan referens till ett huvud- eller underprogram och sedan via tangenten NC-start genast utföras.

Meddelanden

Alla i detaljprogrammet programmerade meddelanden och av systemet identifierade → larm visas på manöverpanelen i klartext med datum och tid och den tillhörande symbolen för raderkriteriet. Indikeringen sker åtskilt efter larm och meddelanden.

Medmatningsaxel

Medmatningsaxeln är den → gantry-axel, vars börposition alltid härleds från förflyttningsrörelsen hos → styraxeln och därmed förflyttas synkront. Ur operatörens och programmerarens synvinkel "finns inte" medmatningsaxeln.

Mellanblock

Förflyttningsrörelser med vald → verktygskompensering (G41/G42) får stoppas av ett begränsat antal mellanblock (block utan axelrörelser i kompenseringsplanet), varvid verktygskompenseringen kan räknas in korrekt. Det tillåtna antalet mellanblock, som styrningen läser i förväg, kan ställas in via systemparameter.

Metriskt mätsystem

Normerat system av enheter: för längder t.ex. mm (millimeter), m (meter).

Minnesprogrammerbar styrning

Minnesprogrammerbara styrningar (SPS) är elektroniska styrningar, vars funktion som program är sparad i styrningsinstrumentet. Uppbyggnad av och kabeldragning i instrumentet beror alltså inte av styrningens funktion. Den minnesprogrammerbara styrningen har

strukturen som en dator; den består av CPU (centralkomponent) med minne, in-/utmatningskomponenter och internt buss-system. Periferin och programmeringsspråket är inställda för styrningsteknikens krav.

Mjukvarugränsbrytare

Mjukvarugränsbrytare begränsar förflyttningsområdet för en axel och förhindrar att sliden kör på hårdvarugränsbrytare. Per axel kan 2 värdepar föreskrivas, som kan aktiveras separat via → PLC.

Måttuppgift metrisk och inch

I bearbetningsprogram kan positions- och stigningsvärden programmeras i inch. Oberoende av den programmerbara måttuppgiften (*G70/G71*) ställs styrningen in på ett grundsystem.

NC

Numerical Control komponenter i → CNC, som genomarbetar → detaljprogrammen och koordinerar verktygsmaskinens rörelseförlopp.

Nollpunktsförflyttning

Uppgift av en ny referenspunkt för ett koordinatsystem genom referens till en bestående nollpunkt och en → frame.

1. Inställbar
Ett projekterbart antal inställbara nollpunktsförflyttningar står till förfogande för varje CNC-axel. De via G-kommandona valbara förflyttningarna är alternativt verksamma.
2. Extern
Förutom alla förflyttningar som läget för arbetsstycks-nollpunkten fastlägger kan en extern nollpunktsförflyttning genom handratt (DRF-förflyttning) eller av PLC överlagras.
3. Programmerbar
Med anvisningen *TRANS* är nollpunktsförflyttningar programmerbara för alla ban- och positioneringsaxlar.

NRK

Numeric Robotic Kernel (operativsystem för → NC)

NURBS

Den styrningsinterna rörelsestyrningen och baninterpoleringen genomförs med NURBS (**N**on **U**niform **R**ational **B-S**plines) som bas. Därmed står styrningsinternt ett enhetligt förfarande till förfogande för alla interpoleringar.

Nyckelbrytare

Nyckelbrytaren på → maskinens styrpanel har 4 lägen, som är belagda med funktioner av styrningens operativsystem. Till nyckelbrytaren hör tre olikfärgade nycklar, som kan dras ur i de angivna lägena.

Nyckelord

Ord med fastlagt skrivsätt som har en definierad betydelse i programmeringsspråket för → detaljprogrammen.

Nät

Ett nät är en förbindelse av flera S7-300 och ytterligare periferiinstrument, t.ex. en PG, via → förbindningskabel. Via nätet sker ett utbyte av data mellan de anslutna instrumenten.

OEM

För maskintillverkare, som skapar sitt eget användargränssnitt eller vill införa teknologispecifika funktioner i styrningen, är fritt utrymme för individuella lösningar (OEM-applikationer) inplanerat.

Orienterat spindelstopp

Stopp för arbetsstycksspindeln i föreskrivet vinkelläge, t.ex. för att göra en extra bearbetning på ett visst ställe.

Periferikomponent

Periferikomponenter upprättar förbindelse mellan CPU och process.

Periferikomponenter är:

- → digitala in-/utmatningsgrupper
- → analoga in-/utmatningsgrupper
- → simulatorkomponenter

PLC

Programmable Logic Control: → Minnesprogrammerbar styrning. Komponent i → NC: Anpassnings-styrning för bearbetning av kontroll-logiken i verktygsmaskinen.

PLC-programmering

PLC programmeras med mjukvaran **STEP 7**. Programmeringsmjukvaran STEP 7 baserar på standardoperativsystemet **WINDOWS** och innehåller funktionerna för STEP 5 - programmeringen med innovativa vidareutvecklingar.

PLC-programminne

SINUMERIK 840D sl: I PLC-användarminnet lagras PLC-användarprogrammet och användardata gemensamt med PLC-grundprogrammet.

Polarkoordinater

Koordinatsystem som fastlägger en punkts läge i ett plan genom dess avstånd från nollpunkten och den vinkel som radievektorn bildar med en fastlagd axel.

Polynom-interpolering

Med polynom-interpolering kan de mest olika kurvförlopp skapas, som **linjär-, parabel-, potensfunktioner** (SINUMERIK 840D sl).

Positioneringsaxel

Axel som utför en hjälprörelse på en verktygsmaskin. (t.ex. verktygsmagasin, palettransport). Positioneringsaxlar är axlar som inte interpolerar med → banaxlarna.

Precisionsstopp

Vid programmerad precisionsstopp-anvisning uppsöks den i ett block angivna positionen exakt och ev. mycket långsamt. För att reducera tiden för närmandet definieras för snabbgång och matning → precisionsstoppgränser.

Precisionsstoppgräns

Uppnår alla banaxlar sin precisionsstoppgräns, så förhåller sig styrningen som om den har uppnått en målpunkt exakt. Det följer blockvidarekoppling i → detaljprogrammet.

Programblock

Programblock innehåller huvud- och underprogram till → detaljprogrammen.

Programmerbar arbetsfältsbegränsning

Begränsning av verktygets rörelsefrihet till ett av programmerade begränsningar definierat utrymme.

Programmerbara frames

Med programmerbara → frames kan dynamiskt under genomarbetningen av detaljprogrammet nya utgångspunkter för koordinatsystem definieras. Det skiljs mellan absolut fastläggande med hjälp av en ny frame och additivt fastläggande med referens till en bestående utgångspunkt.

Programmeringsnyckel

Tecken och teckenföljd som har en fastlagd betydelse i programmeringsspråket för → detaljprogrammen.

Programnivå

Ett i kanalen startat detaljprogram går som → huvudprogram på programnivå 0 (huvudprogramnivå). Varje i huvudprogrammet anropat detaljprogram går som → underprogram på en egen programnivå 1 ... n.

Referenspunkt

Punkt i verktygsmaskinen till vilken mätsystemet för → maskinaxlarna är relaterat.

Rotation

Komponent i en → frame som definierar en vridning av koordinatsystemet med en viss vinkel.

Roterande axel

Roterande axlar framkallar en arbetsstycks- eller verktygsvridning till ett föreskrivet vinkelläge.

R-parameter

Räkneparameter, kan sättas eller avfrågas av programmeraren av → detaljprogrammet för valfria ändamål i programmet.

Rundningsaxel

Rundningsaxlar framkallar en arbetsstycks- eller verktygsvridning till ett vinkelläge som motsvarar ett delningsraster. När rastret uppnås är rundningsaxeln "i position".

Råämne

Del med vilken bearbetningen av ett arbetsstycke börjar.

Settingdata

Data som meddelar NC-styrningen verktygsmaskinens egenskaper på ett av systemsoftware definierat sätt.

Sidoblock

Block som inleds med "N" med informationer för ett arbetssteg, t.ex. en positionsuppgift.

Skalning

Komponent i en → frame som förorsakar axelspecifika skaländringar.

Skruvlinje-interpolering

Skruvlinje-interpoleringen lämpar sig speciellt till enkel tillverkning av inner- eller yttergångar med formfräsar och till fräsning av smörjspår.

Därvid är skruvlinjen sammansatt av två rörelser:

- Cirkelrörelse i ett plan
- Linjär rörelse vinkelrätt mot detta plan

Skyddsutrymme

Tredimensionellt utrymme inom → arbetsutrymmet i vilket verktygsspetsen inte får räcka in.

Skärradiekompensering

Vid programmeringen av en kontur utgås från ett spetsigt verktyg. Då detta i praktiken inte är realiserbart, anges det använda verktygets böjningsradie till styrningen, som tar hänsyn till denna. Därvid förs böjningsmedelpunkten förflyttad med böjningsradien ekvidistant runt konturen.

Snabba digitala in-/utgångar

Via de digitala ingångarna kan t.ex. snabba CNC-programrutiner (interruptrutiner) startas. Via de digitala CNC-utgångarna kan snabba, programstyrda kopplingsfunktioner utlösas.

Snabbgång

Snabbaste förflyttningshastigheten för en axel. Den används t.ex. när verktyget från viloläge körs fram till → arbetsstyckskonturen eller dras tillbaka från arbetsstyckskonturen. Snabbgångshastigheten ställs in maskinspecifikt via maskindatum.

Snabblyftning från konturen

När en interrupt inträffar, kan en rörelse inledas via CNC-bearbetningsprogrammet, som möjliggör en snabb lyftning av verktyget från den arbetsstyckskontur som momentant bearbetas. Dessutom kan återgångsvinkeln och beloppet för sträckan parametreras. Efter snabblyftningen kan dessutom en interruptrutin genomföras.

Softkey

Tangent vars skrift representeras på ett fält i bildskärmen, som anpassar sig dynamiskt till den aktuella manöversituationen. De fritt beläggningsbara funktionstangenterna (Softkeys) tillordnas softwaremässigt definierade funktioner.

Spegling

Vid spegling byts förtecknen för koordinatvärdena till en kontur vad beträffar en axel. Det kan speglas i flera axlar samtidigt.

Spelkompensering

Utgjämning av ett mekaniskt maskinspel, t.ex. vändspel vid kulmutterskruv. För varje axel kan spelkompenseringen matas in separat.

Spindelstigningsfel-kompensering

Utgjämning av bristande mekanisk precision för en i matningen deltagande kulmutterskruv genom styrningen med hjälp av lagrade mätvärden för avvikelserna.

Spline-interpolering

Med spline-interpoleringen kan styrningen ur några få föreskrivna stödpunkter från en börkontur skapa ett jämnt kurvförlopp.

Standardcykler

För ofta återkommande bearbetningsuppgifter står standardcykler till förfogande:

- För teknologin Borra/Fräsa
- För teknologin Svarva

I manöverområdet "Program" finns under menyn "Cykelstöd" de cykler som står till förfogande i en lista. Efter val av den önskade bearbetningscykeln visas de nödvändiga parametrarna för värdetilldelningen i klartext.

Standardradering

Vid standardradering raderas följande minnen i → CPU:

- → arbetsminne
- skriv-/läsområdet i → laddningsminnet
- → systemminne
- → backup-minne

Stegmått

Förflyttningssträcka-längduppgift via antal inkrement (stegmått). Antal inkrement kan vara lagrat som → settingdatum resp. väljas av tangenter med motsvarande skrift 10, 100, 1000, 10000.

Styraxel

Styraxeln är → gantry-axeln, som ur operatörens och programmerarens synvinkel finns och därmed kan påverkas som en normal NC-axel.

Synkronaktioner

1. Hjälpfunktionsutmatning
Under arbetsstycksbearbetningen kan utifrån CNC-programmet teknologiska funktioner (→ hjälpfunktioner) matas ut till PLC. Via dessa hjälpfunktioner styrs till exempel extraanordningar i verktygsmaskinen som pinoler, gripare, chuck etc.
2. Snabb hjälpfunktionsutmatning
För tidskritiska kopplingsfunktioner kan kvitteringstiderna för → hjälpfunktionerna minimeras och onödiga hållningspunkter i bearbetningsprocessen undvikas.

Synkronaxlar

Synkronaxlar behöver för sin väg samma tid som geometriaxlarna för sin banväg.

Synkronisering

Anvisningar i → detaljprogrammen för koordinering av förloppen i olika → kanaler till vissa bearbetningsställen.

Systemminne

Systemminnet är ett minne i CPU, i vilket följande data lagras:

- Data som operativsystemet behöver
- Operanderna tider, räknare, flaggor

Systemvariabel

En i ett → detaljprogram existerande variabel utan medverkan av programmeraren. Den är definierad genom en datatyp och variabelnamnet som inleds med tecknet \$. Se → användardefinierad variabel.

Säkerhetsfunktioner

Styrningen innehåller ständigt aktiva övervakningar, som identifierar störningar i → CNC, anpassningsstyrningen (→ PLC) och maskinen så tidigt att skador på arbetsstycke, verktyg eller maskin i stor utsträckning kan uteslutas. I händelse av störning avbryts bearbetningsförloppet och drivningarna stoppas, störningsorsaken sparas och larm indikeras. Samtidigt meddelas PLC att ett CNC-larm väntar.

Text-editor

Se → editor

Tidsreciprok matning

I stället för matningshastigheten kan för axelrörelsen också den tid programmeras, som ett blocks banväg ska behöva (G93).

TOA-enhet

Varje → TOA-område kan innehålla flera TOA-enheter. Antalet möjliga TOA-enheter begränsas av det maximala antalet aktiva → kanaler. En TOA-enhet omfattar exakt ett WZ-data-block och ett magasin-data-block. Dessutom kan också ett WZ-bärardata-block ingå (option).

TOA-område

TOA-området omfattar alla verktygs- och magasin-data. Standardmässigt sammanfaller området beträffande räckvidden på data med → kanal-området. Via maskindata kan dock fastläggas att flera kanaler delar på en → TOA-enhet, så att sedan gemensamma WZV-data står till förfogande för dessa kanaler.

Transformation

Additiv eller absolut nollpunktsförflyttning för en axel.

Tum-måttsystem

Måttsystem som definierar avstånd i "inch" och bråkdelar därav.

Underprogram

Beteckningen "underprogram" stammar ännu från den tiden när detaljprogram var fast indelade i → huvud- och underprogram. Denna fasta indelning finns inte längre med det nutida SINUMERIK NC-språket. Principiellt kan varje detaljprogram eller varje → cykel anropas inom ett annat detaljprogram än underprogram. Det går då i nästa → programnivå (x+1) (underprogramnivå (x+1)).

V.24

Seriellt gränssnitt för in-/utmatning av data. Via detta gränssnitt kan bearbetningsprogram samt tillverkar- och användardata laddas och lagras.

Variabeldefinition

En variabeldefinition omfattar fastläggande av en datatyp och ett variabelnamn. Med variabelnamnet kan variabelns värde utlösas.

Verktyg

På verktygsmaskinen verksam del, som förorsakar bearbetningen (t.ex. svarvstål, fräs, borr, LASER-stråle ...).

Verktygskompensering

Hänsynstagande till verktygets dimensioner vid beräkningen av banan.

Verktygsradiekompensering

För att direkt kunna programmera en önskad → arbetsstyckskontur, måste styrningen under hänsynstagande till radien på det använda verktyget köra en ekvidistant bana till den programmerade konturen (G41/G42).

WinSCP

WinSCP är ett fritt tillgängligt Open Source-program för Windows för överföring av filer.

Övermannig

Manuell resp. programmerbar möjlighet att ingripa som tillåter operatören att överlagra programmerade matningar eller varvtal för att anpassa dem till ett visst arbetsstycke eller material.

Index

\$

\$AA_ATOL, 540
\$AA_COUP_ACT
 vid axial styrvärdeskoppling, 568
 vid medsläpning, 546
\$AA_ESR_ENABLE, 659
\$AA_LEAD_SP, 568
\$AA_LEAD_SV, 568
\$AC_ACT_PROG_NET_TIME, 648
\$AC_ACTUAL_PARTS, 651
\$AC_AXCTSWA, 634
\$AC_AXCTSWE, 634
\$AC_CTOL, 540
\$AC_CUT_INV, 465
\$AC_CUTMOD, 465
\$AC_CUTMOD_ANG, 465
\$AC_CUTMODK, 465
\$AC_CUTTING_TIME, 647
\$AC_CYCLE_TIME, 647
\$AC_DELAYFST, 522
\$AC_ESR_TRIGGER, 659
\$AC_OLD_PROG_NET_TIME, 648
\$AC_OLD_PROG_NET_TIME_COUNT, 648
\$AC_OPERATING_TIME, 647
\$AC_OTOL, 540
\$AC_PROG_NET_TIME_TRIGGER, 648
\$AC_REPOS_PATH_MODE, 531
\$AC_REQUIRED_PARTS, 650
\$AC_SMAXVELO, 536
\$AC_SMAXVELO_INFO, 536
\$AC_SPECIAL_PARTS, 651
\$AC_TOTAL_PARTS, 650
\$AC_TRAFO_CORR_ELEM_P, 399
\$AC_TRAFO_CORR_ELEM_T, 399
\$AC_TRAFO_ORIAX_LOC, 399
\$AN_AXCTAS, 634
\$AN_AXCTSWA, 634
\$AN_ESR_TRIGGER, 659
\$AN_LANGUAGE_ON_HMI, 908
\$AN_POWERON_TIME, 647
\$AN_SETUP_TIME, 647
\$NT_CLOSE_CHAIN_T, 399
\$NT_CNTRL, 399
\$NT_CORR_ELEM_P, 398
\$NT_CORR_ELEM_T, 398
\$NT_NAME, 391
\$NT_ROT_AX_NAME, 462

\$NT_TRAFO_INDEX, 391
\$P_ACTBFRAME, 309
\$P_AD, 464
\$P_BFRAME, 309
\$P_CHBFRAME, 309
\$P_CHBFRMASK, 310
\$P_CTOL, 541
\$P_CUT_INV, 465
\$P_CUTMOD, 465
\$P_CUTMOD_ANG, 465
\$P_CUTMOD_ERR, 466
\$P_CUTMODK, 465
\$P_DELAYFST, 522
\$P_IFRAME, 310
\$P_IS_EES_PATH, 219
\$P_NCBFRAME, 309
\$P_NCBFRMASK, 310
\$P_ORI_DIFF, 457
\$P_ORI_POS, 457
\$P_ORI_SOL, 458
\$P_ORI_STAT, 461
\$P_OTOL, 541
\$P_PATH, 218
\$P_PFRAME, 311
\$P_PROG, 218
\$P_PROGPATH, 218
\$P_SIM, 277
\$P_STACK, 218
\$P_SUBPAR, 159
\$P_TOOLENV, 474
\$P_TOOLENVN, 474
\$PA_ATOL, 541
\$SA_LEAD_TYPE, 568
\$SC_CONTPREC, 515
\$SC_MINFEED, 515
\$SC_PA_ACTIV_IMMED, 230
\$SN_PA_ACTIV_IMMED, 230
\$TC_CARR1...14, 442
\$TC_CARR18...23, 442
\$TC_CARR18[m], 446
\$TC_DP1 ... 25, 401
\$TC_ECPxy, 405
\$TC_SCPxy, 405

*

* (räknefunktion), 72

/

/ (räknefunktion), 72

+

+ (räknefunktion), 72

<

< (jämförelseoperator), 75

<< (länkningsoperator), 85

<= (jämförelseoperator), 75

<> (jämförelseoperator), 75

=

== (jämförelseoperator), 75

>

> (jämförelseoperator), 75

>= (jämförelseoperator), 75

0

0-tecken, 83

A

ABS, 72

Accelerationsmode, 508

ACCLIMA, 510

ACOS, 72

ACTBLOCNO, 171

ACTFRAME, 285

ADISPOSA, 279

Adressering, 213

ALF

för snabblyftning från konturen, 131

AND, 75

Användar-XML, 642

APR, 44

APRB, 44

APRP, 44

APW, 44

APWB, 44

APWP, 44

Arbetsminne, 220

Arbetsstycke

-huvudkatalog, 210

-kataloger, 210

-räknare, 650

Arbetsstycks-följd, 397

Array, 51

AS, 206

ASIN, 72

ASPLINE, 242

A-spline, 248

Asynkron pendling, 599

ATAN2, 72

ATOL, 538

Automatisk stoppvisare, 524

Automatisk väguppdelning, 616

AV, 575

Avrundning uppåt, 153

Avspåna - CYCLE951

programmera externt, 785

Avspåning

stödjande funktioner, 673

Avstick - CYCLE92

programmera externt, 743

AX, 625

AXCTSWE, 633

AXCTSWEC, 633

AXCTSWED, 633

Axel

-byte, 135

Axial styrvärdeskoppling, 564

AXIS, 28

Axlar

Medsläpnings-, 545

AXNAME, 84

AXSTRING, 625

AXTOCHAN, 140

AXTOSPI, 625

återdragning

drivningsautark, 664

NC-styrd,

Ändlös slinga, 115

Ärvärdeskoppling, 578

B

B_AND, 75

B_NOT, 75

B_OR, 75

B_XOR, 75

Banfräsa - CYCLE72

programmera externt, 719

Banrelativ verktygsorientering, 348
 BAUTO, 242
 Bearbetningstid, 647
 BFRAME, 285
 BLOCK, 194
 Blockindikering
 undertrycka, 171
 BLSYNC, 126
 BNAT, 242
 BOOL, 28
 Borra - CYCLE82
 programmera externt, 732
 BOUND, 79
 BRISK, 508
 BRISKA, 508
 Brotscha - CYCLE85
 programmera externt, 741
 BSPLINE, 242
 B-spline, 249
 BTAN, 242
 Början programblock - GROUP_BEGIN, 813
 Börvärdeskoppling, 578

C

CAC, 241
 CACN, 241
 CACP, 241
 CALL, 193
 CALLPATH, 197
 CASE, 104
 case-insensitiv, 212
 CDC, 241
 Centra - CYCLE81
 programmera externt, 731
 CFINE, 295
 CHAN, 28
 CHANDATA, 220
 CHAR, 28
 CHKDNO, 439
 CIC, 241
 Cirkeldata
 beräkna, 687
 Cirkelficka - POCKET4
 programmera externt, 699
 Cirkelspår - SLOT2
 programmera externt, 704
 Cirkeltapp - CYCLE77
 programmera externt, 725
 CLEARM, 119
 CLRINT, 128
 COARSE, 575

COARSEA, 279
 COLPAIR, 386
 COMPCAD, 255
 COMPCURV, 255
 COMPLETE, 220
 COMPOF, 255
 COMPON, 255
 COMPSURF, 255
 CONTDCON, 680
 CONTPRON, 674
 CORRTRAFO, 392
 COS, 72
 COUPDEF, 575
 COUPDEL, 575
 COUPOF, 575
 COUPOFS, 575
 COUPON, 575
 COUPONC, 575
 COUPRES, 575
 CP, 367
 CPBC, 587
 CPDEF, 586
 CPDEL, 586
 CPFMOF, 589
 CPFMON, 589
 CPFMON, 588
 CPFPOS + CPOF, 589
 CPFPOS + CPON, 587
 CPFRS, 587
 CPLA, 586
 CPLCTID, 587
 CPLDEF, 586
 CPLDEL, 586
 CPLDEN, 587
 CPLINSC, 591
 CPLINTR, 591
 CPLNUM, 587
 CPLOF, 587
 CPLON, 586
 CPLOUTSC, 591
 CPLOUTTR, 591
 CPLPOS, 587
 CPLSETVAL, 587
 CPMALARM, 592
 CPMBRAKE, 592
 CPMPT, 591
 CPMRESET, 590
 CPMSTART, 591
 CPMVDI, 592
 CPOF, 586
 CPON, 586
 CPRECOF, 515

CPRECON, 515
CPROT, 227
CPROTDEF, 223
CPSETTYPE, 592
CPSYNCOV, 591
CPSYNCOV2, 591
CPSYNCOV, 591
CPSYNFIP, 591
CPSYNFIP2, 591
CPSYNFIV, 591
CSPLINE, 242
C-spline, 250
CTAB, 557
CTABDEF, 547
CTABDEL, 554
CTABEND, 547
CTABEXISTS, 553
CTABFNO, 562
CTABFPOL, 562
CTABFSEG, 562
CTABID, 556
CTABINV, 557
CTABISLOCK, 556
CTABLOCK, 555
CTABMEMTYP, 556
CTABMPOL, 562
CTABMSEG, 562
CTABNO, 562
CTABNOMEM, 562
CTABPERIOD, 556
CTABPOL, 562
CTABPOLID, 562
CTABSEG, 562
CTABSEGID, 562
CTABSEV, 557
CTABSSV, 557
CTABTEP, 557
CTABTEV, 557
CTABTMAX, 557
CTABTMIN, 557
CTABTSP, 557
CTABTSV, 557
CTABUNLOCK, 555
CTOL, 538
CTRANS, 295
CUT3DC, 419
CUT3DCC, 429
CUT3DCCD, 429
CUT3DCD, 419
CUT3DF, 423
CUT3DFD, 423
CUT3DFF, 423
CUT3DFS, 423
CUTMOD, 462
CUTMODK, 462
CYCLE4071
 programmera externt, 794
CYCLE4072
 programmera externt, 795
CYCLE4073
 programmera externt, 799
CYCLE4074
 programmera externt, 800
CYCLE4075
 programmera externt, 803
CYCLE4077
 programmera externt, 806
CYCLE4078
 programmera externt, 809
CYCLE4079
 programmera externt, 811
CYCLE435 - ställa in koordinatsystem för
skärpverktyg
 programmera externt, 756
CYCLE495 - profilera
 programmera externt, 756
CYCLE60 - gravyr
 programmera externt, 708
CYCLE61- planfräsa
 programmera externt, 711
CYCLE62 - konturupprop
 programmera externt, 713
CYCLE63 - fräsa konturficka
 programmera externt, 714
CYCLE64 - förborra konturficka
 programmera externt, 716
CYCLE70 - gängfräsning
 programmera externt, 717
CYCLE72 - banfräsa
 programmera externt, 719
CYCLE76 - fyrkanttapp
 programmera externt, 723
CYCLE77 - cirkeltapp
 programmera externt, 725
CYCLE78 - fräsa hålgänga
 programmera externt, 727
CYCLE79 - flerkant
 programmera externt, 729
CYCLE800 - vridning
 programmera externt, 758
CYCLE801 - positionsmönster gitter/ram
 programmera externt, 761
CYCLE802 - valfria positioner
 programmera externt, 763

CYCLE81 - centrera
 programmera externt, 731
 CYCLE82 - borra
 programmera externt, 732
 CYCLE83 - djuphålsborra
 programmera externt, 735
 CYCLE830 - Djuphålsborra 2
 programmera externt, 765
 CYCLE832 - High Speed Settings
 programmera externt, 771
 CYCLE84 - gängtappning utan flytande gänghållare
 programmera externt, 738
 CYCLE840 - gängtappning med flytande gänghållare
 programmera externt, 774
 CYCLE85 - brotscha
 programmera externt, 741
 CYCLE86 - ursvarvning
 programmera externt, 742
 CYCLE899 - fräsa öppet spår
 programmera externt, 777
 CYCLE92 - avstick
 programmera externt, 743
 CYCLE930 - instick
 programmera externt, 780
 CYCLE940 - fristick
 programmera externt, 783
 CYCLE95 - konturavspåna
 programmera externt, 745
 CYCLE951- avspåna
 programmera externt, 785
 CYCLE952 - kontursticka
 programmera externt, 788
 CYCLE98 - kopplade gängor
 programmera externt, 747
 CYCLE99 - gängsvarvning
 programmera externt, 751
 Cykellarm, 657
 Cylindermanteltransformation, 317

D

Dataklass, 49
 DCI, 49
 DCM, 49
 DCU, 49
 DEF, 28
 DEFAULT, 104
 DEFINE ... AS, 206
 DELAYFSTOF, 520
 DELAYFSTON, 520
 DELDL, 406
 DELETE, 147

DELOBJ, 381
 DELTOOLENV, 472
 DIN underprogramnamn, 217
 DISABLE, 127
 DISPLOF, 171
 DISPLON, 171
 Disponibilitet
 System-beroende, 5
 DISPR, 525
 DIV, 72
 Djuphålsborra - CYCLE83
 programmera externt, 735
 Djuphålsborra 2 - CYCLE830
 programmera externt, 765
 DL, 404
 D-nummer
 byta namn, 440
 fritt fördela, 439
 kontrollera, 439
 DO, 597
 Driftmode
 vid mätning, 272
 DRIVE, 508
 DRIVEA, 508
 DV, 575
 DYNFINISH, 512
 DYNNORM, 512
 DYNPOS, 512
 DYNROUGH, 512
 DYNSEMIFIN, 512

E

Easy XML, 642
 EAUTO, 242
 EES, 211
 EES-Notation, 213
 EG
 Elektronisk växel, 569
 EGDEF, 569
 EGDEL, 574
 EGOFC, 573
 EGOFs, 573
 EGON, 570
 EGONSYN, 570
 EGONSYNE, 570
 Elektronisk växel, 569
 ELSE, 113
 ENABLE, 127
 ENAT, 242
 ENDFOR, 115
 ENDIF, 113

ENDLABEL, 106
ENDLOOP, 115
ENDWHILE, 117
Enhetsnamn, 214
Enkelblock
 -undertryckning, 166
ESR, 658
ESRR, 664
ESRS, 663
ETAN, 242
Eulervinkel, 329
EVERY, 597
EXECSTRING, 71
EXECTAB, 686
EXECUTE, 689
EXP, 72
EXTCALL
 för SINUMERIK 828D, 201
 för SINUMERIK 840D sl, 198
EXTCLOSE, 652
EXTERN, 188
Extern nollpunktsförläggning, 297
EXTOPEN, 652

F

FALSE, 28
FCTDEF, 415
FCUB, 503
FENDNORM, 278
FFWOF, 514
FFWON, 514
FIFOCTRL, 517
Fil
 -informationer, 151
FILEDATE, 151
FILEINFO, 151
FILESIZE, 151
FILESTAT, 151
FILETIME, 151
Filnamn, 216
FINE, 575
FINEA, 279
Flerkant - CYCLE79
 programmera externt, 729
FLIN, 503
FNORM, 503
FOR, 115
FPO, 503
FRAME, 28
 anropa,
 -länkning,

Framekomponent
 FI, 291
 MI, 291
 RT, 291
 SC, 291
 TR, 291
frames
 Framelänkar,
 Kanalspecifika, 308
 NCU-globala, 307
 System-, 308
 tillordna,
Framevariabel
 Anrop av koordinattransformationer, 283
 Fördefinierad framevariabel, 285
 Tillordna värden, 289
Fristick - CYCLE940
 programmera externt, 783
FROM, 597
Fräsa hålgänga - CYCLE78
 programmera externt, 727
Fräsa konturficka - CYCLE63
 programmera externt, 714
Fräsa öppet spår - CYCLE899
 programmera externt, 777
Fräsbearbetningspunkt, 422
Fräshjälpunkt, 422
Frässpets, 422
FTOCOF, 418
FTOCON, 418
Fyrkantficka - POCKET3
 programmera externt, 696
Fyrkanttapp - CYCLE76
 programmera externt, 723
Fält
 -definition, 51
 -element, 51
Fältindex, 53
Följdaxel
 vid axial styrvärdeskoppling, 564
Förborra konturficka - CYCLE64
 programmera externt, 716
Förkörning
 -minne, 517
Förslitningsvärde, 405

G

G290, 670
G291, 670
G5, 363
G62, 278

G621, 278
 G7, 363
 G810 ... G819, 277
 G820 ... G829, 277
 GEOAX, 628
 geometriaxel
 koppla om, 628
 GET, 135
 GETACTTD, 441
 GETD, 135
 GETDNO, 440
 GETTCOR, 474
 GETTENV, 473
 GETVARAP, 59
 GETVARDFT, 62
 GETVARDIM, 61
 GETVARLIM, 60
 GETVARPHU, 59
 GETVARTYP, 63
 G-grupp
 Teknologi, 512
 G-kod
 indirekt programmera, 67
 Globalt detaljprogramminne (GDIR), 211
 GOTO, 101
 GOTOB, 101
 GOTOC, 101
 GOTOF, 101
 GOTOS, 100
 GP, 68
 Gravyr - CYCLE60
 programmera externt, 708
 GROUP_ADDEND - slut inkörningstillägg
 programmera externt, 814
 GROUP_BEGIN - början programblock
 programmera externt, 813
 GROUP_END - slut programblock
 programmera externt, 814
 Grundläge för verktygsorienteringen
 ORIRESET, 326
 GUD, 29
 Gängfräsning - CYCLE70
 programmera externt, 717
 Gängsvarvning - CYCLE99
 programmera externt, 751
 Gängtappning med flytande gänghållare - CYCLE840
 programmera externt, 774
 Gängtappning utan flytande gänghållare - CYCLE84
 programmera externt, 738

H

Hastighetskoppling, 578
 High Speed Settings - CYCLE832
 programmera externt, 771
 HOLES1 - positionsmönster linje
 programmera externt, 694
 HOLES2 - positionsmönster cirkel
 programmera externt, 694
 Hopp
 till hoppmärken, 101
 till programbörjan, 100
 Hoppmärke
 vid programdelsupprepningar, 106
 vid programhopp, 101
 Hörfördröjning vid alla hörn, 278
 Hörfördröjning vid innerhörn, 278

I

ID, 597
 IDS, 597
 IF, 113
 IFRAME, 285
 INDEX, 88
 Indirekt programmering
 av adresser, 65
 av detaljprogramrader, 71
 av G-koder, 67
 av positionsattribut, 68
 INICF, 28
 INIPO, 28
 INIRE, 28
 INIT, 119
 INITIAL, 220
 INITIAL_INI, 220
 Initiering
 av fält, 51
 Initieringsprogram, 220
 Instick - CYCLE930
 programmera externt, 780
 Inställningsvärde, 405
 INT, 28
 Interpolering av vridvektorn, 345
 Interruptrutin
 Programmerbar förflyttningsriktning, 131
 Radera, 128
 Snabbflytning från konturen, 129
 till-/frånkoppling, 127

- Tillordna på nytt, 127
- Återgångsrörelse, 131
- INTERSEC, 684
- IPOBRKA, 279
- IPOENDA, 279
- IPOSTOP, 575
- IPTRLOCK, 523
- IPTRUNLOCK, 523
- ISAXIS, 625
- ISFILE, 150
- ISNUMBER, 84
- ISOCALL, 195
- ISVAR, 57

J

- Jerk
 - begränsning, 508
 - kompensering, 534
- JERKLIM, 534
- JERKLIMA, 510
- Jämförelseoperatorer, 75
- Jämning
 - för orienteringsförloppet, 357

K

- Kapslingsdjup
 - av kontrollstrukturer, 112
- Kartesisk PTP-körning, 367
- Kinematik
 - Upplöst, 446
- Kinematiktyp, 446
- Kompenseringsminne, 401
- Kontroll
 - strukturer, 112
- Kontur
 - förberedelse, 674
 - kodning, 680
 - tabell, 674
 - åter uppsöka, 525
- Konturavspåna - CYCLE95
 - programmera externt, 745
- Konturelement
 - bortkörning, 686
- Konturförberedelse
 - Felsvar, 689
- Konturprecision
 - programmerbar, 515
- Kontursticka - CYCLE952
 - programmera externt, 788

- Konturupprop - CYCLE62
 - programmera externt, 713
- Kopplade gångor - CYCLE98
 - programmera externt, 747
- Koppling
 - Generisk, 586
- Kopplingsfaktor, 543
- Kopplingsstatus
 - vid axial styrvärdeskoppling, 568
 - vid medsläpning, 546
- Kopplingstyp, 578
- Körtid
 - beteende för kontrollstrukturer, 112

L

- L..., 186
- Label, 106
- Larm
 - sätta i NC-programmet, 657
- LEAD, 327
- LEADOF, 564
- LEADON, 564
- LENTOAX, 493
- LIFTFAST, 129
- Link
 - variabler, 27
- LLI, 40
- LN, 72
- Logiska operatorer, 75
- LONGHOLE - långhål
 - programmera externt, 706
- LOOP, 115
- LUD, 29
- Långhål - LONGHOLE
 - programmera externt, 706
- Lägessynkronitet, 575
- Lägesynkronitet med vinkelförskjutning, 575
- Längsspår - SLOT1
 - programmera externt, 701
- Länkning
 - av strings, 85

M

- M17, 174
- M30, 174
- Makro, 206
- MASLDEF, 593
- MASLDEL, 593
- MASLOF, 593

MASLOFS, 593
 MASLON, 593
 MATCH, 88
 MAXVAL, 79
 MCALL, 191
 MD10010, 119
 MD10280, 119
 MD15800, 26
 MD18104, 471
 MD18116, 472
 MD18156, 26
 MD20360, 478
 MD24558, 479
 MD24658, 479
 MEAC, 267
 MEAFRAME, 303
 MEAS, 264
 MEASA, 267
 MEAW, 264
 MEAWA, 267
 Medsläpning, 543
 Medsläpningsaxlar, 545
 Medsläpningsförband, 543
 MINDEX, 88
 Minne
 Arbets-, 220
 Förberäknings-, 517
 Program-, 209
 MINVAL, 79
 MMC, 642
 MOD, 72
 MODAXVAL, 625
 MPF, 210
 Mätuppdagsstatus, 275

N

NAMETOINT, 384
 NCK, 28
 NCK-Notation, 213
 Nermatningsdjup, 421
 NEWCONF, 142
 Nibbling, 611
 NOC, 575
 Nollpunktsförflyttning
 Extern, 297
 NOT, 75
 NPROT, 227
 NPROTDEF, 223
 NUMBER, 84
 NUT, 338
 Nämnar-polynom, 260

O

OEM-adresser, 277
 OEM-funktioner, 277
 OEMIPO1/2, 277
 OMA1 ... OMA5, 277
 Omkopplingsbara geometriaxlar, 628
 Online-verktygslängdkompensering, 450
 OR, 75
 ORIXES, 336
 ORIC, 433
 ORICONCCW, 338
 ORICONCW, 338
 ORICONIO, 338
 ORICONTO, 338
 ORICURVE, 341
 ORID, 433
 Orienterbara verktygsbärare, 442
 Orienteringsaxlar, 336
 Orienteringsprogrammering, 336
 Orienteringstransformation TRAORI
 Generisk 5/6-axel transformation, 316
 Maskinkinematik, 316
 Orienteringsprogrammering, 325
 Orienteringsrörelser, 315
 Varianter för orienteringsprogrammering, 325
 Orienteringsvektor THETA, 345
 ORIEULER, 336
 ORIMKS, 334
 ORIPATH, 349
 ORIPATHS, 349
 ORIPLANE, 338
 ORIRESET(A, B, C), 325
 ORIROTA, 345
 ORIROTC
 vid interpolering av verktygsvridningen, 351
 vid rotation av verktygsorienteringen, 345
 ORIROTR, 345
 ORIROTT, 345
 ORIRPY, 336
 ORIRPY2, 336
 ORIS, 433
 ORISOF, 357
 ORISOLH, 453
 ORISON, 357
 ORIVECT, 336
 ORIVIRT1, 336
 ORIVIRT2, 336
 ORIWKS, 334
 OS, 599
 OSB, 599

OSC, 433
OSCILL, 604
OSCTRL, 599
OSD, 433
OSE, 599
OSNSC, 599
OSOF, 433
OSP1, 599
OSP2, 599
OSS, 433
OSSE, 433
OST, 433
OST1, 599
OST2, 599
OTOL, 538

P

P..., 190
P_ACTFRAME, 311
Parameter
 Aktual-, 157
 Formal-, 157
 -överföring vid underprogramanrop, 158
Parametrar
 Verktys-, 401
 -överföring vid underprogramanrop, 188
PCALL, 196
PDELAYOF, 611
PDELAYON, 611
Pendelrörelse
 Ansättning i vändpunkt, 608
 Vändområde, 606
 Vändpunkt, 606
Pendling
 Asynkron, 599
 Asynkron pendling, 599
 Delansättning, 606
 styra via synkronaktion, 604
 Synkron pendling, 604
PFRAME, 285
PHI
 Orienteringspolynom, 343
 vid orientering längs en konisk mantelyta, 338
PHU, 41
PL
 vid polynom-interpolering, 256
 vid spline-interpolering, 242
Planfräsa, 333
Planfräsa - CYCLE61
 programmera externt, 711
PO, 256
PO[PHI]
 Orienteringspolynom, 343
 vid orientering längs en konisk mantelyta, 338
 vid rotation av verktygsorienteringen, 349
PO[PSI]
 Orienteringspolynom, 343
 vid orientering längs en konisk mantelyta, 338
 vid rotation av verktygsorienteringen, 349
PO[THT]
 Orienteringspolynom, 343
 vid rotation av verktygsorienteringen, 349
PO[XH]
 Orienteringspolynom, 344
 vid orienteringsföreskrift för två
 kontaktpunkter, 341
PO[YH]
 Orienteringspolynom, 344
 vid orienteringsföreskrift för två
 kontaktpunkter, 341
PO[ZH]
 Orienteringspolynom, 344
 vid orienteringsföreskrift för två
 kontaktpunkter, 341
POCKET3 - fyrkantficka
 programmera externt, 696
POCKET4 - cirkelficka
 programmera externt, 699
Polar-transformation, 317
POLF
 för NC-styrd återdragning, 659
POLFA, 659
POLFMASK
 för NC-styrd återdragning, 659
POLFMLIN
 för NC-styrd återdragning, 659
POLY, 256
Polynom-interpolering, 256
Polynomkoefficient, 257
POLYPATH, 256
PON, 619
PONS, 611
POSFS, 575
Positionsattribut
 indirekt programmera, 68
Positionsmönster cirkel - HOLES2
 programmera externt, 694
Positionsmönster gitter/ram - CYCLE801
 programmera externt, 761
Positionsmönster linje - HOLES1
 programmera externt, 694
POT, 72
PREPRO, 174

PRESETON, 299
 PRESETONS, 301
 PRIO, 126
 PRLOC, 28
 Process DataShare, 652
 Profilera - CYCLE495
 programmera externt, 756
 Program
 -adressering, 213
 -förgrening, 104
 -hopp, 101
 Initierings-, 220
 -körningstider, 647
 -minne, 210
 -upprepning, 190
 Programdel
 -upprepning, 106
 Programdelsupprepning
 med indirekt programmering CALL, 194
 programmera externt, 813
 Programminne
 Filtyper, 210
 Standard-kataloger, 210
 Programslinga
 IF-slinga, 113
 REPEAT-slinga, 117
 Räkneslinga, 115
 Slutslinga, 115
 WHILE-slinga, 117
 PROTA, 387
 PROTD, 389
 PROTS, 388
 PSI
 Orienteringspolynom, 343
 vid orientering längs en konisk mantelyta, 338
 PTP, 367
 PTPG0, 367
 PTPWOC, 367
 PUD, 29
 PUNCHACC, 611
 Punkt-till-punkt-körning, 367
 PUTFTOC, 417
 PUTFTOCF, 416
 PW, 242

R

Randvillkor vid transformationer, 378
 READ, 148
 REAL, 28
 REDEF, 34
 Registrera och söka sökodugliga områden, 524

RELEASE, 135
 REP, 51
 REPEAT, 106
 REPEATB, 106
 REPOSA, 525
 REPOSH, 525
 REPOSHA, 525
 REPOSL, 525
 REPOSQ, 525
 REPOSQA, 525
 Resttid
 för ett arbetsstycke, 649
 Restvägsradering, 273
 RET, 175
 RET (parametrerbar), 176
 RETB (parametrerbar), 182
 RG, 25
 Riktningsvektor, 330
 RINDEX, 88
 RMBBL, 525
 RMEBL, 525
 RMIBL, 525
 RMNBL, 525
 ROUND, 72
 ROUNDUP, 153
 RPY-vinkel, 330
 Run MyScreens, 642
 Råämnesdefinition, 666
 Räkneparameter
 Global, 25
 Kanalspecifika, 24
 Räkneslinga, 115
 Rörelseslutkriterium
 programmerbar, 279

S

SAVE, 164
 SBLOF, 166
 SBLON, 166
 SCPARA, 636
 SD, 242
 SD41610, 399
 SD41611, 399
 SD42475, 355
 SD42476, 355
 SD42477, 355
 SD42900, 409
 SD42910, 409
 SD42920, 410
 SD42930, 411
 SD42935, 412

- SD42940, 413
- SD42984, 464
- SET, 51
- SETAL, 657
- SETDNO, 440
- SETINT, 126
- SETM, 119
- SETTCOR, 480
- SIN, 72
- Singulära ställen, 335
- Skyddsområden, 223
- Skärnummer, 439
- Slipning med sned instickning, 363
- SLOT1 - längsspår
 - programmera externt, 701
- SLOT2 - cirkelspår
 - programmera externt, 704
- Slut inkörningstillägg - GROUP_ADDEND
 - programmera externt, 814
- Slut programblock - GROUP_END
 - programmera externt, 814
- Snabblyftning från konturen, 129
- Snedvinkeltransformation (TRAANG)
 - med programmerbar vinkel, 362
- SOFT, 508
- SOFTA, 508
- SON, 611
- SONS, 611
- SPF, 210
- SPI, 625
- SPIF1, 611
- SPIF2, 611
- Spindel
 - byte, 135
- Spline
 - interpolering, 242
 - typer, 248
- Spline-förband, 253
- SPLINEPATH, 253
- SPN, 616
- SPOF, 611
- SPP, 616
- SPRINT, 91
- Språkmode, 670
- SQRT, 72
- Stansning, 611
- START, 119
- STARTFIFO, 517
- STAT, 368
- STOPFIFO, 517
- Stoppande
 - drivningsautark, 663
 - NC-styrd, 662
- Stoppblock, 524
- STOPRE, 517
- String,
 - formatera, 91
 - längd, 87
 - länkning, 85
 - operationer, 83
- STRINGIS, 638
- STRLEN, 87
- Styraxel
 - vid axial styrvärdeskoppling, 564
- Styrvärdeskoppling
 - Synkronisering styr- och följdaxel, 566
 - Är- och börvärdeskoppling, 567
- Styrvärdessimulering, 568
- Ställa in skärpverktygskoordinater - CYCLE435
 - programmera externt, 756
- SUBSTR, 89
- Synkron pendling
 - Ansättning i vändområde, 608
 - Ansättningsrörelse, 608
 - Fastlägga ansättningar, 607
 - Nästa delansättning, 610
 - Synkronaktioner, 607
 - Tillordning av pendel- och ansättningsaxel, 607
 - Utvärdering IPO-takt, 609
- Synkronkörning
 - fin, 578
 - gorv, 578
- Synkronspindel
 - Koppling, 575
 - par fastlägga, 580
- SYNR, 28
- SYNRW, 28
- SYNW, 28
- System
 - beroende disponibilitet, 5
- Systemframes, 308
- Systemvariabler
 - Mätprobebegränsning, 275
 - Mätprobestatus, 274
- Sökväg
 - Programmerbar sökväg, 197
 - vid underprogramanrop, 217
- Sökväg för katalog, 215
- Sökvägsuppgift, 214

T

TAN, 72
 TANG, 497
 TANGDEL, 501
 TANGOF, 501
 TANGON, 499
 TCARR, 447
 TCOABS, 447
 TCOFR, 447
 TCOFRX, 447
 TCOFRY, 447
 TCOFRZ, 447
 THETA
 vid interpolering av verktygsvridningen, 351
 vid rotation av verktygsorienteringen, 345
 TILT, 327
 TLIFT, 498
 TMOF, 623
 TMON, 623
 TOFFOF, 450
 TOFFON, 450
 TOWER, 86
 TOOLENV, 469
 TOUPPER, 86
 TOWBCS, 411
 TOWKCS, 411
 TOWMCS, 411
 TOWSTD, 411
 TOWTCS, 411
 TOWWCS, 411
 TRAANG
 med programmerbar vinkel, 362
 TRACON, 365
 TRACYL, 359
 TRAFOOF, 379
 TRAFOON, 391
 TRAILOF, 543
 TRAILON, 543
 Transformation med vridbar linjäxaxel, 323
 Transformationer
 Kinematikoberoende grundläge för
 verktygsorienteringen, 314
 Kinematiska transformationer, 314
 Orienteringstransformation, 313
 sammanlänkade, 365
 Sammanlänkade transformationer, 315
 Tre- fyr- och fem-axelstransformation, 324
 Transformationstyper
 Allmän funktion, 313
 TRANSMIT, 359

TRAORI, 324
 Trigger-händelse
 vid mätning, 271
 TRUE, 28
 TRUNC, 72
 TU, 372

U

ULI, 40
 Underprogram
 -anrop indirekt, 193
 -anrop med parameteröverföring, 188
 -anrop modalt, 191
 -anrop utan parameteröverföring, 186
 Användning, 154
 -namn, 155
 Programmerbar sökväg, 197
 -upprepning, 190
 -återhopp parametrerbart (RETB...), 182
 -återhopp, parametrerbart (RET ...), 176
 Underprogramanrop med sökvägsuppgift och
 parametrar, 196
 UNTIL, 117
 Ursvarvning - CYCLE86
 programmera externt, 742
 Utmatning
 på extern apparat/fil, 652

V

Valfria positioner - CYCLE802
 programmera externt, 763
 Variabel
 Typkonvertering, 82
 Variabler
 användardefinierad, 28
 Typkonvertering, 83
 Varvtalssynkronitet, 575
 VELOLIM, 535
 VELOLIMA, 510
 Verktyg
 -orientering, 433
 -orientering vid framebyte, 449
 -parameter, 401
 Verktygsbärare
 -kinematik, 442
 -orienterbar, 447
 Verktygs-följd, 397

Verkygskompensering
 Kompenseringsminne, 401
 Koordinatsystem för förslitningsvärden, 411
Verkygskorrektörer
 additiva, 404
Verkygsorientering
 banrelativ, 348
Verkygsradiekompensering
 Hörfördröjning, 278
Vridaxlar
 Avståndsvektorer, 442
 Riktningsvektorer, 442
 Vridvinkel, 442
Vridning
 Orienteringsvektorns, 345
Vridning – CYCLE800
 programmera externt, 758

W

WAITC, 575
WAITE, 119
WAITENC, 635
WAITM, 119
WAITMC, 119
WHEN, 597
WHEN-DO, 607
WHENEVER, 597
WHENEVER-DO, 607
WHILE, 117
WORKPIECE, 666
WRITE, 143

X

XOR, 75